

Package ‘brisk’

July 22, 2025

Title Bayesian Benefit Risk Analysis

Version 0.1.0

Description Quantitative methods for benefit-risk analysis help to condense complex decisions into a univariate metric describing the overall benefit relative to risk. One approach is to use the multi-criteria decision analysis framework (MCDA), as in Mussen, Salek, and Walker (2007) <[doi:10.1002/pds.1435](https://doi.org/10.1002/pds.1435)>. Bayesian benefit-risk analysis incorporates uncertainty through posterior distributions which are inputs to the benefit-risk framework. The brisk package provides functions to assist with Bayesian benefit-risk analyses, such as MCDA. Users input posterior samples, utility functions, weights, and the package outputs quantitative benefit-risk scores. The posterior of the benefit-risk scores for each group can be compared. Some plotting capabilities are also included.

License MIT + file LICENSE

Imports dplyr (>= 1.0), ellipsis (>= 0.3), ggplot2 (>= 3.3), hitandrun (>= 0.5), purrr (>= 0.3), rlang (>= 1.0), tidyr (>= 1.1)

Encoding UTF-8

RoxygenNote 7.2.1

Suggests knitr, fs (>= 1.5), testthat (>= 3.0.0), tibble (>= 3.1), rmarkdown

Config/testthat/edition 3

VignetteBuilder knitr

URL <https://rich-payne.github.io/brisk/>

BugReports <https://github.com/rich-payne/brisk/issues>

NeedsCompilation no

Author Richard Payne [aut, cre],
Sai Dharmarajan [rev],
Eli Lilly and Company [cph]

Maintainer Richard Payne <paynestatistics@gmail.com>

Repository CRAN

Date/Publication 2022-08-31 08:20:05 UTC

Contents

benefit	2
br_group	3
pbrisk	4
plot.brisk_br	5
plot_utility	7
sim_weights	8
summary.brisk_br	9
Index	11

benefit	<i>Bayesian Benefit Risk</i>
---------	------------------------------

Description

Bayesian Benefit Risk

Usage

```
benefit(name, fun, weight)

risk(name, fun, weight)

br(...)

mcda(...)
```

Arguments

name	a string indicating the name of the benefit or risk.
fun	a utility function which maps a parameter value to a utility value.
weight	the weight of the benefit/risk.
...	calls to benefit(), risk(), and br_group() to define the utility functions and treatment groups.

Details

The br() function allows the user to define an arbitrary number of "benefits" and "risks". Each benefit/risk requires a utility function (fun) and a weight. The utility function maps the benefit/risk parameter to a utility score. The br_group() function supplies samples from the posterior distribution for each benefit risk for a specific group (e.g. treatment arm).

The br() function then calculates the posterior distribution of the overall utility for each group. The overall utility is a weighted sum of the utilities for each benefit/risk.

The mcda() function is the same as br(), but has extra checks to ensure that the total weight of all benefits and risks is 1, and that the utility functions produce values between 0 and 1 for all posterior samples.

Value

A named list with posterior summaries of utility for each group and the raw posterior utility scores.

Examples

```
set.seed(1132)
ilogit <- function(x) 1 / (1 + exp(-x))
out <- mcda(
  benefit("CV", function(x) ilogit(x), weight = .75),
  risk("DVT", function(x) ilogit(-.5 * x), weight = .25),
  br_group(
    label = "PBO",
    CV = rnorm(1e4, .1),
    DVT = rnorm(1e4, .1)
  ),
  br_group(
    label = "TRT",
    CV = rnorm(1e4, 2),
    DVT = rnorm(1e4, 1)
  )
)

out

summary(out, probs = c(.025, .5, .975))
summary(out, reference = "PBO")

plot(out)
plot(out, reference = "PBO")

plot_utility(out)
plot_utility(out, reference = "PBO")
plot_utility(out, stacked = TRUE)
```

br_group

*Posterior Samples for a Benefit/Risk Group***Description**

Posterior Samples for a Benefit/Risk Group

Usage

```
br_group(label, ...)
```

Arguments

label	a string indicating the name of the group.
...	named arguments which correspond to the names of the benefits/risks specified by benefit() and risk() in a call to br().

Details

This function is intended to be used as an input argument to the `br()` function.

Value

A named list with the posterior samples and an assigned S3 class.

Examples

```
set.seed(1132)
out <- br(
  benefit("CV", function(x) x, weight = 1),
  risk("DVT", function(x) - .5 * x, weight = 1),

  br_group(
    label = "PBO",
    CV = rnorm(1e4, .1),
    DVT = rnorm(1e4, .1)
  ),
  br_group(
    label = "TRT",
    CV = rnorm(1e4, 2),
    DVT = rnorm(1e4, 1)
  )
)

out

summary(out, probs = c(.025, .5, .975))
summary(out, reference = "PBO")

plot(out)
# adjusted relative to PBO
plot(out, reference = "PBO")

plot_utility(out)
plot_utility(out, reference = "PBO")
plot_utility(out, stacked = TRUE)
```

pbrisk

Calculate Quantiles and Probabilities

Description

Calculates posterior quantiles and probabilities on benefit-risk scores.

Usage

```
pbrisk(x, q, reference = NULL, direction = c("upper", "lower"))

qbrisk(x, p, reference = NULL)
```

Arguments

x	output from a call to <code>brisk::br()</code> or <code>brisk::mcda()</code> .
q	vector of quantiles.
reference	a string indicating which group is the reference group which is used to subtract scores from other groups.
direction	the direction of the posterior probability to compute.
p	a vector of probabilities from which to compute posterior quantiles.

Value

A tibble with the quantile and posterior probability of the benefit-risk score for each group.

Examples

```
set.seed(1132)
out <- br(
  benefit("CV", function(x) x, weight = 1),
  risk("DVT", function(x) - .5 * x, weight = 1),

  br_group(
    label = "PB0",
    CV = rnorm(1e4, .1),
    DVT = rnorm(1e4, .1)
  ),
  br_group(
    label = "TRT",
    CV = rnorm(1e4, 2),
    DVT = rnorm(1e4, 1)
  )
)

pbrisk(out, q = c(.03, .04))
pbrisk(out, q = c(.03, .04), direction = "lower")
pbrisk(out, q = c(.03, .04), reference = "PB0")

qbrisk(out, p = c(.025, .975))
qbrisk(out, p = c(.025, .975), reference = "PB0")
```

plot.brisk_br

*Plot Benefit/Risk Posterior Scores***Description**

Plot Benefit/Risk Posterior Scores

Usage

```
## S3 method for class 'brisk_br'
plot(x, reference = NULL, ...)
```

Arguments

<code>x</code>	output from a call to <code>br()</code> or <code>mcda()</code> .
<code>reference</code>	a string indicating which group is the reference group which is used to subtract scores from other groups.
<code>...</code>	additional arguments throw an error.

Value

A ggplot object plotting the posterior densities of the weighted utility scores.

See Also

Other plots: [plot_utility\(\)](#)

Examples

```
set.seed(1132)
ilogit <- function(x) 1 / (1 + exp(-x))
out <- mcda(
  benefit("CV", function(x) ilogit(x), weight = .75),
  risk("DVT", function(x) ilogit(-.5 * x), weight = .25),
  br_group(
    label = "PB0",
    CV = rnorm(1e4, .1),
    DVT = rnorm(1e4, .1)
  ),
  br_group(
    label = "TRT",
    CV = rnorm(1e4, 2),
    DVT = rnorm(1e4, 1)
  )
)

out

summary(out, probs = c(.025, .5, .975))
summary(out, reference = "PB0")

plot(out)
plot(out, reference = "PB0")

plot_utility(out)
plot_utility(out, reference = "PB0")
plot_utility(out, stacked = TRUE)
```

plot_utility	<i>Plot Posterior Mean Utility Scores</i>
--------------	---

Description

Plot Posterior Mean Utility Scores

Usage

```
plot_utility(x, reference = NULL, stacked = FALSE)
```

Arguments

x	output from a call to <code>br()</code> or <code>mcda()</code> .
reference	a string indicating which group is the reference group which is used to subtract scores from other groups.
stacked	logical indicating if a stacked version of the barplot should be produced.

Value

A ggplot barplot of the posterior mean weighted utility scores.

See Also

Other plots: [plot.brisk_br\(\)](#)

Examples

```
set.seed(1132)
ilogit <- function(x) 1 / (1 + exp(-x))
out <- mcda(
  benefit("CV", function(x) ilogit(x), weight = .75),
  risk("DVT", function(x) ilogit(-.5 * x), weight = .25),
  br_group(
    label = "PB0",
    CV = rnorm(1e4, .1),
    DVT = rnorm(1e4, .1)
  ),
  br_group(
    label = "TRT",
    CV = rnorm(1e4, 2),
    DVT = rnorm(1e4, 1)
  )
)

out

summary(out, probs = c(.025, .5, .975))
summary(out, reference = "PB0")
```

```

plot(out)
plot(out, reference = "PBO")

plot_utility(out)
plot_utility(out, reference = "PBO")
plot_utility(out, stacked = TRUE)

```

sim_weights

Simulate Normalized Weights

Description

Simulate Normalized Weights

Usage

```
sim_weights(n, ...)
```

Arguments

n	number of weights to simulate.
...	vectors of length 2 indicating the lower and upper bound (respectively) of the un-normalized weights. At least one set of bounds must be equal to each other (e.g. c(1, 1)) and be the largest set of bounds in the set specified.

Details

The weights are normalized relative to a set of bounds which are equal to each other (e.g. c(1, 1)), and also are the largest set of bounds in the set specified. See Example.

Value

A tibble with weights for each argument supplied to ... Each column represents the weights, and each row (total of n rows) is a set of random weights across groups. Column names are obtained from the argument names of ..., if supplied.

Examples

```

w <- sim_weights(1e4, a = c(1, 1), b = c(.4, .6), c = c(.2, .3))
# ratio of b to a is between c(.4, .6) / c(1, 1)
summary(w$b / w$a)
# ratio of c to a is between c(.2, .3) / c(1, 1)
summary(w$c / w$a)

# Weights can be used to add uncertainty to the benefit/risk analysis
set.seed(1132)
ilogit <- function(x) 1 / (1 + exp(-x))
out <- mcda(

```



```

benefit("CV", function(x) ilogit(x), weight = w$a),
risk("DVT", function(x) ilogit(- .5 * x), weight = w$b),
risk("MI", function(x) ilogit(- .5 * x), weight = w$c),
br_group(
  label = "PBO",
  CV = rnorm(1e4, .1),
  DVT = rnorm(1e4, .1),
  MI = rnorm(1e4, .1)
),
br_group(
  label = "TRT",
  CV = rnorm(1e4, 2),
  DVT = rnorm(1e4, 1),
  MI = rnorm(1e4, 0.5)
)
)

```

summary.brisk_br

*Summarize Bayesian Benefit-Risk Scores***Description**

Summarize Bayesian Benefit-Risk Scores

Usage

```

## S3 method for class 'brisk_br'
summary(object, probs = c(0.025, 0.975), reference = NULL, ...)

```

Arguments

object	output from a call to <code>brisk::br()</code> or <code>brisk::mcda()</code> .
probs	a vector of probabilities used to obtain quantiles of the posterior of the weighted utilities for each group.
reference	a string indicating which group is the reference group which is used to subtract scores from other groups.
...	Additional arguments which throw an error if specified.

Value

A named list with the posterior summary, and the scores from the object `object` (which are adjusted if `reference` is specified).

Examples

```
set.seed(1132)
out <- br(
  benefit("CV", function(x) x, weight = 1),
  risk("DVT", function(x) - .5 * x, weight = 1),

  br_group(
    label = "PBO",
    CV = rnorm(1e4, .1),
    DVT = rnorm(1e4, .1)
  ),
  br_group(
    label = "TRT",
    CV = rnorm(1e4, 2),
    DVT = rnorm(1e4, 1)
  )
)

out

summary(out, probs = c(.025, .5, .975))
summary(out, reference = "PBO")

plot(out)
# adjusted relative to PBO
plot(out, reference = "PBO")

plot_utility(out)
plot_utility(out, reference = "PBO")
plot_utility(out, stacked = TRUE)
```

Index

* **plots**

- plot.brisk_br, [5](#)
- plot_utility, [7](#)

benefit, [2](#)

br (benefit), [2](#)

br_group, [3](#)

mcda (benefit), [2](#)

pbrisk, [4](#)

plot.brisk_br, [5](#), [7](#)

plot_utility, [6](#), [7](#)

qbrisk (pbrisk), [4](#)

risk (benefit), [2](#)

sim_weights, [8](#)

summary.brisk_br, [9](#)