Package: door (via r-universe)

May 20, 2026

Type Package

Title Analysis of Clinical Trials with the Desirability of Outcome Ranking Methodology

Version 0.0.3

Description Statistical methods and related graphical representations for the Desirability of Outcome Ranking (DOOR) methodology. The DOOR is a paradigm for the design, analysis, interpretation of clinical trials and other research studies based on the patient centric benefit risk evaluation. The package provides functions for generating summary statistics from individual level/summary level datasets, conduct DOOR probability-based inference, and visualization of the results. For more details of DOOR methodology, see Hamasaki and Evans (2025) <doi:10.1201/9781003390855>. For more explanation of the statistical methods and the graphics, see the technical document and user manual of the DOOR 'Shiny' apps at <https://methods.bsc.gwu.edu>.

Imports dplyr, tidyr, ggplot2, forestplot, scales, methods, labeling

License GPL (>= 3)

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Collate 'DOOR.R' 'door_summary.R' 'calc_doorprob.R' 'check.HLP.R' 'check.homog.R' 'create.U.R' 'create.Z.ZF.R' 'data.R' 'door_barplot.R' 'var_pi.R' 'door_ci.R' 'door_test.R' 'forest_plot.R' 'mph.fit.R' 'mph.summary.R' 'num.deriv.fct.R' 'partial_credit_analysis.R' 'partial_credit_plots.R'

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Depends R (>= 3.5)

NeedsCompilation no

Author Yijie He [aut, cre], Qihang Wu [ctb], Toshimitsu Hamasaki [ctb]

Maintainer Yijie He <yih148@gwu.edu>

Config/pak/sysreqs libicu-dev

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

Date/Publication 2025-08-23 02:30:02 UTC

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

RemoteRef HEAD

RemoteSha 2b7175b6630635728aec4f304e85980515665e57

Contents

`calc_doorprob` *Calculate DOOR probability*

Description

For summary level data, `y1` and `y2` should be given. For individual level data, a `summary_obj` should be given.

Usage

```
calc_doorprob(
  y1 = NULL,
  y2 = NULL,
  data_type = c("freq", "prop"),
  summary_obj = NULL
)
```

Arguments

y1	A vector of proportion or frequency distribution for group 1
y2	A vector of proportion or frequency distribution for group 2
data_type	Either "freq" for frequency input or "prop" for proportion input when using y1 and y2
summary_obj	A object returned by door_summary(); Alternative input for y1 and y2

Value

DOOR probability

See Also

door_summary()

Examples

```
y1 = c(60, 30, 10)
y2 = c(50, 40, 10)
calc_doorprob(y1, y2)

## DOOR probability
##            0.545

p1 = c(.6, .3, .1)
p2 = c(.5, .4, .1)
calc_doorprob(p1, p2, data_type = "prop")

## DOOR probability
##            0.545
```

---

door_barplot *Create DOOR summary barplot*

Description

Create DOOR summary barplot

Usage

```
door_barplot(
  y1 = NULL,
  y2 = NULL,
  summary_obj = NULL,
  data_type = c("freq", "prop")
)
```

Arguments

y1, y2	Numeric vectors of DOOR proportion or frequency distribution for group 1, group 2. The entries should be ordered from most desirable to least desirable
summary_obj	An object returned by door_summary(); Alternative input for y1 and y2
data_type	Either "freq" for frequency input or "prop" for proportion input when using y1 and y2

Value

a ggplot object

Examples

```
y1 = c(60, 30, 10)
y2 = c(50, 40, 10)
door_barplot(y1, y2)
```

---

door_ci *Calculate confidence intervals for DOOR probability*

Description

This is a wrapper function for all CI calculation functions

Usage

```
door_ci(
  y1 = NULL,
  y2 = NULL,
  n1 = NULL,
  n2 = NULL,
  summary_obj = NULL,
  conf_level = 0.95,
  data_type = c("freq", "prop"),
  ci_method = c("all", "halperin", "ps_h", "tanh"),
  ...
)
```

Arguments

y1, y2	Numeric vectors of DOOR proportion or frequency distribution for group 1, group 2. The entries should be ordered from most desirable to least desirable
n1, n2	Sample sizes of group 1, group 2; must be specified if method = "prop"
summary_obj	A object returned by individual_to_summary(); Alternative input for y1 and y2
conf_level	Confidence level

data_type	Either "freq" for frequency input or "prop" for proportion input when using y1 and y2
ci_method	One of "all" for all available methods, "halperin" for Halperin et al. (1989)'s method, "ps_h" for pseudo-score approach for Halperin's method, "tanh" for inverse hyperbolc tangent transformed method
...	Additional parameters passed for calculating pseudo-score type confidence interval

Value

List of CIs

See Also

halperin_ci(), pseudo_score_ci()

Examples

```
door_ci(c(60,30,10), c(50,40,10), ci_method = "all")
```

---

door_component_barplot

Create DOOR component barplot

Description

Create DOOR component barplot

Usage

```
door_component_barplot(
  comp_table = NULL,
  n1 = NULL,
  n2 = NULL,
  summary_obj = NULL,
  data_type = c("freq", "prop")
)
```

Arguments

comp_table	A DOOR component table
n1, n2	Sample sizes of group 1, group 2
summary_obj	An object returned by individual_to_summary(); Alternative input for comp_table.
data_type	Either "freq" for frequency input or "prop" for proportion input if "comp_table" is used

Value

A ggplot object

Examples

```
comp_table = data.frame(compname = c("A", "B"), trt = c(30, 20), ctr = c(40, 25))
door_component_barplot(comp_table = comp_table, n1 = 100, n2 = 100)
```

`door_component_forestplot`

Create DOOR component forest plot

Description

Create DOOR component forest plot

Usage

```
door_component_forestplot(
  comp_table = NULL,
  y1 = NULL,
  y2 = NULL,
  n1 = NULL,
  n2 = NULL,
  data_type = c("freq", "prop"),
  summary_obj = NULL,
  conf_level = 0.95,
  ci_method = c("halperin", "ps_h", "ps_tanh")
)
```

Arguments

`comp_table`	a data frame of DOOR components. See example.
`y1`, `y2`	Numeric vectors of DOOR proportion or frequency distribution for group 1, group 2. The entries should be ordered from most desirable to least desirable
`n1`, `n2`	Sample sizes of group 1, group 2; must be specified if method = "prop"
`data_type`	Either "freq" for frequency input or "prop" for proportion input when using y1 and y2
`summary_obj`	An object returned by `individual_to_summary()`; Alternative input for y1 and y2
`conf_level`	confidence level
`ci_method`	method for confidence interval calculation; one of "halperin", "ps_h", "ps_tanh"

Value

a forest plot object

Examples

```
comp_table = data.frame(compname = c("A", "B"), trt = c(30, 20), ctr = c(40, 25))
y1 = c(60, 30, 10)
y2 = c(60, 30, 10)
door_component_forestplot(comp_table = comp_table,
                          y1 = y1,
                          y2 = y2)
```

`door_cumulative_forestplot` *Generate cumulative DOOR forest plot*

Description

Generate cumulative DOOR forest plot

Usage

```
door_cumulative_forestplot(
  y1 = NULL,
  y2 = NULL,
  n1 = NULL,
  n2 = NULL,
  data_type = c("freq", "prop"),
  summary_obj = NULL,
  conf_level = 0.95,
  ci_method = c("halperin", "ps_h", "tanh")
)
```

Arguments

y1, y2	Numeric vectors of DOOR proportion or frequency distribution for group 1, group 2. The entries should be ordered from most desirable to least desirable
n1, n2	Sample sizes of group 1, group 2; must be specified if method = "prop"
data_type	Either "freq" for frequency input or "prop" for proportion input when using y1 and y2
summary_obj	An object returned by `individual_to_summary()`; Alternative input for y1 and y2
conf_level	confidence level
ci_method	methods for calculating confidence interval

Value

a forestplot object

Examples

```
y1 = c(60, 30, 10)
y2 = c(50, 40, 10)
door_cumulative_forestplot(y1, y2)
```

---

door_summary *Summarize individual level data into summary level data*

Description

Transform an individual level dataset that contains DOOR outcome variable and treatment/intervention variable to summary level. By default, the levels of the DOOR outcome is ordered from 1 to K.

Usage

```
door_summary(
  data,
  trtVar,
  doorVar,
  trtCodes,
  trtLabels = NULL,
  compVars = NULL,
  decreasing = FALSE
)
```

Arguments

data	Data frame that includes DOOR outcome variable and treatment variable at individual level
trtVar	Variable name of treatments
doorVar	Variable name of DOOR outcome; the doorVar should be numeric
trtCodes	A numeric vector contains the codes for interventions in trtVar, ordered by c(trt, ctr)
trtLabels	An optional vector contains the intervention labels for trtCodes, ordered by c(trt, ctr)
compVars	An optional character vector of variable names of DOOR components
decreasing	A logical value indicating the order of desirability of the DOOR levels. By default, smaller value represents better outcomes

Value

An object of DOOR outcome distribution summary

Examples

```
data(mock_raw_data)
door_summary(data = mock_raw_data,
             trtVar = "ARM",
             doorVar = "DOOR",
             trtCodes = c(1,2),
             trtLabels = c("Test, Control"),
             compVars = c("infectious complications", "clinical failure", "death"))
```

`door_test` *Hypothesis testing for the DOOR probability*

Description

Hypothesis testing for the DOOR probability

Usage

```
door_test(
  y1 = NULL,
  y2 = NULL,
  n1 = NULL,
  n2 = NULL,
  summary_obj = NULL,
  data_type = c("freq", "prop"),
  null_value = 0.5,
  alternative = c("two.sided", "less", "greater")
)
```

Arguments

`y1`, `y2`	Numeric vectors of DOOR proportion or frequency distribution for group 1, group 2. The entries should be ordered from most desirable to least desirable
`n1`, `n2`	Sample sizes of group 1, group 2; must be specified if method = "prop"
`summary_obj`	An object returned by `individual_to_summary()`; Alternative input for y1 and y2
`data_type`	Either "freq" for frequency input or "prop" for proportion input when using y1 and y2
`null_value`	A number specifying the hypothesized value of the DOOR probability
`alternative`	A character describing the alternative hypothesis

Value

A `htest` object containing information of hypothesis test of DOOR probability

See Also

door_summary()

Examples

```
y1 <- c(60, 30, 10)
y2 <- c(50, 40, 10)
door_test(y1 = y1, y2 = y2)

## Hypothesis test for DOOR probability
## data:  y1 and y2
## WMW statistic = 1.2372, p-value = 0.216
## alternative hypothesis: true  is not equal to 0.5
## sample estimates:
## DOOR probability
##            0.545
```

---

halperin_ci *Calculate confidence interval of DOOR probability based on Halperin et al. (1989)'s method*

Description

Calculate confidence interval of DOOR probability based on Halperin et al. (1989)'s method

Usage

```
halperin_ci(
  y1 = NULL,
  y2 = NULL,
  n1 = NULL,
  n2 = NULL,
  data_type = c("freq", "prop"),
  summary_obj = NULL,
  conf_level = 0.95
)
```

Arguments

y1, y2	Numeric vectors of DOOR proportion or frequency distribution for group 1, group 2. The entries should be ordered from most desirable to least desirable
n1, n2	Sample sizes of group 1, group 2; must be specified if data_type = "prop"
data_type	Either "freq" for frequency input or "prop" for proportion input when using y1 and y2
summary_obj	A object returned by door_summary(); Alternative input for y1 and y2
conf_level	Confidence level

Value

Halperin et al. (1989)'s CI

References

reference

See Also

door_ci()

Examples

```
y1 = c(60, 30, 10)
y2 = c(50, 40, 10)
halperin_ci(y1, y2)

## $halperin_ci
## [1] 0.4734504 0.6147386
```

---

inv_tanh_ci *Calculate confidence interval of DOOR probability based on inverse hyperbolic tangent transformation of Wald-type CI*

Description

Calculate confidence interval of DOOR probability based on inverse hyperbolic tangent transformation of Wald-type CI

Usage

```
inv_tanh_ci(
  y1 = NULL,
  y2 = NULL,
  n1 = NULL,
  n2 = NULL,
  data_type = c("freq", "prop"),
  summary_obj = NULL,
  conf_level = 0.95
)
```

Arguments

y1, y2	Numeric vectors of DOOR proportion or frequency distribution for group 1, group 2. The entries should be ordered from most desirable to least desirable
n1, n2	Sample sizes of group 1, group 2; must be specified if method = "prop"

data_type	Either "freq" for frequency input or "prop" for proportion input when using y1 and y2
summary_obj	A object returned by individual_to_summary(); Alternative input for y1 and y2
conf_level	Confidence level

Value

Inverse hyperbolic tangent transformation CI

See Also

door_ci()

Examples

```
inv_tanh_ci(c(60,30,10), c(50,40,10))
```

---

mock_raw_data *Mock Raw Data*

Description

A mock data that contains examples of a raw dataset of DOOR outcomes, treatment information, and DOOR components

Usage

```
data(mock_raw_data)
```

Format

A data frame with 55 observations with the following columns

ARM, Arm text Codes and labels of treatment arm

DOOR, DOORtext Codes and labels of DOOR outcome

clinical failure, clinical failure text Codes and labels for one of the DOOR components

infectious complications, infectious complications text Codes and labels for one of the DOOR components

death, death text Codes and labels for one of the DOOR components

Weight IPW weights

Duration Tie breaker

`partial_credit_analysis` *Partial credit analysis for DOOR*

Description

Partial credit analysis for DOOR

Usage

```
partial_credit_analysis(
  grade_key,
  y1 = NULL,
  y2 = NULL,
  n1 = NULL,
  n2 = NULL,
  summary_obj = NULL,
  data_type = c("freq", "prop"),
  ci_method = "halperin",
  conf_level = 0.95,
  ...
)
```

Arguments

`grade_key`	A numeric vector of grade key or a dataframe contains columns of grade keys
`y1, y2`	Numeric vectors of DOOR proportion or frequency distribution for group 1, group 2. The entries should be ordered from most desirable to least desirable
`n1, n2`	Sample sizes of group 1, group 2; must be specified if method = "prop"
`summary_obj`	An object returned by `door_summary()`; Alternative input for y1 and y2
`data_type`	Either "freq" for frequency input or "prop" for proportion input when using y1 and y2
`ci_method`	Specify the type of CI for DOOR probability given a grade key. The default is "halperin" for Halperin et al. (1989)'s method. Other options include "ps_h" for pseudo-score approach for Halperin's method and "tanh" for inverse hyperbolc tangent transformed method
`conf_level`	Confidence level
`...`	Optional additional parameters if `ci_method = "ps_h"`

Value

An object containing information of partial credit analysis given grade keys

Examples

```
grade.key <- c(100, 80, 60, 40, 0)
y1 <- c(60, 30, 20, 10, 5)
y2 <- c(50, 40, 10, 20, 5)
partial_credit_analysis(grade_key = grade.key, y1 = y1, y2 = y2)
```

`partial_credit_biplot` *Partial credit plot*

Description

Partial credit plot

Usage

```
partial_credit_biplot(pc_object, ...)
```

Arguments

`pc_object`	an object returned by `partial_credit_analysis()`
`...`	additional arguments for other functions

Value

a plot object

Examples

```
grade.key <- c(100, 60, 0)
y1 <- c(60, 30, 10)
y2 <- c(50, 40, 10)
pc_object <- partial_credit_analysis(grade_key = grade.key, y1 = y1, y2 = y2)
partial_credit_biplot(pc_object)
```

`partial_credit_contour_plot`
Generate contour plot for partial credit analysis

Description

The contour plot is for sensitivity analysis. Currently it supports given DOOR outcome categories of three or four. The contour plot assigns every combinations of grade keys given a DOOR outcome distribution

Usage

```
partial_credit_contour_plot(
  y1 = NULL,
  y2 = NULL,
  n1 = NULL,
  n2 = NULL,
  summary_obj = NULL,
  data_type = c("freq", "prop"),
  pc_inc = 10,
  contour_inc = 1
)
```

Arguments

y1, y2	Numeric vectors of DOOR proportion or frequency distribution for group 1, group 2. The entries should be ordered from most desirable to least desirable
n1, n2	Sample sizes of group 1, group 2; must be specified if method = "prop"
summary_obj	An object returned by `individual_to_summary()`; Alternative input for y1 and y2
data_type	Either "freq" for frequency input or "prop" for proportion input when using y1 and y2
pc_inc	Increment of partial credits
contour_inc	Increment of contour lines

Value

A graph object

Examples

```
y1 <- c(60, 30, 10)
y2 <- c(50, 40, 10)
partial_credit_contour_plot(y1, y2)
```

---

pseudo_score_ci *Calculate pseudo score type confidence interval of DOOR probability*

Description

Some code of this function is adpated from the now-archived CRAN package "cta", originally authored by Joseph B. Lang. The original package was licensed under GPL-2, and the adapted code complies with this license.

Usage

```
pseudo_score_ci(
  y1 = NULL,
  y2 = NULL,
  n1 = NULL,
  n2 = NULL,
  summary_obj = NULL,
  data_type = c("freq", "prop"),
  cil = 0.4,
  ciu = 0.6,
  conf_level = 0.95,
  epsilon = 1e-04,
  maxiter = 100
)
```

Arguments

y1, y2	Numeric vectors of DOOR proportion or frequency distribution for group 1, group 2. The entries should be ordered from most desirable to least desirable
n1, n2	Sample sizes of group 1, group 2; must be specified if method = "prop"
summary_obj	A object returned by `individual_to_summary()`; Alternative input for y1 and y2
data_type	Either "freq" for frequency input or "prop" for proportion input when using y1 and y2
cil, ciu	Initial guesses of lower and upper limit, respectively
conf_level	Confidence level
epsilon	Convergence tolerance. Default to 1e-4
maxiter	Maximum iteration

Value

pseudo-score type CI and the number of iterations to calculate the lower bound and upper bound

See Also

`door_ci()`

Examples

```
pseudo_score_ci(c(60,30,10), c(50,40,10))
```

var_pi *Variance of DOOR probability*

Description

Calculate multiple types of variances of DOOR probability

Usage

```
var_pi(p1, p2, n1, n2)
```

Arguments

p1	Vector of DOOR outcome proportion distribution for group 1
p2	Vector of DOOR outcome proportion distribution for group 2
n1	Sample size of group 1
n2	Sample size of group 2

Value

DOOR probability, Wald-type variance, exact variance, Halperin variance; theta for Halperin method

Index