Package 'anomo'

July 22, 2025

Type Package

Title Analysis of Moderation, Statistical Power, and Optimal Design
for Studies Detecting Difference and Equivalence

Version 1.0.0

Date 2025-2-22

Description Analysis of moderation (ANOMO) method conceptualizes
the difference and equivalence tests as a moderation problem to
test the difference and equivalence of
two means (or two effects in two studies).

Language en-US

Depends R (>= 4.0.0), graphics (>= 4.0.0), base(>= 4.0.0)

License GPL-3

Encoding UTF-8

Suggests rmarkdown, knitr, markdown, odr

VignetteBuilder rmarkdown, knitr, markdown

RoxygenNote 7.3.2

NeedsCompilation no

Author Zuchao Shen [aut, cre] (ORCID: <https://orcid.org/0000-0003-3483-0451>)

Maintainer Zuchao Shen <zuchao.shen@gmail.com>

Repository CRAN

Date/Publication 2025-02-23 23:30:10 UTC

Contents

---

mcci	*Compute Monte Carlo Confidence Intervals*

Description

Compute Monte Carlo confidence intervals (MCCIs) for the difference and equivalence tests.

Usage

```
mcci(
  d1 = NULL,
  se1 = NULL,
  d2 = NULL,
  se2 = NULL,
  n.mcci = 10000,
  sig.level = 0.05,
  two.tailed = TRUE,
  eq.bd = NULL,
  xlim = NULL,
  xlab = NULL,
  ylab = NULL,
  dashed.lines = TRUE,
  verbose = TRUE
)
```

Arguments

d1	The estimated mean(s) (effect(s)) for group 1 (study 1). If more than one effect is specified, it assumes that the effects are components of a mediation effect.
se1	The estimated standard error for d1. If more than one standard error is specified, it assumes that they are estimated components of a mediation effect.
d2	The estimated mean(s) (effect(s)) for group 2 (study 2). If more than one effect is specified, it assumes that the effects are components of a mediation effect.
se2	The estimated standard error for d2. If more than one standard error is specified, it assumes that they are estimated components of a mediation effect.
n.mcci	The number of draws for the MCCI method. Default is 10,000.
sig.level	The significance level. Default is .05.
two.tailed	Logical of two tailed test for difference test. Default is TRUE.
eq.bd	The limit of the equivalence bounds for an equivalence test. Default is the MCCI for the equivalence test. It can be specified in the arguments as eq.bd = a positive number or eq.bd = c(lower bound #, upper bound #).
xlim	The limits set for the x-axis in the plot. Default is the MCCI for the difference test. It can be specified in the arguments as xlim = c(lower #, higher #).
xlab	The label for the x-axis in the plot. Default is "Differences in Effects".

ylab	The label for the y-axis in the plot. Default is NULL.
dashed.lines	Logical of whether dashed lines of equivalence bounds and zero should be added in the plot. Default is TRUE.
verbose	Logical; print the process if TRUE, otherwise not; default value is TRUE.

Value

The results of moderation analysis and equivalence tests using the MCCI method. It will also provide a plot for the MCCIs.

Examples

```
library(anomo)
# compute MCCI from two studies
 myci <- mcci(d1 = .1, se1 = .1, d2 = .2, se2 = .1)
# compute MCCI from one study
myci <- mcci(d1 = .1, se1 = .1)

# See the package vignettes for more examples, including the MCCI for the
# test of significance and equivalence for mediation effects in two studies.
```

---

od.eq.2group	*Optimal sample allocation calculation for equivalence test of two-group means*

Description

The optimal design of single-level experiments detecting equivalence of two-group means is to choose the optimal sample allocation that minimizes the variance of a treatment effect under a fixed budget, which is approximately the optimal sample allocation that maximizes statistical power under a fixed budget. The optimal design parameter is the proportion of individuals to be assigned to treatment (p).

Usage

```
od.eq.2group(
  p = NULL,
  r12 = NULL,
  c1 = NULL,
  c1t = NULL,
  m = NULL,
  plots = TRUE,
  plim = NULL,
  varlim = NULL,
  plab = NULL,
  varlab = NULL,
```

```
    vartitle = NULL,
    verbose = TRUE
)
```

Arguments

p	The proportion of individuals to be assigned to treatment.
r12	The proportion of outcome variance explained by covariates.
c1	The cost of sampling one unit in the control condition.
c1t	The cost of sampling one unit in the treated condition.
m	Total budget, default value is the total costs of sampling 600 individuals across treatment conditions.
plots	Logical, provide variance plots if TRUE, otherwise not; default value is TRUE.
plim	The plot range for p, default value is c(0, 1).
varlim	The plot range for variance, default value is c(0, 0.05).
plab	The plot label for p, default value is "Proportion of Units in Treatment: p".
varlab	The plot label for variance, default value is "Variance".
vartitle	The title of variance plot, default value is NULL.
verbose	Logical; print the value of p if TRUE, otherwise not; default value is TRUE.

Value

Unconstrained or constrained optimal sample allocation (p). The function also returns function name, design type, and parameters used in the calculation.

Examples

```
# Unconstrained optimal design #---------
  myod <- od.eq.2group(r12 = 0.5, c1 = 1, c1t = 50)
  myod$out # output
```

---

power.eq.2group   *Statistical Power Analysis for Equivalence Tests of Two-Group Means*

Description

Statistical power analysis for equivalence test of two-group means.

Usage

```
power.eq.2group(
  cost.model = FALSE,
  expr = NULL,
  constraint = NULL,
  d = NULL,
  eq.dis = NULL,
  m = NULL,
  c1 = NULL,
  c1t = NULL,
  n = NULL,
  p = NULL,
  q = 1,
  sig.level = 0.05,
  r12 = NULL,
  power = NULL,
  powerlim = NULL,
  nlim = NULL,
  mlim = NULL,
  eq.dislim = NULL,
  verbose = TRUE
)
```

Arguments

cost.model	Logical; power analyses accommodating costs and budget (e.g., required budget for desired power, power, minimum detectable eq.dis under a fixed budget) if TRUE. Otherwise, conventional power analysis is performed (e.g., required sample size, power, or minimum detectable eq.dis calculation); default value is FALSE, and it will be changed to TRUE if expr is not NULL.
expr	Returned object from function [od.eq.2group](); default value is NULL; if expr is specified, parameter values of r12, c1, c1t, and p used or solved in function [od.eq.2group]() will be passed to the current function; only the value of p that specified or solved in function [od.eq.2group]() can be overwritten if constraint is specified.
constraint	Specify the constrained value of p in list format to overwrite that from expr; default value is NULL.
d	The estimated difference in two-group means.
eq.dis	A positive number to specify the distance from equivalence bounds to d. The equivalence bounds are c(-abs(d)-eq.dis, abs(d)+eq.dis).
m	Total budget.
c1	The cost of sampling one unit in the control condition.
c1t	The cost of sampling one unit in the treated condition.
n	The total sample size across groups.
p	The proportion of individuals in the intervention group or group 1.

q	The number of predictors in the combined linear regression model. Default is 1.
sig.level	The significance level. Default is .05.
r12	The proportion of variance explained by covariates if any.
power	Statistical power.
powerlim	The range for solving the root of power (power) numerically, default value is c(1e-10, 1 - 1e-10).
nlim	The range for searching the root of sample size (n) numerically, default value is c(4, 10e10).
mlim	The range for searching the root of budget (m) numerically, default value is the costs sampling nlim units across treatment conditions or c(4 * ncost, 10e10 * ncost) with ncost = ((1 - p) * c1 + p * c1t).
eq.dislim	The range for solving the root of equivalence difference with the effect size (d) numerically, default value is c(0, 10).
verbose	Logical; print the process if TRUE, otherwise not; default value is TRUE.

Value

Required budget (and/or required sample size), statistical power, or minimum detectable eq.dis depending on the specification of parameters. The function also returns the function name, design type, and parameters used in the calculation.

Examples

```
    library(anomo)
# 1. Conventional Power Analyses from Difference Perspectives
# Calculate the required sample size to achieve certain level of power
mysample <- power.eq.2group(d = .1, eq.dis = 0.1,  p =.5,
                            r12 = .5, q = 1, power = .8)
mysample$out

# Calculate power provided by a sample size allocation
mypower <- power.eq.2group(d = 1, eq.dis = .1, n = 1238, p =.5,
                           r12 = .5, q = 1)
mypower$out

# Calculate the minimum detectable distance a given sample size allocation
# can achieve
myeq.dis <- power.eq.2group(d = .1, n = 1238, p =.5,
                            r12 = .5, q = 1, power = .8)
myeq.dis$out

# 2. Power Analyses Using Optimal Sample Allocation
myod <- od.eq.2group(r12 = 0.5, c1 = 1, c1t = 10)
budget <- power.eq.2group(expr = myod, d = .1, eq.dis = 0.1,
                          q = 1, power = .8)
budget.balanced <- power.eq.2group(expr = myod, d = .1, eq.dis = 0.1,
                                   q = 1, power = .8,
                                   constraint = list(p = .50))
(budget.balanced$out$m-budget$out$m)/budget$out$m *100
```

```
# 27% more budget required from the balanced design with p = 0.50.
```

---

re   *Relative efficiency (RE) calculation*

Description

Calculate the relative efficiency (RE) between two designs using the re function from the R package odr.

Usage

```
re(od, subod, rounded = TRUE, verbose = TRUE)
```

Arguments

od	Returned object of first design (e.g., unconstrained optimal design) from function od.eq.2group.
subod	Returned object of second design (e.g., constrained optimal design) from function od.eq.2group.
rounded	Logical; round the values of p to two decimal places if TRUE. No rounding if FALSE; default is TRUE.
verbose	Logical; print the value of relative efficiency if TRUE, otherwise not; default is TRUE.

Value

Relative efficiency value.

Examples

```
# Unconstrained optimal design #----------
  myod1 <- od.eq.2group(r12 = 0.5, c1 = 1, c1t = 20)
# Constrained optimal design with p = .50
  myod2 <- od.eq.2group(r12 = 0.5, c1 = 1, c1t = 20, p = .50)
# Relative efficiency (RE)
  myre <- re(od = myod1, subod= myod2)
  myre$re # RE = 0.71
```

Index