Package 'PowerUpR'

June 14, 2017

Type Package

Title R version of PowerUp!

Version 1.1.3

Date 2017-06-15

Description
R version of PowerUp! allows calculation of statistical power, minimum detectable effect size, and constrained optimal sample allocation for main, and moderator effects in various multilevel randomized experiments.

URL https://github.com/metinbulus/PowerUpR

BugReports https://github.com/metinbulus/PowerUpR/issues

Depends nloptr(>= 1.0.4)

Suggests knitr, rmarkdown

VignetteBuilder knitr

License GPL (>= 3)

Author Metin Bulus [aut, cre, trl],
Nianbo Dong [aut],
Benjamin Kelcey [aut],
Jessaca Spybrook [aut],
Bang Le [aut]

Maintainer Metin Bulus <bulusmetin@gmail.com>

R topics documented:

`introduction` *Introduction to* `PowerUpR`

Description

`PowerUpR` is an implementation of *PowerUp!*, and *PowerUp!-Moderator* in R environment (R Core Team, 2017). *PowerUp!* series consist of convenient excel based functions to conduct statistical power analysis for various experimental and quasi-experimental designs (Dong & Maynard, 2013). It also enables users to conduct statistical power analysis for moderator effects in two- and three-level cluster randomized trials (Dong, Kelcey, & Spybrook, in press; Spybrook, Kelcey, & Dong, in press), and for mediator effects in two-level cluster randomized trials (Kelcey, Dong, Spybrook, & Cox, 2017; Kelcey, Dong, Spybrook, & Shen, in press).

The `PowerUpR` package bases its framework on three fundemental concepts in statistical power analysis; statistical power calculation, minimum detectable effect size calculation, and constrained optimal sample allocation (COSA; Hedges & Borenstein, 2014; Raudenbush, 1997; Raudenbush & Liu, 2000). COSA problems can be solved in the following forms, (i) under budgetary constraints given marginal costs per unit, (ii) under power constraints given marginal costs per unit, (iii) under MDES constraints given marginal costs per unit, and (iv) under sample size constraints for one or more levels along with any of the i, ii or iii options.

A design parameter (one of the MDES, power, or COSA) can be requested by using approriate function given design characteristics.

Each function begins with an **output** name, followed by a period, an **effect** (for moderator only), followed by a period, and a **design** name.

There are three types of output; `mdes`, `power`, and `optimal`, and 14 types of design; `ira1r1`, `bira2r1`, `bira2f1`, `bira2c1`, `cra2r2`, `bira3r1`, `bcra3r2`, `bcra3f2`, `cra3r3`, `bira4r1`, `bcra4r2`, `bcra4r3`, `bcra4f3`, and `cra4r4`. The first three or four letters of the design stands for the type of assignment, for individual random assignment `ira`, for blocked individual random assignment `bira`, for cluster random assignment `cra`, and for blocked cluster random assignment `bcra`. It is followed by a number indicating number of levels. A single letter followed by a number indicates whether a block is considered to be `r`, random; `f`, fixed; or `c`, constant and the level at which random assingment takes place.

A function with two keywords (output.design) returns results for main effect. For moderator effect, there is an additional keyword so the function has three keywords (output.effect.design). There are three moderator effects available for design `cra2r2` and five moderator effect available for design `cra3r3`. In total there are six moderator effects; `mod1n`, `mod1r`, `mod2`, `mod2n`, `mod2r`, and `mod3`.The number and the single letter at the end stands for the level of moderator variable and whether it varies randomly or non-randomly across higher level unit.

So, to find MDES for main effect in two-level cluster randomized design where random assignment is at level-2, function `mdes.cra2r2` is used. Similiarly, to find MDES for a non-randoly varying moderator effect at level-1 for the same design, function `mdes.mod1n.cra2r2` is used.

Each function requires slightly different arguments depending on the output it produces and the design. Most of the arguments have default values to provide users a starting point, which can be found in *usage* section of the documentation. For all functions default values are

- `mdes` = .25
- `power` $(1 - \beta)$ = .80
- `alpha` (α) = .05
- `two.tail` = `TRUE`
- `P` = .50

and depending on the effect and design

- any of one of `g1, g2, g3, g4` = 0
- any sequence of `R12, R22, R32, R42` = 0
- any sequence of `RT22, RT32, RT42` = 0
- `Q` = `NULL`, implies continuous moderator

Users should be aware of default values and change them if necessary. Depending on the function minimum required arguments are

- any sequence of `rho2, rho3, rho4`
- any sequence of `omega2, omega3, omega4`
- any one of, any sequence of, or any combination of `n, J, K, L`

For definition of above-mentioned parameters, statistical models and formulas see Dong and Maynard (2013), Dong, Kelcey, and Spybrook (in press), Spybrook, Kelcey, and Dong (in press), Kelcey, Dong, Spybrook, and Cox (2017), and Kelcey, Dong, Spybrook, and Shen (in press).

For reference intraclass correlation (`rho2, rho3`) values see Dong, Reinke, Herman, Bradshaw, and Murray (2016), Hedberg and Hedges (2014), Hedges and Hedberg (2007, 2013), Kelcey, and Phelps (2013), Schochet (2008), Spybrook, Westine, and Taylor (2016). For reference variance (`R12, R22, R32`) values see Bloom, Richburg-Hayes, and Black (2007), Deke et al. (2010), Dong et al. (2016), Hedges and Hedberg (2013), Kelcey, and Phelps (2013), Spybrook, Westine,and Taylor (2016), Westine, Spybrook, and Taylor (2013). Users can also obtain design parameters for various levels using publicly available state or district data.

References

Bloom, H. S., Richburg- Hayes, L. & Black, A. R. (2007). Using Covariates to Improve Precision for Studies that Randomize Schools to Evaluate Educational Interventions. *Educational Evaluation and Policy Analysis, 29(1)*, 0-59.

Deke, John, Dragoset, Lisa, and Moore, Ravaris (2010). Precision Gains from Publically Available School Proficiency Measures Compared to Study-Collected Test Scores in Education Cluster-Randomized Trials (NCEE 2010-4003). Washington, DC: National Center for Education Evaluation and Regional Assistance, Institute of Education Sciences, U.S. Department of Education. http://ies.ed.gov/ncee/pubs/20104003/

Dong, N., Kelcey, B., & Spybrook, J. (in press). Power analyses of moderator effects in threelevel cluster randomized trials. *Journal of Experimental Education*.

Dong, N., & Maynard, R. A. (2013). PowerUp!: A Tool for Calculating Minum Detectable Effect Sizes and Minimum Required Sample Sizes for Experimental and Quasi-Experimental Design Studies,*Journal of Research on Educational Effectiveness, 6(1)*, 24-6.

Dong, N., Reinke, W. M., Herman, K. C., Bradshaw, C. P., & Murray, D. W. (2016). Meaningful effect sizes, intraclass correlations, and proportions of variance explained by covariates for panning two-and three-level cluster randomized trials of social and behavioral outcomes. *Evaluation Review*. doi: 10.1177/0193841X16671283

Hedges, L. V., & Borenstein, M. (2014). Conditional Optimal Design in Three- and Four-Level Experiments. *Journal of Educational and Behavioral Statistics, 39(4)*, 257-281

Hedberg, E., & Hedges, L. V.(2014). Reference Values of Within-District Intraclass Correlations of Academic Achivement by District Characteristics: Results From a Meta-Analysis of District-Specified Values. *Evaluation Review, 38(6)*, 546-582.

Hedges, L. V., & Hedberg, E. (2007). Interclass correlation values for planning group-randomized trials in education. *Educational Evaluation and Policy Analysis, 29(1)*, 60-87.

Hedges, L. V., & Hedberg, E. (2013). Interclass Correlations and Covariate Outcome Correlations for Planning Two- and Three-Level Cluster-Randomized Experiments in Education. *Evaluation Review, 37(6)*, 445-489.

Kelcey, B., Dong, N., Spybrook, J., & Cox, K. (2017). Statistical Power for Causally Defined Indirect Effects in Group-Randomized Trials With Individual-Level Mediators. *Journal of Educational and Behavioral Statistics*, 1076998617695506.

Kelcey, B., Dong, N., Spybrook, J., & Shen, Z. (in press). Statistical power for causally-defined mediation in group-randomized studies. *Multivariate Behavioral Research*.

Kelcey, B., & Phelps, G. (2013). Strategies for improving power in school randomized studies of professional development. *Evaluation Review, 37(6)*, 520-554.

R Core Team (2017). R: A language and environment for statistical computin . R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org.

Raudenbush, S. W. (1997). Statistical analysis and optimal design for cluster randomized trials. *Psychological Methods, 2*, 173-185.

Raudenbush, S. W., & Liu, X. (2000). Statistical power and optimal design for multisite trials. *Psychological Methods, 5*, 199-213.

Schochet, P. Z. (2008). Statistical Power for Random Assignment Evaluations of Education Programs. *Journal of Educational and Behavioral Statistics, 33(1)*, 62-87

Spybrook, J., Kelcey, B., & Dong, N. (2016). Power for detecting treatment by moderator effects in two-and three-level cluster randomized trials. *Journal of Educational and Behavioral Statistics, 41*(6), 605-627.

Spybrook, J., Westine, C. D., & Taylor, J. A. (2016). Design Parameters for Impact Research in Science Education: A Multisite Anlaysis. *AERA Open, 2(1)*, 1-15.

Westine, C. D., Spybrook, J., & Taylor, J. A. (2013). An Empirical Investigation of Variance Design Parameters for Planning Cluster-Randomized Trials of Science Achievement. *Evaluation Review, 37(6)*, 490-519.

`mdes.bcra3f2` *MDES Calculator for Three-Level Fixed Effects Blocked Cluster Random Assignment Design, Treatment at Level-2*

Description

`mdes.bcra3f2` calculates minimum detectable effect size (MDES) for designs with three-levels where level-2 units are randomly assigned to treatment and control groups within level-3 units (fixed blocks).

Usage

```
mdes.bcra3f2(power=.80, alpha=.05, two.tail=TRUE,
             rho2, P=.50, g2=0, R12=0, R22=0,
             n, J, K, ...)
```

Arguments

`power`	statistical power $(1 - \beta)$.
`alpha`	probability of type I error.
`two.tail`	logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing.
`rho2`	proportion of variance in the outcome explained by level-2 units.
`P`	average proportion of level-2 units randomly assigned to treatment within level-3 units.
`g2`	number of covariates at level-2.
`R12`	proportion of level-1 variance in the outcome explained by level-1 covariates.
`R22`	proportion of level-2 variance in the outcome explained by level-2 covariates.
`n`	harmonic mean of level-1 units across level-2 units (or simple average).
`J`	harmonic mean of level-2 units across level-3 units (or simple average).
`K`	level-3 sample size.
`...`	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

`fun`	function name.
`parms`	list of parameters used in MDES calculation.
`ncp`	noncentrality parameter.
`mdes`	minimum detectable effect size and $(1-\alpha)\%$ lower and upper confidence limits.

See Also

`power.bcra3f2`, `optimal.bcra3f2`

Examples

```
## Not run:

 mdes.bcra3f2(rho2=.10, n=20,
             J=44, K=5)


## End(Not run)
```

`mdes.bcra3r2` *MDES Calculator for Three-Level Random Effects Blocked Cluster Random Assignment Designs, Treatment at Level-2*

Description

`mdes.bcra3r2` calculates minimum detectable effect size (MDES) for designs with three-levels where level-2 units are randomly assigned to treatment and control groups within level-3 units (random blocks).

Usage

```
mdes.bcra3r2(power=.80, alpha=.05, two.tail=TRUE,
             rho2, rho3, omega3,
             P=.50, g3=0, R12=0, R22=0, RT32=0,
             n, J, K, ...)
```

Arguments

`power`	statistical power $(1 - \beta)$.
`alpha`	probability of type I error.
`two.tail`	logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing.
`rho2`	proportion of variance in the outcome explained by level-2 units.
`rho3`	proportion of variance in the outcome explained by level-3 units.
`omega3`	treatment effect heterogeneity as ratio of treatment effect variance among level-3 units to the residual variance at level-3.
`P`	average proportion of level-2 units randomly assigned to treatment within level-3 units.
`g3`	number of covariates at level-3.
`R12`	proportion of level-1 variance in the outcome explained by level-1 covariates.
`R22`	proportion of level-2 variance in the outcome explained by level-2 covariates.
`RT32`	proportion of treatment effect variance among level-3 units explained by level-3 covariates.
`n`	harmonic mean of level-1 units across level-2 units (or simple average).
`J`	harmonic mean of level-2 units across level-3 units (or simple average).
`K`	level-3 sample size.
`...`	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in MDES calculation.
ncp	noncentrality parameter.
mdes	minimum detectable effect size and $(1-\alpha)\%$ lower and upper confidence limits.

See Also

power.bcra3r2, optimal.bcra3r2

Examples

```
## Not run:

 mdes.bcra3r2(rho3=.13, rho2=.10, omega3=.40,
             n=10, J=6, K=24)


## End(Not run)
```

---

mdes.bcra4f3 *MDES Calculator for Four-Level Fixed Effects Blocked Cluster Random Assignment Designs, Treatment at Level-3*

Description

mdes.bcra4f3 calculates minimum detectable effect size (MDES) for designs with four-levels where level-3 units are randomly assigned to treatment and control groups within level-4 units (fixed blocks).

Usage

```
mdes.bcra4f3(power=.80, alpha=.05, two.tail=TRUE,
     rho2, rho3,
     P=.50, R12=0, R22=0, R32=0, g3=0,
     n, J, K, L, ...)
```

Arguments

power	statistical power $(1-\beta)$.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
P	average proportion of level-3 units randomly assigned to treatment within level-4 units.
g3	number of covariates at level-3.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.

R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
R32	proportion of level-3 variance in the outcome explained by level-3 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	harmonic mean of level-3 units across level-4 units (or simple average).
L	number of level-4 units.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in MDES calculation.
ncp	noncentrality parameter.
mdes	minimum detectable effect size and $(1-\alpha)\%$ lower and upper confidence limits.

See Also

power.bcra4f3, optimal.bcra4f3

Examples

```
## Not run:

  mdes.bcra4f3(rho3=.15, rho2=.15,
               n=10, J=4, K=4, L=15)

## End(Not run)
```

---

mdes.bcra4r2 *MDES Calculator for Four-Level Random Effects Block Random Assignment Designs, Treatment at Level-2*

Description

mdes.bcra4r2 calculates minimum detectable effect size (MDES) for designs with four-levels where level-2 units are randomly assigned to treatment and control groups within level-3 units (random blocks).

Usage

```
mdes.bcra4r2(power=.80, alpha=.05, two.tail=TRUE,
     rho2, rho3, rho4, omega3, omega4,
     P=.50, R12=0, R22=0, RT32=0, RT42=0, g4=0,
     n, J, K, L, ...)
```

Arguments

power	statistical power $(1 - \beta)$.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
rho4	proportion of variance in the outcome explained by level-4 units.
omega3	treatment effect heterogeneity as ratio of treatment effect variance among level-3 units to the residual variance at level-3.
omega4	treatment effect heterogeneity as ratio of treatment effect variance among level-4 units to the residual variance at level-4.
P	average proportion of level-2 units randomly assigned to treatment within level-3 units.
g4	number of covariates at level-4.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
RT32	proportion of treatment effect variance among level-3 units explained by level-3 covariates.
RT42	proportion of treatment effect variance among level-4 units explained by level-4 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	harmonic mean of level-3 units across level-4 units (or simple average).
L	number of level-4 units.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in MDES calculation.
ncp	noncentrality parameter.
mdes	minimum detectable effect size and $(1-\alpha)\%$ lower and upper confidence limits.

See Also

power.bcra4r2, optimal.bcra4r2

Examples

```
## Not run:

  mdes.bcra4r2(rho4=.05, rho3=.15, rho2=.15,
               omega4=.50, omega3=.50,
               n=10, J=4, L=27, K=4)

## End(Not run)
```

mdes.bcra4r3 *MDES Calculator for Four-Level Random Effects Blocked Cluster Random Assignment Designs, Treatment at Level-3*

Description

mdes.bcra4r3 calculates minimum detectable effect size (MDES) for designs with four-levels where level-3 units are randomly assigned to treatment and control groups within level-4 units (random blocks).

Usage

```
mdes.bcra4r3(power=.80, alpha=.05, two.tail=TRUE,
     rho2, rho3, rho4, omega4,
     P=.50, R12=0, R22=0, R32=0, RT42=0, g4=0,
     n, J, K, L, ...)
```

Arguments

power	statistical power $(1 - \beta)$.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
rho4	proportion of variance in the outcome explained by level-4 units.
omega4	treatment effect heterogeneity as ratio of treatment effect variance among level-4 units to the residual variance at level-4.
P	average proportion of level-3 units randomly assigned to treatment within level-4 units.
g4	number of covariates at level-4.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
R32	proportion of level-3 variance in the outcome explained by level-3 covariates.
RT42	proportion of treatment effect variance among level-4 units explained by level-4 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	harmonic mean of level-3 units across level-4 units (or simple average).
L	number of level-4 units.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in MDES calculation.
ncp	noncentrality parameter.
mdes	minimum detectable effect size and 95% lower and upper confidence limits.

See Also

power.bcra4r3, optimal.bcra4r3

Examples

```
## Not run:

  mdes.bcra4r3(rho4=.05, rho3=.15, rho2=.15,
              omega4=.50,
              n=10, J=4, L=27, K=4)

## End(Not run)
```

---

mdes.bira2c1 *MDES Calculator for Two-Level Constant Effects Blocked Individual Random Assignment Designs, Treatment at Level-1*

Description

mdes.bira2c1 calculates minimum detectable effect size (MDES) for designs with two-levels where level-1 units are randomly assigned to treatment and control groups within level-2 units (school intercepts only).

Usage

```
  mdes.bira2c1(power=.80, alpha=.05, two.tail=TRUE,
              P=.50, g1=0, R12=0,
              n, J, ...)
```

Arguments

power	statistical power $(1 - \beta)$.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
P	average proportion of level-1 units randomly assigned to treatment within level-2 units.
g1	number of covariates at level-1.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	level-2 sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in MDES calculation.
ncp	noncentrality parameter.
mdes	minimum detectable effect size and $(1-\alpha)\%$ lower and upper confidence limits.

See Also

power.bira2c1, optimal.bira2c1

Examples

```
## Not run:

 mdes.bira2c1(n=55, J=3)


## End(Not run)
```

---

mdes.bira2f1 *MDES Calculator for Two-Level Fixed Effects Blocked Individual Random Assignment Designs, Treatment at Level-1*

Description

mdes.bira2f1 calculates minimum detectable effect size (MDES) for designs with two-levels where level-1 units are randomly assigned to treatment and control groups within level-2 units (fixed blocks).

Usage

```
mdes.bira2f1(power=.80, alpha=.05, two.tail=TRUE,
             P=.50, g1=0, R12=0,
             n, J, ...)
```

Arguments

power	statistical power $(1-\beta)$.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
P	average proportion of level-1 units randomly assigned to treatment within level-2 units.
g1	number of covariates at level-1.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	level-2 sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in MDES calculation.
ncp	noncentrality parameter.
mdes	minimum detectable effect size and $(1-\alpha)\%$ lower and upper confidence limits.

See Also

power.bira2f1, optimal.bira2f1

Examples

```
## Not run:

 mdes.bira2f1(n=55, J=3)


## End(Not run)
```

---

mdes.bira2r1 *MDES Calculator for Two-Level Random Effects Blocked Individual Random Assignment Designs, Individuals Randomized within Blocks*

Description

mdes.bira2r1 calculates minimum detectable effect size (MDES) for designs with two-levels where level-1 units are randomly assigned to treatment and control groups within level-2 units (random blocks).

Usage

```
mdes.bira2r1(power=.80, alpha=.05, two.tail=TRUE,
             rho2, omega2,
             P=.50, g2=0, R12=0, RT22=0,
             n, J, ...)
```

Arguments

power	statistical power $(1-\beta)$.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
omega2	treatment effect heterogeneity as ratio of treatment effect variance among level-2 units to the residual variance at level-2.
P	average proportion of level-1 units randomly assigned to treatment within level-2 units.
g2	number of covariates at level-2.

R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
RT22	proportion of treatment effect variance among level-2 units explained by level-2 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	level-2 sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in MDES calculation.
ncp	noncentrality parameter.
mdes	minimum detectable effect size and $(1-\alpha)\%$ lower and upper confidence limits.

See Also

power.bira2r1, optimal.bira2r1

Examples

```
## Not run:

 mdes.bira2r1(rho2=.35, omega2=.10,
             n=83, J=480)


## End(Not run)
```

---

mdes.bira3r1 *MDES Calculator for Three-Level Random Effects Blocked Individual Random Assignment Design, Individuals Randomized within Blocks*

Description

mdes.bira3r1 calculates minimum detectable effect size (MDES) for designs with three-levels where level-1 units are randomly assigned to treatment and control groups within level-2 units (random blocks).

Usage

```
mdes.bira3r1(power=.80, alpha=.05, two.tail=TRUE,
             rho2, rho3, omega2, omega3,
             P=.50, R12=0, RT22=0, RT32=0, g3=0,
             n, J, K, ...)
```

Arguments

power	statistical power $(1 - \beta)$.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
omega2	treatment effect heterogeneity as ratio of treatment effect variance among level-2 units to the residual variance at level-2.
omega3	treatment effect heterogeneity as ratio of treatment effect variance among level-3 units to the residual variance at level-3.
P	average proportion of level-1 units randomly assigned to treatment within level-2 units.
g3	number of covariates at level-3.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
RT22	proportion of treatment effect variance among level-2 units explained by level-2 covariates.
RT32	proportion of treatment effect variance among level-3 units explained by level-3 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	level-3 sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in MDES calculation.
ncp	noncentrality parameter.
mdes	minimum detectable effect size and $(1-\alpha)\%$ lower and upper confidence limits.

See Also

power.bira3r1, optimal.bira3r1

Examples

```
## Not run:

 mdes.bira3r1(rho3=.20, rho2=.15, omega3=.10, omega2=.10,
              n=69, J=10, K=100)

## End(Not run)
```

`mdes.bira4r1` *MDES Calculator for Four-Level Random Effects Blocked Individual Random Assignment Design, Individuals Randomized within Blocks*

Description

`mdes.bira4r1` calculates minimum detectable effect size (MDES) for designs with four-levels where level-1 units are randomly assigned to treatment and control groups within level-2 units (random blocks).

Usage

```
mdes.bira4r1(power=.80, alpha=.05, two.tail=TRUE,
     rho2, rho3, rho4, omega2, omega3, omega4,
     P=.50, R12=0, RT22=0, RT32=0, RT42=0, g4=0,
     n, J, K, L, ...)
```

Arguments

`power`	statistical power $(1 - \beta)$.
`alpha`	probability of type I error.
`two.tail`	logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing.
`rho2`	proportion of variance in the outcome explained by level-2 units.
`rho3`	proportion of variance in the outcome explained by level-3 units.
`rho4`	proportion of variance in the outcome explained by level-4 units.
`omega2`	treatment effect heterogeneity as ratio of treatment effect variance among level-2 units to the residual variance at level-2.
`omega3`	treatment effect heterogeneity as ratio of treatment effect variance among level-3 units to the residual variance at level-3.
`omega4`	treatment effect heterogeneity as ratio of treatment effect variance among level-4 units to the residual variance at level-4.
`P`	average proportion of level-1 units randomly assigned to treatment within level-2 units.
`g4`	number of covariates at level-4.
`R12`	proportion of level-1 variance in the outcome explained by level-1 covariates.
`RT22`	proportion of treatment effect variance among level-2 units explained by level-2 covariates.
`RT32`	proportion of treatment effect variance among level-3 units explained by level-3 covariates.
`RT42`	proportion of treatment effect variance among level-4 units explained by level-4 covariates.
`n`	harmonic mean of level-1 units across level-2 units (or simple average).
`J`	harmonic mean of level-2 units across level-3 units (or simple average).
`K`	harmonic mean of level-3 units across level-4 units (or simple average).
`L`	number of level-4 units.
`...`	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in MDES calculation.
ncp	noncentrality parameter.
mdes	minimum detectable effect size and $(1-\alpha)\%$ lower and upper confidence limits.

See Also

power.bira4r1, optimal.bira4r1

Examples

```
## Not run:

  mdes.bira4r1(rho4=.05, rho3=.15, rho2=.15,
              omega4=.50, omega3=.50, omega2=.50,
              n=10, J=4, L=27, K=4)

## End(Not run)
```

---

mdes.cra2r2 *MDES Calculator for Main Effect in Two-Level Cluster Random Assignment Design, Treatment at Level-2*

Description

mdes.cra2r2 calculates minimum detectable effect size (MDES) for main effect in designs with two-levels where level-2 units are randomly assigned to treatment and control groups.

Usage

```
mdes.cra2r2(power=.80, alpha=.05, two.tail=TRUE,
                  rho2,
                  P=.50,  g2=0, R12=0, R22=0,
                  n, J, ...)
```

Arguments

power	statistical power $(1-\beta)$.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
P	proportion of level-2 units randomly assigned to treatment.
g2	number of covariates at level-2.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	level-2 sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in MDES calculation.
ncp	noncentrality parameter.
mdes	minimum detectable effect size and $(1-\alpha)\%$ lower and upper confidence limits.

See Also

power.cra2r2, optimal.cra2r2

Examples

```
## Not run:

 mdes.cra2r2(rho2=.20,
                  n=4, J=20)


## End(Not run)
```

---

mdes.cra3r3 *MDES Calculator for Main Effect in Three-Level Cluster Random Assignment Designs, Treatment at Level-3*

Description

mdes.cra3r3 calculates minimum detectable effect size (MDES) for main effect in designs with three-levels where level-3 units are randomly assigned to treatment and control groups.

Usage

```
 mdes.cra3r3(power=.80, alpha=.05, two.tail=TRUE,
                  rho2, rho3, P=.50, g3=0, R12=0, R22=0, R32=0,
                  n, J, K, ...)
```

Arguments

power	statistical power $(1-\beta)$.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
P	proportion of level-3 units randomly assigned to treatment.
g3	number of covariates at level-3.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
R32	proportion of level-3 variance in the outcome explained by level-3 covariates.

n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	level-3 sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in MDES calculation.
ncp	noncentrality parameter.
mdes	minimum detectable effect size and $(1-\alpha)\%$ lower and upper confidence limits.

See Also

power.cra3r3, optimal.cra3r3

Examples

```
## Not run:

 mdes.cra3r3(rho3=.13, rho2=.10, omega3=.40,
                   n=10, J=6, K=24)


## End(Not run)
```

---

mdes.cra4r4 *MDES Calculator for Four-Level Cluster Random Assignment Designs, Treatment at Level-4*

Description

mdes.cra4r4 calculates minimum detectable effect size (MDES) for designs with four-levels where level-4 units are randomly assigned to treatment and control groups.

Usage

```
mdes.cra4r4(power=.80, alpha=.05, two.tail=TRUE,
     rho2, rho3, rho4,
     P=.50, R12=0, R22=0, R32=0, R42=0, g4=0,
     n, J, K, L, ...)
```

Arguments

power	statistical power $(1-\beta)$.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.

rho3	proportion of variance in the outcome explained by level-3 units.
rho4	proportion of variance in the outcome explained by level-4 units.
P	proportion of level-4 units randomly assigned to treatment.
g4	number of covariates at level-4.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
R32	proportion of level-3 variance in the outcome explained by level-3 covariates.
R42	proportion of level-4 variance in the outcome explained by level-4 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	harmonic mean of level-3 units across level-4 units (or simple average).
L	number of level-4 units.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in MDES calculation.
ncp	noncentrality parameter.
mdes	minimum detectable effect size and $(1-\alpha)\%$ lower and upper confidence limits.

See Also

power.cra4r4, optimal.cra4r4

Examples

```
## Not run:

  mdes.cra4r4(rho4=.05, rho3=.05, rho2=.10,
              n=10, J=2, K=3, L=20)

## End(Not run)
```

---

mdes.ira1r1 *MDES Calculator for Individual Random Assignment Designs, Completely Randomized Controlled Trials*

Description

mdes.ira1r1 calculates minimum detectable effect size (MDES) for completely randomized controlled trials where individuals are randomly assigned to treatment or control groups.

Usage

```
mdes.ira1r1(power=.80, alpha=.05, two.tail=TRUE,
            P=.50, g1=0, R12=0,
            n, ...)
```

Arguments

power	statistical power $(1 - \beta)$.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
P	proportion of units randomly assigned to treatment.
g1	number of covariates.
R12	proportion of variance in the outcome explained by covariates.
n	sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in MDES calculation.
ncp	noncentrality parameter.
mdes	minimum detectable effect size and $(1-\alpha)\%$ lower and upper confidence limits.

See Also

power.ira1r1, optimal.ira1r1

Examples

```
## Not run:

 mdes.ira1r1(n=55)


## End(Not run)
```

---

mdes.mod1n.cra2r2 *MDES Calculator for Non-Randomly Varying Moderator Effect in Two-Level Cluster Random Assignment Design, Treatment at Level-2, Moderator at Level-1*

Description

mdes.mod1n.cra2r2 calculates minimum detectable effect size (MDES) for non-randomly varying moderator effect in designs with two-levels where level-2 units are randomly assigned to treatment and control groups and moderator is at level-1.

Usage

```
mdes.mod1n.cra2r2(power=.80, alpha=.05, two.tail=TRUE,
                  rho2, P=.50, Q=NULL, g1=0, R12=0,
                  n, J, ...)
```

Arguments

power	statistical power $(1 - \beta)$.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
P	proportion of level-2 units randomly assigned to treatment.
Q	proportion of level-1 units within moderator subgroup. NULL implies continuous moderator.
g1	number of covariates at level-1 excluding moderator.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	level-2 sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in MDES calculation.
ncp	noncentrality parameter.
mdes	minimum detectable effect size and $(1-\alpha)\%$ lower and upper confidence limits.

See Also

power.mod1n.cra2r2, optimal.mod1n.cra2r2

Examples

```
## Not run:

 mdes.mod1n.cra2r2(rho2=.20,
                   n=4, J=20)


## End(Not run)
```

---

mdes.mod1n.cra3r3 *MDES Calculator for Non-Randomly Varying Moderator Effect in Three-Level Cluster Random Assignment Design, Treatment at Level-3, Moderator at Level-1*

Description

mdes.mod1n.cra3r3 calculates minimum detectable effect size (MDES) for non-randomly varying moderator effect in designs with three-levels where level-3 units are randomly assigned to treatment and control groups and moderator is at level-1.

Usage

```
mdes.mod1n.cra3r3(power=.80, alpha=.05, two.tail=TRUE,
                  rho2, rho3, P=.50, Q=NULL, g1=0, R12=0,
                  n, J, K, ...)
```

Arguments

power	statistical power $(1 - \beta)$.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
P	proportion of level-3 units randomly assigned to treatment.
Q	proportion of level-1 units within moderator subgroup. NULL implies continuous moderator.
g1	number of covariates at level-1 excluding moderator.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	level-3 sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in MDES calculation.
ncp	noncentrality parameter.
mdes	minimum detectable effect size and $(1-\alpha)\%$ lower and upper confidence limits.

See Also

power.mod1n.cra3r3, optimal.mod1n.cra3r3

Examples

```
## Not run:

 mdes.mod1n.cra3r3(rho3=.13, rho2=.10, omega3=.40,
                   n=10, J=6, K=24)


## End(Not run)
```

`mdes.mod1r.cra2r2` *MDES Calculator for Randomly Varying Moderator Effect in Two-Level Cluster Random Assignment Design, Treatment at Level-2, Moderator at Level-1*

Description

`mdes.mod1r.cra2r2` calculates minimum detectable effect size (MDES) for randomly varying moderator effect in designs with two-levels where level-2 units are randomly assigned to treatment and control groups and moderator is at level-1.

Usage

```
mdes.mod1r.cra2r2(power=.80, alpha=.05, two.tail=TRUE,
                  rho2, omega2, RT22=0, P=.50, Q=NULL, g1=0, R12=0,
                  n, J, ...)
```

Arguments

`power`	statistical power $(1 - \beta)$.
`alpha`	probability of type I error.
`two.tail`	logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing.
`rho2`	proportion of variance in the outcome explained by level-2 units.
`omega2`	moderator effect heterogeneity (unconditional) across level-2 units.
`P`	proportion of level-2 units randomly assigned to treatment.
`Q`	proportion of level-1 units within moderator subgroup. `NULL` implies continuous moderator.
`g1`	number of covariates at level-1 excluding moderator.
`R12`	proportion of level-1 variance in the outcome explained by level-1 covariates.
`RT22`	proportion of level-2 variance in the moderator effect explained by level-2 covariates.
`n`	harmonic mean of level-1 units across level-2 units (or simple average).
`J`	level-2 sample size.
`...`	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

`fun`	function name.
`parms`	list of parameters used in MDES calculation.
`ncp`	noncentrality parameter.
`mdes`	minimum detectable effect size and $(1-\alpha)\%$ lower and upper confidence limits.

See Also

`power.mod1r.cra2r2`, `optimal.mod1r.cra2r2`

Examples

```
## Not run:

 mdes.mod1r.cra2r2(rho2=.2, omega2=.2, RT22=.2, n=4, J=20)


## End(Not run)
```

`mdes.mod1r.cra3r3` *MDES Calculator for Randomly Varying Moderator Effect in Three-Level Cluster Random Assignment Designs, Treatment at Level-3, Moderator at Level-1*

Description

`mdes.mod1r.cra3r3` calculates minimum detectable effect size (MDES) for randomly varying moderator effect in designs with three-levels where level-3 units are randomly assigned to treatment and control groups and moderator is at level-1.

Usage

```
 mdes.mod1r.cra3r3(power=.80, alpha=.05, two.tail=TRUE,
                      rho2, rho3, omega2, omega3, P=.50, Q=NULL,
                      g1=0, R12=0, RT22=0, RT32=0,
                      n, J, K, ...)
```

Arguments

`power`	statistical power $(1 - \beta)$.
`alpha`	probability of type I error.
`two.tail`	logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing.
`rho2`	proportion of variance in the outcome explained by level-2 units.
`rho3`	proportion of variance in the outcome explained by level-3 units.
`omega2`	moderator effect heterogeneity (unconditional) across level-2 units.
`omega3`	moderator effect heterogeneity (unconditional) across level-3 units.
`P`	proportion of level-3 units randomly assigned to treatment.
`Q`	proportion of level-1 units within moderator subgroup. `NULL` implies continuous moderator.
`g1`	number of covariates at level-1 excluding moderator.
`R12`	proportion of level-1 variance in the outcome explained by level-1 covariates.
`RT22`	proportion of level-2 variance in the moderator effect explained by level-2 covariates.
`RT32`	proportion of level-3 variance in the moderator effect explained by level-3 covariates.
`n`	harmonic mean of level-1 units across level-2 units (or simple average).
`J`	harmonic mean of level-2 units across level-3 units (or simple average).
`K`	level-3 sample size.
`...`	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in MDES calculation.
ncp	noncentrality parameter.
mdes	minimum detectable effect size and $(1-\alpha)\%$ lower and upper confidence limits.

See Also

power.mod1r.cra3r3, optimal.mod1r.cra3r3

Examples

```
## Not run:

  mdes.mod1r.cra3r3(rho3=.05, rho2=.12, omega2=.08, omega3=.07,
                    P=.4, Q=.7, g1=1, R12=.20, RT22=0, RT32=0,
                    n=20, J=4, K=60)


## End(Not run)
```

---

mdes.mod2.cra2r2 *MDES Calculator for Moderator Effect in Two-Level Cluster Random Assignment Design, Treatment and Moderator at Level-2*

Description

mdes.mod2.cra2r2 calculates minimum detectable effect size (MDES) for moderator effect in designs with two-levels where level-2 units are randomly assigned to treatment and control groups and moderator is at level-2.

Usage

```
  mdes.mod2.cra2r2(power=.80, alpha=.05, two.tail=TRUE,
                   rho2, P=.50, Q=NULL, g2=0, R12=0, R22=0,
                   n, J, ...)
```

Arguments

power	statistical power $(1-\beta)$.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
P	proportion of level-2 units randomly assigned to treatment.
Q	proportion of level-1 units within moderator subgroup. NULL implies continuous moderator.
g2	number of covariates at level-2 excluding moderator.

R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	level-2 sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in MDES calculation.
ncp	noncentrality parameter.
mdes	minimum detectable effect size and $(1-\alpha)\%$ lower and upper confidence limits.

See Also

power.mod2.cra2r2, optimal.mod2.cra2r2

Examples

```
## Not run:

 mdes.mod2.cra2r2(rho2=.20, n=4, J=20, Q=.5)


## End(Not run)
```

---

mdes.mod2n.cra3r3 *MDES Calculator for Non-Randomly Varying Moderator Effect in Three-Level Cluster Random Assignment Design, Treatment at Level-3, Moderator at Level-2*

Description

mdes.mod2n.cra3r3 calculates minimum detectable effect size (MDES) for non-randomly varying moderator effect in designs with three-levels where level-3 units are randomly assigned to treatment and control groups and moderator is at level-2.

Usage

```
mdes.mod2n.cra3r3(power=.80, alpha=.05, two.tail=TRUE,
                  rho2, rho3, P=.50, Q=NULL, g2=0, R12=0, R22=0,
                  n, J, K, ...)
```

Arguments

power	statistical power $(1 - \beta)$.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
P	proportion of level-3 units randomly assigned to treatment.
Q	proportion of level-1 units within moderator subgroup. NULL implies continuous moderator.
g2	number of covariates at level-2 excluding moderator.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	level-3 sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in MDES calculation.
ncp	noncentrality parameter.
mdes	minimum detectable effect size and $(1-\alpha)\%$ lower and upper confidence limits.

See Also

power.mod2n.cra3r3, optimal.mod2n.cra3r3

Examples

```
## Not run:

  mdes.mod2n.cra3r3(rho3=.10, rho2=.10, Q=.5, g2=1,
                    R12=.30, R22=.40,  n=20, J=4, K=60)


## End(Not run)
```

`mdes.mod2r.cra3r3` *MDES Calculator for Randomly Varying Moderator Effect in Three-Level Cluster Random Assignment Designs, Treatment at Level-3, Moderator at Level-2*

Description

`mdes.mod2r.cra3r3` calculates minimum detectable effect size (MDES) for randomly varying moderator effect in designs with three-levels where level-3 units are randomly assigned to treatment and control groups and moderator is at level-2.

Usage

```
mdes.mod2r.cra3r3(power=.80, alpha=.05, two.tail=TRUE,
                  rho2, rho3, omega3, P=.50, Q=NULL, g2=0, R12=0, R22=0, RT32=0,
                    n, J, K, ...)
```

Arguments

`power`	statistical power $(1 - \beta)$.
`alpha`	probability of type I error.
`two.tail`	logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing.
`rho2`	proportion of variance in the outcome explained by level-2 units.
`rho3`	proportion of variance in the outcome explained by level-3 units.
`omega3`	moderator effect heterogeneity (unconditional) across level-3 units.
`P`	proportion of level-3 units randomly assigned to treatment.
`Q`	proportion of level-1 units within moderator subgroup. `NULL` implies continuous moderator.
`g2`	number of covariates at level-2 excluding moderator.
`R12`	proportion of level-1 variance in the outcome explained by level-1 covariates.
`R22`	proportion of level-2 variance in the outcome explained by level-2 covariates.
`RT32`	proportion of level-3 variance in the moderator effect explained by level-3 covariates.
`n`	harmonic mean of level-1 units across level-2 units (or simple average).
`J`	harmonic mean of level-2 units across level-3 units (or simple average).
`K`	level-3 sample size.
`...`	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

`fun`	function name.
`parms`	list of parameters used in MDES calculation.
`ncp`	noncentrality parameter.
`mdes`	minimum detectable effect size and $(1-\alpha)\%$ lower and upper confidence limits.

See Also

power.mod2r.cra3r3, optimal.mod2r.cra3r3

Examples

```
## Not run:

  mdes.mod2r.cra3r3(rho3=.10, rho2=.10, omega3=.05, Q=.5, g2=1,
                    R12=.30, R22=.40, RT32=0,  n=20, J=4, K=60)


## End(Not run)
```

---

mdes.mod3.cra3r3 *MDES Calculator for Moderator Effect in Three-Level Cluster Random Assignment Designs, Treatment and Moderator at Level-3*

Description

mdes.mod3.cra3r3 calculates minimum detectable effect size (MDES) for moderator effect in designs with three-levels where level-3 units are randomly assigned to treatment and control groups and moderator is at level-3.

Usage

```
mdes.mod3.cra3r3(power=.80, alpha=.05, two.tail=TRUE,
                   rho2, rho3, P=.50, Q=NULL, g3=0, R12=0, R22=0, R32=0,
                   n, J, K, ...)
```

Arguments

power	statistical power $(1 - \beta)$.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
P	proportion of level-3 units randomly assigned to treatment.
Q	proportion of level-1 units within moderator subgroup. NULL implies continuous moderator.
g3	number of covariates at level-3 excluding moderator.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
R32	proportion of level-3 variance in the outcome explained by level-3 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	level-3 sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in MDES calculation.
ncp	noncentrality parameter.
mdes	minimum detectable effect size and $(1-\alpha)\%$ lower and upper confidence limits.

See Also

power.mod3.cra3r3, optimal.mod3.cra3r3

Examples

```
## Not run:

  mdes.mod3.cra3r3(rho3=.1, rho2=.1, omega3=.05, Q=.5,
                   g3=1, R12=.3, R22=.4, R32=.5, n=20, J=4, K=60)


## End(Not run)
```

---

mdes.to.pctl *MDES to Percentiles*

Description

mdes.to.pctl converts MDES values returned from mdes function into percentile values.

Usage

```
mdes.to.pctl(x)
```

Arguments

x	an object returned from mdes function or a numeric value in standard deviation units.

Details

mdes.to.pctl converts MDES values returned from mdes function into percentile values. Values should be interpreted as percentile increase with reference to 50th percentile.

See Also

optimal.to.mdes

Examples

```
## Not run:

  # MDES values returned from mdes function
  design1 <- mdes.cra3r3(power=.80, rho2=.06, rho3=.18,
                    g3=1, R12=.55, R22=.50, R32=.45,
                    P=.40, n=10, J=2, K=83)
  # convert MDES values into percentiles
  percentiles <- mdes.to.pctl(design1)


## End(Not run)
```

`mdes.to.power` *MDES to power*

Description

`mdes.to.power` converts an object returned from `mdes` function into an object returned from `power` function.

Usage

```
mdes.to.power(x)
```

Arguments

`x`	an object returned from one of the `mdes` functions.

Details

`mdes.to.power` converts an object returned from `mdes` function into an object returned from `power` function by passing parameters through `power` function.

See Also

`optimal.to.mdes`, `power.to.mdes`

Examples

```
## Not run:

  # object returned from power function
  design1 <- mdes.bira2r1(rho2=.35, omega2=.10, n=83, J=10)
  # convert the object into an object returned from mdes function
  design2 <- mdes.to.power(design1)


## End(Not run)
```

`optimal.bcra3f2` *COSA Solver for Main Effect in Three-Level Fixed Effects Blocked Cluster Random Assignment Designs, Treatment at Level-2*

Description

`optimal.bcra3f2` finds constrained optimal sample allocation (COSA) solutions for main effect in designs with three-levels where level-2 units are randomly assigned to treatment and control groups within level-3 units (fixed blocks). COSA can be found in the following forms, (i) under budgetary constraints given marginal costs per unit, (ii) under power contraints given marginal costs per unit, (iii) under MDES contraints given marginal costs per unit, and (iv) under sample size contraints for one or more levels along with any of the i,ii, or iii options.

Usage

```
optimal.bcra3f2(cn=0, cJ=0, cK=0, cost=NULL, n=NULL, J=NULL, K=NULL,
                power=.80, mdes=.25, alpha=.05, two.tail=TRUE,
                nJK0=c(10,10,10), ncase=10, gm=2,
                constrain="power", optimizer="auglag_cobyla",
                rho2,
                P=.50, g2=0,  R12=0, R22=0)
```

Arguments

`cn`	marginal cost per level-1 unit.
`cJ`	marginal cost per level-2 unit.
`cK`	marginal cost per level-3 unit.
`cost`	total cost or budget.
`n`	harmonic mean of level-1 units across level-2 units (or simple average).
`J`	harmonic mean of level-2 units across level-3 units (or simple average).
`K`	level-3 sample size.
`power`	statistical power (1 - type II error).
`mdes`	minimum detectable effect size.
`alpha`	probability of type I error.
`two.tail`	logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing.
`nJK0`	vector with a length of three to specifiy starting values for level-1, level-2, and level-3 sample sizes.
`ncase`	number of cases to show in the output.
`gm`	grid multiplier to increase the range of sample size search for each level.
`constrain`	one of the followings can be constrained at a specified cost or value: `"cost"`, `"power"`, or `"mdes"`.
`optimizer`	algorithm to find optimal sample sizes given total cost, power, or MDES. Available algorithms: `"auglag_cobyla"`, `"auglag_lbfgs"`, `"auglag_mma"`, or `"auglag_slsqp"`.
`rho2`	proportion of variance in the outcome explained by level-2 units.

P	average proportion of level-2 units randomly assigned to treatment within level-3 units.
g2	number of covariates at level-2.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.

Value

fun	function name.
parms	list of parameters used in the function.
nloptr	list of nloptr log and output.
round.optim	solution after rounding. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).
integer.optim	best integer solutions around round.optim solution. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).

See Also

mdes.bcra3f2, power.bcra3f2

Examples

```
## Not run:

     optimal.bcra3f2(cn=1, cJ=10, cK=100, cost=5600,
                     constrain="cost",
                     rho2=.10)


## End(Not run)
```

---

optimal.bcra3r2 *COSA Solver for Main Effect in Three-Level Random Effects Blocked Cluster Random Assignment Designs, Treatment at Level-2*

Description

optimal.bcra3r2 finds constrained optimal sample allocation (COSA) solutions for main effect in designs with three-levels where level-2 units are randomly assigned to treatment and control groups within level-3 units (random blocks). COSA can be found in the following forms, (i) under budgetary constraints given marginal costs per unit, (ii) under power contraints given marginal costs per unit, (iii) under MDES contraints given marginal costs per unit, and (iv) under sample size contraints for one or more levels along with any of the i,ii, or iii options.

Usage

```
optimal.bcra3r2(cn=0, cJ=0, cK=0, cost=NULL, n=NULL, J=NULL, K=NULL,
                power=.80, mdes=.25, alpha=.05, two.tail=TRUE,
                nJK0=c(10,10,10), ncase=10, gm=2,
                constrain="power", optimizer="auglag_cobyla",
                rho2, rho3, omega3,
                P=.50, g3=0, R12=0, R22=0, RT32=0)
```

Arguments

cn	marginal cost per level-1 unit.
cJ	marginal cost per level-2 unit.
cK	marginal cost per level-3 unit.
cost	total cost or budget.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	level-3 sample size.
power	statistical power (1 - type II error).
mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
nJK0	vector with a length of three to specifiy starting values for level-1, level-2, and level-3 sample sizes.
ncase	number of cases to show in the output.
gm	grid multiplier to increase the range of sample size search for each level.
constrain	one of the followings can be constrained at a specified cost or value: "cost", "power", or "mdes".
optimizer	algorithm to find optimal sample sizes given total cost, power, or MDES. Available algorithms: "auglag_cobyla", "auglag_lbfgs", "auglag_mma", or "auglag_slsqp".
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
omega3	treatment effect heterogeneity as ratio of treatment effect variance among level-3 units to the residual variance at level-3.
P	average proportion of level-2 units randomly assigned to treatment within level-3 units.
g3	number of covariates at level-3.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
RT32	proportion of treatment effect variance among level-3 units explained by level-3 covariates.

Value

fun	function name.
parms	list of parameters used in the function.
nloptr	list of nloptr log and output.
round.optim	solution after rounding. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).
integer.optim	best integer solutions around round.optim solution. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).

See Also

mdes.bcra3r2, power.bcra3r2

Examples

```
## Not run:

    optimal.bcra3r2(cn=1, cJ=10, cK=100, cost=5600,
                    constrain="cost",
                    rho3=.13, rho2=.10, omega3=.40)


## End(Not run)
```

---

optimal.bcra4f3 *COSA Solver for Main Effect in Four-Level Fixed Effects Blocked Cluster Random Assignment Designs, Treatment at Level-3*

Description

optimal.bcra4f3 finds constrained optimal sample allocation (COSA) solutions for main effect in designs with four-levels where level-3 units are randomly assigned to treatment and control groups within level-4 units (fixed blocks). COSA can be found in the following forms, (i) under budgetary constraints given marginal costs per unit, (ii) under power contraints given marginal costs per unit, (iii) under MDES contraints given marginal costs per unit, and (iv) under sample size contraints for one or more levels along with any of the i,ii, or iii options.

Usage

```
optimal.bcra4f3(cn=0, cJ=0, cK=0, cL=0, cost=NULL,
                n=NULL, J=NULL, K=NULL, L=NULL,
                power=.80, mdes=.25, alpha=.05, two.tail=TRUE,
                nJKL0=c(10,10,10,10), ncase=10, gm=2,
                constrain="power", optimizer="auglag_cobyla",
                rho3, rho2,
                P=.50, g3=0, R32=0, R22=0, R12=0)
```

Arguments

cn	marginal cost per level-1 unit.
cJ	marginal cost per level-2 unit.
cK	marginal cost per level-3 unit.
cL	marginal cost per level-4 unit.
cost	total cost or budget.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	harmonic mean of level-3 units across level-4 units (or simple average).
L	number of level-4 units.
power	statistical power (1 - type II error).

mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
nJKL0	vector with a length of four to specifiy starting values for level-1, level-2, level-3, and level-4 sample sizes.
ncase	number of cases to show in the output.
gm	grid multiplier to increase the range of sample size search for each level.
constrain	one of the followings can be constrained at a specified cost or value: "cost", "power", or "mdes".
optimizer	algorithm to find optimal sample sizes given total cost, power, or MDES. Available algorithms: "auglag_cobyla", "auglag_lbfgs", "auglag_mma", or "auglag_slsqp".
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
P	average proportion of level-3 units randomly assigned to treatment within level-4 units.
g3	number of covariates at level-3.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
R32	proportion of level-3 variance in the outcome explained by level-3 covariates.

Value

fun	function name.
parms	list of parameters used in the function.
nloptr	list of nloptr log and output.
round.optim	solution after rounding. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).
integer.optim	best integer solutions around round.optim solution. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).

See Also

mdes.bcra4f3, power.bcra4f3

Examples

```
## Not run:

     optimal.bcra4f3(cn=1, cJ=10, cK=100, cL=1000, cost=75600,
                     constrain="cost",
                     rho3=.15, rho2=.20)


## End(Not run)
```

`optimal.bcra4r2` *COSA Solver for Main Effect in Four-Level Random Effects Block Random Assignment Designs, Treatment at Level-2*

Description

`optimal.bcra4r2` finds constrained optimal sample allocation (COSA) solutions for main effect in designs with four-levels where level-2 units are randomly assigned to treatment and control groups within level-3 units (random blocks). COSA can be found in the following forms, (i) under budgetary constraints given marginal costs per unit, (ii) under power contraints given marginal costs per unit, (iii) under MDES contraints given marginal costs per unit, and (iv) under sample size contraints for one or more levels along with any of the i,ii, or iii options.

Usage

```
optimal.bcra4r2(cn=0, cJ=0, cK=0, cL=0, cost=NULL,
                n=NULL, J=NULL, K=NULL, L=NULL,
                power=.80, mdes=.25, alpha=.05, two.tail=TRUE,
                nJKL0=c(10,10,10,10), ncase=10, gm=2,
                constrain="power", optimizer="auglag_cobyla",
                rho4, rho3, rho2, omega4, omega3,
                P=.50, g4=0, RT42=0, RT32=0, R22=0, R12=0)
```

Arguments

`cn`	marginal cost per level-1 unit.
`cJ`	marginal cost per level-2 unit.
`cK`	marginal cost per level-3 unit.
`cL`	marginal cost per level-4 unit.
`cost`	total cost or budget.
`n`	harmonic mean of level-1 units across level-2 units (or simple average).
`J`	harmonic mean of level-2 units across level-3 units (or simple average).
`K`	harmonic mean of level-3 units across level-4 units (or simple average).
`L`	number of level-4 units.
`power`	statistical power (1 - type II error).
`mdes`	minimum detectable effect size.
`alpha`	probability of type I error.
`two.tail`	logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing.
`nJKL0`	vector with a length of four to specifiy starting values for level-1, level-2, level-3, and level-4 sample sizes.
`ncase`	number of cases to show in the output.
`gm`	grid multiplier to increase the range of sample size search for each level.
`constrain`	one of the followings can be constrained at a specified cost or value: `"cost"`, `"power"`, or `"mdes"`.

optimizer	algorithm to find optimal sample sizes given total cost, power, or MDES. Available algorithms: "auglag_cobyla", "auglag_lbfgs", "auglag_mma", or "auglag_slsqp".
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
rho4	proportion of variance in the outcome explained by level-4 units.
omega3	treatment effect heterogeneity as ratio of treatment effect variance among level-3 units to the residual variance at level-3.
omega4	treatment effect heterogeneity as ratio of treatment effect variance among level-4 units to the residual variance at level-4.
P	average proportion of level-2 units randomly assigned to treatment within level-3 units.
g4	number of covariates at level-4.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
RT32	proportion of treatment effect variance among level-3 units explained by level-3 covariates.
RT42	proportion of treatment effect variance among level-4 units explained by level-4 covariates.

Value

fun	function name.
parms	list of parameters used in the function.
nloptr	list of nloptr log and output.
round.optim	solution after rounding. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).
integer.optim	best integer solutions around round.optim solution. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).

See Also

mdes.bcra4r2, power.bcra4r2

Examples

```
## Not run:

     optimal.bcra4r2(cn=1, cJ=10, cK=100, cL=1000, cost=75600,
                     constrain="cost",
                     rho4=.10, rho3=.15, rho2=.20,
                     omega4=.50, omega3=.50)


## End(Not run)
```

`optimal.bcra4r3` *COSA Solver for Main Effect in Four-Level Random Effects Blocked Cluster Random Assignment Designs, Treatment at Level-3*

Description

`optimal.bcra4r3` finds constrained optimal sample allocation (COSA) solutions for main effect in designs with four-levels where level-3 units are randomly assigned to treatment and control groups within level-4 units (random blocks). COSA can be found in the following forms, (i) under budgetary constraints given marginal costs per unit, (ii) under power contraints given marginal costs per unit, (iii) under MDES contraints given marginal costs per unit, and (iv) under sample size contraints for one or more levels along with any of the i,ii, or iii options.

Usage

```
optimal.bcra4r3(cn=0, cJ=0, cK=0, cL=0, cost=NULL,
                n=NULL, J=NULL, K=NULL, L=NULL,
                power=.80, mdes=.25, alpha=.05, two.tail=TRUE,
                nJKL0=c(10,10,10,10), ncase=10, gm=2,
                constrain="power", optimizer="auglag_cobyla",
                rho4, rho3, rho2, omega4,
                P=.50, g4=0, RT42=0, R32=0, R22=0, R12=0)
```

Arguments

`cn`	marginal cost per level-1 unit.
`cJ`	marginal cost per level-2 unit.
`cK`	marginal cost per level-3 unit.
`cL`	marginal cost per level-4 unit.
`cost`	total cost or budget.
`n`	harmonic mean of level-1 units across level-2 units (or simple average).
`J`	harmonic mean of level-2 units across level-3 units (or simple average).
`K`	harmonic mean of level-3 units across level-4 units (or simple average).
`L`	number of level-4 units.
`power`	statistical power (1 - type II error).
`mdes`	minimum detectable effect size.
`alpha`	probability of type I error.
`two.tail`	logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing.
`nJKL0`	vector with a length of four to specifiy starting values for level-1, level-2, level-3, and level-4 sample sizes.
`ncase`	number of cases to show in the output.
`gm`	grid multiplier to increase the range of sample size search for each level.
`constrain`	one of the followings can be constrained at a specified cost or value: `"cost"`, `"power"`, or `"mdes"`.

optimizer	algorithm to find optimal sample sizes given total cost, power, or MDES. Available algorithms: "auglag_cobyla", "auglag_lbfgs", "auglag_mma", or "auglag_slsqp".
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
rho4	proportion of variance in the outcome explained by level-4 units.
omega4	treatment effect heterogeneity as ratio of treatment effect variance among level-4 units to the residual variance at level-4.
P	average proportion of level-3 units randomly assigned to treatment within level-4 units.
g4	number of covariates at level-4.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
R32	proportion of level-3 variance in the outcome explained by level-3 covariates.
RT42	proportion of treatment effect variance among level-4 units explained by level-4 covariates.

Value

fun	function name.
parms	list of parameters used in the function.
nloptr	list of nloptr log and output.
round.optim	solution after rounding. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).
integer.optim	best integer solutions around round.optim solution. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).

See Also

mdes.bcra4r3, power.bcra4r3

Examples

```
## Not run:

     optimal.bcra4r3(cn=1, cJ=10, cK=100, cL=1000, cost=75600,
                     constrain="cost",
                     rho4=.10, rho3=.15, rho2=.20,
                     omega4=.50)


## End(Not run)
```

`optimal.bira2c1` *COSA Solver for Main Effect in Two-Level Constant Effects Blocked Individual Random Assignment Designs, Treatment at Level-2*

Description

`optimal.bira2c1` finds constrained optimal sample allocation (COSA) solutions for main effct in designs with two-levels where level-1 units are randomly assigned to treatment and control groups within level-2 units (school intercepts only). COSA can be found in the following forms, (i) under budgetary constraints given marginal costs per unit, (ii) under power contraints given marginal costs per unit, (iii) under MDES contraints given marginal costs per unit, and (iv) under sample size contraints for one or more levels along with any of the i,ii, or iii options.

Usage

```
optimal.bira2c1(cn=0, cJ=0, cost=NULL, n=NULL, J=NULL,
                power=.80, mdes=.25, alpha=.05, two.tail=TRUE,
                nJ0=c(10,10), ncase=10, gm=2,
                constrain="power", optimizer="auglag_cobyla",
                P=.50, g1=0, R12=0)
```

Arguments

`cn`	marginal cost per level-1 unit.
`cJ`	marginal cost per level-2 unit.
`cost`	total cost or budget.
`n`	harmonic mean of level-1 units across level-2 units (or simple average).
`J`	level-2 sample size.
`power`	statistical power (1 - type II error).
`mdes`	minimum detectable effect size.
`alpha`	probability of type I error.
`two.tail`	logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing.
`nJ0`	vector with a length of two to specifiy starting values for level-1 and level-2 sample sizes.
`ncase`	number of cases to show in the output.
`gm`	grid multiplier to increase the range of sample size search for each level.
`constrain`	one of the followings can be constrained at a specified cost or value: `"cost"`, `"power"`, or `"mdes"`.
`optimizer`	algorithm to find optimal sample sizes given total cost, power, or MDES. Available algorithms: `"auglag_cobyla"`, `"auglag_lbfgs"`, `"auglag_mma"`, or `"auglag_slsqp"`.
`P`	average proportion of level-1 units randomly assigned to treatment within level-2 units.
`g1`	number of covariates at level-1.
`R12`	proportion of level-1 variance in the outcome explained by level-1 covariates.

Value

fun	function name.
parms	list of parameters used in the function.
nloptr	list of nloptr log and output.
round.optim	solution after rounding. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).
integer.optim	best integer solutions around round.optim solution. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).

See Also

mdes.bira2c1, power.bira2c1

Examples

```
## Not run:

     optimal.bira2c1(cn=1, cJ=10, cost=560,
                     constrain="cost")


## End(Not run)
```

---

optimal.bira2f1 *COSA Solver for Main Effect in Two-Level Fixed Effects Blocked Individual Random Assignment Designs, Treatment at Level-1*

Description

optimal.bira2f1 finds constrained optimal sample allocation (COSA) solutions for main effect in designs with two-levels where level-1 units are randomly assigned to treatment and control groups within level-2 units (fixed blocks). COSA can be found in the following forms, (i) under budgetary constraints given marginal costs per unit, (ii) under power contraints given marginal costs per unit, (iii) under MDES contraints given marginal costs per unit, and (iv) under sample size contraints for one or more levels along with any of the i,ii, or iii options.

Usage

```
optimal.bira2f1(cn=0, cJ=0, cost=NULL, n=NULL, J=NULL,
                power=.80, mdes=.25, alpha=.05, two.tail=TRUE,
                nJ0=c(10,10), ncase=10, gm=2,
                constrain="power", optimizer="auglag_cobyla",
                P=.50, g1=0, R12=0)
```

Arguments

cn	marginal cost per level-1 unit.
cJ	marginal cost per level-2 unit.
cost	total cost or budget.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	level-2 sample size.
power	statistical power (1 - type II error).
mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
nJ0	vector with a length of four to specifiy starting values for level-1 and level-2 sample sizes.
ncase	number of cases to show in the output.
gm	grid multiplier to increase the range of sample size search for each level.
constrain	one of the followings can be constrained at a specified cost or value: "cost", "power", or "mdes".
optimizer	algorithm to find optimal sample sizes given total cost, power, or MDES. Available algorithms: "auglag_cobyla", "auglag_lbfgs", "auglag_mma", or "auglag_slsqp".
P	average proportion of level-1 units randomly assigned to treatment within level-2 units.
g1	number of covariates at level-1.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.

Value

fun	function name.
parms	list of parameters used in the function.
nloptr	list of nloptr log and output.
round.optim	solution after rounding. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).
integer.optim	best integer solutions around round.optim solution. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).

See Also

mdes.bira2f1, power.bira2f1

Examples

```
## Not run:

    optimal.bira2f1(cn=1, cJ=10, cost=560,
                    constrain="cost")


## End(Not run)
```

`optimal.bira2r1` *COSA Solver for Main Effect in Two-Level Random Effects Blocked Individual Random Assignment Designs, Individuals Randomized within Blocks*

Description

`optimal.bira2r1` finds constrained optimal sample allocation (COSA) solutions for main effect in designs with two-levels where level-1 units are randomly assigned to treatment and control groups within level-2 units (random blocks). COSA can be found in the following forms, (i) under budgetary constraints given marginal costs per unit, (ii) under power contraints given marginal costs per unit, (iii) under MDES contraints given marginal costs per unit, and (iv) under sample size contraints for one or more levels along with any of the i,ii, or iii options.

Usage

```
optimal.bira2r1(cn=0, cJ=0, cost=NULL, n=NULL, J=NULL,
                power=.80, mdes=.25, alpha=.05, two.tail=TRUE,
                nJ0=c(10,10), ncase=10, gm=2,
                constrain="power", optimizer="auglag_cobyla",
                rho2, omega2,
                P=.50, g2=0, R12=0, RT22=0)
```

Arguments

`cn`	marginal cost per level-1 unit.
`cJ`	marginal cost per level-2 unit.
`cost`	total cost or budget.
`n`	harmonic mean of level-1 units across level-2 units (or simple average).
`J`	level-2 sample size.
`power`	statistical power (1 - type II error).
`mdes`	minimum detectable effect size.
`alpha`	probability of type I error.
`two.tail`	logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing.
`nJ0`	vector with a length of two to specifiy starting values for level-1 and level-2 sample sizes.
`ncase`	number of cases to show in the output.
`gm`	grid multiplier to increase the range of sample size search for each level.
`constrain`	one of the followings can be constrained at a specified cost or value: `"cost"`, `"power"`, or `"mdes"`.
`optimizer`	algorithm to find optimal sample sizes given total cost, power, or MDES. Available algorithms: `"auglag_cobyla"`, `"auglag_lbfgs"`, `"auglag_mma"`, or `"auglag_slsqp"`.
`rho2`	proportion of variance in the outcome explained by level-2 units.
`omega2`	treatment effect heterogeneity as ratio of treatment effect variance among level-2 units to the residual variance at level-2.

P	average proportion of level-1 units randomly assigned to treatment within level-2 units.
g2	number of covariates at level-2.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
RT22	proportion of treatment effect variance among level-2 units explained by level-2 covariates.

Value

fun	function name.
parms	list of parameters used in the function.
nloptr	list of nloptr log and output.
round.optim	solution after rounding. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).
integer.optim	best integer solutions around round.optim solution. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).

See Also

mdes.bira2r1, power.bira2r1

Examples

```
## Not run:

     optimal.bira2r1(cn=1, cJ=10, cost=560,
                     constrain="cost", rho2=.20, omega2=.50)


## End(Not run)
```

---

optimal.bira3r1 *COSA Solver for Main Effect in Three-Level Random Effects Blocked Individual Random Assignment Design, Individuals Randomized within Blocks*

Description

optimal.bira3r1 finds constrained optimal sample allocation (COSA) solutions for main effect in designs with three-levels where level-1 units are randomly assigned to treatment and control groups within level-2 units (random blocks). COSA can be found in the following forms, (i) under budgetary constraints given marginal costs per unit, (ii) under power contraints given marginal costs per unit, (iii) under MDES contraints given marginal costs per unit, and (iv) under sample size contraints for one or more levels along with any of the i,ii, or iii options.

Usage

```
optimal.bira3r1(cn=0, cJ=0, cK=0, cost=NULL, n=NULL, J=NULL, K=NULL,
                power=.80, mdes=.25, alpha=.05, two.tail=TRUE,
                nJK0=c(10,10,10), ncase=10, gm=2,
                constrain="power", optimizer="auglag_cobyla",
                rho2, rho3, omega2, omega3,
                P=.50, g3=0, R12=0, RT22=0, RT32=0)
```

Arguments

cn	marginal cost per level-1 unit.
cJ	marginal cost per level-2 unit.
cK	marginal cost per level-3 unit.
cost	total cost or budget.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	level-3 sample size.
power	statistical power (1 - type II error).
mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
nJK0	vector with a length of three to specifiy starting values for level-1, level-2, and level-3 sample sizes.
ncase	number of cases to show in the output.
gm	grid multiplier to increase the range of sample size search for each level.
constrain	one of the followings can be constrained at a specified cost or value: "cost", "power", or "mdes".
optimizer	algorithm to find optimal sample sizes given total cost, power, or MDES. Available algorithms: "auglag_cobyla", "auglag_lbfgs", "auglag_mma", or "auglag_slsqp".
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
omega2	treatment effect heterogeneity as ratio of treatment effect variance among level-2 units to the residual variance at level-2.
omega3	treatment effect heterogeneity as ratio of treatment effect variance among level-3 units to the residual variance at level-3.
P	average proportion of level-1 units randomly assigned to treatment within level-2 units.
g3	number of covariates at level-3.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
RT22	proportion of treatment effect variance among level-2 units explained by level-2 covariates.
RT32	proportion of treatment effect variance among level-3 units explained by level-3 covariates.

Value

fun	function name.
parms	list of parameters used in the function.
nloptr	list of nloptr log and output.
round.optim	solution after rounding. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).
integer.optim	best integer solutions around round.optim solution. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).

See Also

mdes.bira3r1, power.bira3r1

Examples

```
## Not run:

     optimal.bira3r1(cn=1, cJ=10, cK=100, cost=5600,
                     constrain="cost",
                     rho3=.20, rho2=.15, omega3=.10, omega2=.10)


## End(Not run)
```

---

optimal.bira4r1 *COSA Solver for Main Effect in Four-Level Random Effects Blocked Individual Random Assignment Design, Individuals Randomized within Blocks*

Description

optimal.bira4r1 finds constrained optimal sample allocation (COSA) solutions for main effect in designs with four-levels where level-1 units are randomly assigned to treatment and control groups within level-2 units (random blocks). COSA can be found in the following forms, (i) under budgetary constraints given marginal costs per unit, (ii) under power contraints given marginal costs per unit, (iii) under MDES contraints given marginal costs per unit, and (iv) under sample size contraints for one or more levels along with any of the i,ii, or iii options.

Usage

```
optimal.bira4r1(cn=0, cJ=0, cK=0, cL=0, cost=NULL,
                n=NULL, J=NULL, K=NULL, L=NULL,
                power=.80, mdes=.25, alpha=.05, two.tail=TRUE,
                nJKL0=c(10,10,10,10), ncase=10, gm=2,
                constrain="power", optimizer="auglag_cobyla",
                rho4, rho3, rho2, omega4, omega3, omega2,
                P=.50, g4=0, RT42=0, RT32=0, RT22=0, R12=0)
```

Arguments

cn	marginal cost per level-1 unit.
cJ	marginal cost per level-2 unit.
cK	marginal cost per level-3 unit.
cL	marginal cost per level-4 unit.
cost	total cost or budget.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	harmonic mean of level-3 units across level-4 units (or simple average).
L	number of level-4 units.
power	statistical power (1 - type II error).
mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
nJKL0	vector with a length of four to specifiy starting values for level-1, level-2, level-3, and level-4 sample sizes.
ncase	number of cases to show in the output.
gm	grid multiplier to increase the range of sample size search for each level.
constrain	one of the followings can be constrained at a specified cost or value: "cost", "power", or "mdes".
optimizer	algorithm to find optimal sample sizes given total cost, power, or MDES. Available algorithms: "auglag_cobyla", "auglag_lbfgs", "auglag_mma", or "auglag_slsqp".
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
rho4	proportion of variance in the outcome explained by level-4 units.
omega2	treatment effect heterogeneity as ratio of treatment effect variance among level-2 units to the residual variance at level-2.
omega3	treatment effect heterogeneity as ratio of treatment effect variance among level-3 units to the residual variance at level-3.
omega4	treatment effect heterogeneity as ratio of treatment effect variance among level-4 units to the residual variance at level-4.
P	average proportion of level-1 units randomly assigned to treatment within level-2 units.
g4	number of covariates at level-4.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
RT22	proportion of treatment effect variance among level-2 units explained by level-2 covariates.
RT32	proportion of treatment effect variance among level-3 units explained by level-3 covariates.
RT42	proportion of treatment effect variance among level-4 units explained by level-4 covariates.

Value

fun	function name.
parms	list of parameters used in the function.
nloptr	list of nloptr log and output.
round.optim	solution after rounding. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).
integer.optim	best integer solutions around round.optim solution. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).

See Also

mdes.bira4r1, power.bira4r1

Examples

```
## Not run:

     optimal.bira4r1(cn=1, cJ=10, cK=100, cL=1000, cost=75600,
                     constrain="cost",
                     rho4=.10, rho3=.15, rho2=.20,
                     omega4=.50, omega3=.50, omega2=.50)


## End(Not run)
```

---

optimal.cra2r2 *COSA Solver for Main Effect in Two-Level Cluster Random Assignment Design, Treatment at Level-2*

Description

optimal.cra2r2 finds constrained optimal sample allocation (COSA) solutions for main effect in designs with two-levels where level-2 units are randomly assigned to treatment and control groups. COSA can be found in the following forms, (i) under budgetary constraints given marginal costs per unit, (ii) under power contraints given marginal costs per unit, (iii) under MDES contraints given marginal costs per unit, and (iv) under sample size contraints for one or more levels along with any of the i,ii, or iii options.

Usage

```
optimal.cra2r2(cn=0, cJ=0, cost=NULL, n=NULL, J=NULL,
               power=.80, mdes=.25, alpha=.05, two.tail=TRUE,
               nJ0=c(10,10), ncase=10, gm=2,
               constrain="power", optimizer="auglag_cobyla",
               rho2, P=.50, g2=0, R12=0, R22=0)
```

Arguments

cn	marginal cost per level-1 unit.
cJ	marginal cost per level-2 unit.
cost	total cost or budget.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	level-2 sample size.
power	statistical power (1 - type II error).
mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
nJ0	vector with a length of two to specifiy starting values for level-1 and level-2 sample sizes.
ncase	number of cases to show in the output.
gm	grid multiplier to increase the range of sample size search for each level.
constrain	one of the followings can be constrained at a specified cost or value: "cost", "power", or "mdes".
optimizer	algorithm to find optimal sample sizes given total cost, power, or MDES. Available algorithms: "auglag_cobyla", "auglag_lbfgs", "auglag_mma", or "auglag_slsqp".
rho2	proportion of variance in the outcome explained by level-2 units.
P	proportion of level-2 units randomly assigned to treatment.
g2	number of covariates at level-2.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.

Value

fun	function name.
parms	list of parameters used in the function.
nloptr	list of nloptr log and output.
round.optim	solution after rounding. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).
integer.optim	best integer solutions around round.optim solution. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).

See Also

mdes.cra2r2, power.cra2r2

Examples

```
## Not run:

     optimal.cra2r2(cn=1, cJ=10, cost=560,
                    constrain="cost", rho2=.20)


## End(Not run)
```

`optimal.cra3r3` *COSA Solver for Main Effect in Three-Level Cluster Random Assignment Designs, Treatment at Level-3*

Description

`optimal.cra3r3` finds constrained optimal sample allocation (COSA) solutions for main effect in designs with three-levels where level-3 units are randomly assigned to treatment and control groups. COSA can be found in the following forms, (i) under budgetary constraints given marginal costs per unit, (ii) under power contraints given marginal costs per unit, (iii) under MDES contraints given marginal costs per unit, and (iv) under sample size contraints for one or more levels along with any of the i,ii, or iii options.

Usage

```
optimal.cra3r3(cn=0, cJ=0, cK=0, cost=NULL,
               n=NULL, J=NULL, K=NULL,
               power=.80, mdes=.25, alpha=.05, two.tail=TRUE,
               nJK0=c(10,10,10), ncase=10, gm=2,
               constrain="power", optimizer="auglag_cobyla",
               rho2, rho3,
               P=.50, g3=0, R12=0, R22=0, R32=0)
```

Arguments

`cn`	marginal cost per level-1 unit.
`cJ`	marginal cost per level-2 unit.
`cK`	marginal cost per level-3 unit.
`cost`	total cost or budget.
`n`	harmonic mean of level-1 units across level-2 units (or simple average).
`J`	harmonic mean of level-2 units across level-3 units (or simple average).
`K`	level-3 sample size.
`power`	statistical power (1 - type II error).
`mdes`	minimum detectable effect size.
`alpha`	probability of type I error.
`two.tail`	logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing.
`nJK0`	vector with a length of three to specifiy starting values for level-1, level-2, and level-3 sample sizes.
`ncase`	number of cases to show in the output.
`gm`	grid multiplier to increase the range of sample size search for each level.
`constrain`	one of the followings can be constrained at a specified cost or value: `"cost"`, `"power"`, or `"mdes"`.
`optimizer`	algorithm to find optimal sample sizes given total cost, power, or MDES. Available algorithms: `"auglag_cobyla"`, `"auglag_lbfgs"`, `"auglag_mma"`, or `"auglag_slsqp"`.
`rho2`	proportion of variance in the outcome explained by level-2 units.

rho3	proportion of variance in the outcome explained by level-3 units.
P	proportion of level-3 units randomly assigned to treatment.
g3	number of covariates at level-3.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
R32	proportion of level-3 variance in the outcome explained by level-3 covariates.

Value

fun	function name.
parms	list of parameters used in the function.
nloptr	list of nloptr log and output.
round.optim	solution after rounding. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).
integer.optim	best integer solutions around round.optim solution. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).

See Also

mdes.cra3r3, power.cra3r3

Examples

```
## Not run:

     optimal.cra3r3(cn=1, cJ=10, cK=100, cost=5600,
                    constrain="cost",
                    rho3=.06, rho2=.17)


## End(Not run)
```

---

optimal.cra4r4 *COSA Solver for Main Effect in Four-Level Cluster Random Assignment Designs, Treatment at Level-4*

Description

optimal.cra4r4 finds constrained optimal sample allocation (COSA) solutions for main effect in designs with four-levels where level-4 units are randomly assigned to treatment and control groups. COSA can be found in the following forms, (i) under budgetary constraints given marginal costs per unit, (ii) under power contraints given marginal costs per unit, (iii) under MDES contraints given marginal costs per unit, and (iv) under sample size contraints for one or more levels along with any of the i,ii, or iii options.

Usage

```
optimal.cra4r4(cn=0, cJ=0, cK=0, cL=0, cost=NULL,
               n=NULL, J=NULL, K=NULL, L=NULL,
               power=.80, mdes=.25, alpha=.05, two.tail=TRUE,
               nJKL0=c(10,10,10,10), ncase=10, gm=2,
               constrain="power", optimizer="auglag_cobyla",
               rho4, rho3, rho2,
               P=.50, g4=0, R42=0, R32=0, R22=0, R12=0)
```

Arguments

cn	marginal cost per level-1 unit.
cJ	marginal cost per level-2 unit.
cK	marginal cost per level-3 unit.
cL	marginal cost per level-4 unit.
cost	total cost or budget.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	harmonic mean of level-3 units across level-4 units (or simple average).
L	number of level-4 units.
power	statistical power (1 - type II error).
mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
nJKL0	vector with a length of four to specifiy starting values for level-1, level-2, level-3, and level-4 sample sizes.
ncase	number of cases to show in the output.
gm	grid multiplier to increase the range of sample size search for each level.
constrain	one of the followings can be constrained at a specified cost or value: "cost", "power", or "mdes".
optimizer	algorithm to find optimal sample sizes given total cost, power, or MDES. Available algorithms: "auglag_cobyla", "auglag_lbfgs", "auglag_mma", or "auglag_slsqp".
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
rho4	proportion of variance in the outcome explained by level-4 units.
P	proportion of level-4 units randomly assigned to treatment.
g4	number of covariates at level-4.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
R32	proportion of level-3 variance in the outcome explained by level-3 covariates.
R42	proportion of level-4 variance in the outcome explained by level-4 covariates.

Value

fun	function name.
parms	list of parameters used in the function.
nloptr	list of nloptr log and output.
round.optim	solution after rounding. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).
integer.optim	best integer solutions around round.optim solution. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).

See Also

mdes.cra4r4, power.cra4r4

Examples

```
## Not run:

    optimal.cra4r4(cn=1, cJ=10, cK=100, cL=1000, cost=75600,
                   constrain="cost",
                   rho4=.05, rho3=.05, rho2=.10)


## End(Not run)
```

---

optimal.ira1r1 *Sample Size Calculator for Main Effect in Individual Random Assignment Designs, Completely Randomized Controlled Trials*

Description

optimal.ira1r1 calculates minimum required sample size for main effect in completely randomized controlled trials where individuals are randomly assigned to treatment or control groups.

Usage

```
optimal.ira1r1(mdes=.25, power=.80, alpha=.05, two.tail=TRUE,
                       n0=10, tol=.10,
                       P=.50, g1=0, R12=0)
```

Arguments

power	statistical power (1 - type II error).
mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
n0	starting value for sample size.
tol	tolerance.

P	average proportion of units randomly assigned to treatment.
g1	number of covariates.
R12	proportion of variance in the outcome explained by covariates.

Value

fun	function name.
parms	list of parameters used in the function.
n	sample size.

See Also

mdes.ira1r1, power.ira1r1

Examples

```
## Not run:

     optimal.ira1r1(R12=.50)


## End(Not run)
```

---

optimal.mod1n.cra2r2 *COSA Solver for Non-randomly Varying Moderator Effect in Two-Level Cluster Random Assignment Design, Treatment at level-2 and Moderator at Level-1*

Description

optimal.mod1n.cra2r2 finds constrained optimal sample allocation (COSA) solutions for non-randomly varying moderator effect in designs with two-levels where level-2 units are randomly assigned to treatment and control groups and moderator is at level-1. COSA can be found in the following forms, (i) under budgetary constraints given marginal costs per unit, (ii) under power contraints given marginal costs per unit, (iii) under MDES contraints given marginal costs per unit, and (iv) under sample size contraints for one or more levels along with any of the i,ii, or iii options.

Usage

```
optimal.mod1n.cra2r2(cn=0, cJ=0, cost=NULL, n=NULL, J=NULL,
                power=.80, mdes=.25, alpha=.05, two.tail=TRUE,
                nJ0=c(10,10), ncase=10, gm=2,
                constrain="power", optimizer="auglag_cobyla",
                rho2, P=.50, Q=NULL, g1=0, R12=0)
```

Arguments

cn	marginal cost per level-1 unit.
cJ	marginal cost per level-2 unit.
cost	total cost or budget.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	level-2 sample size.
power	statistical power (1 - type II error).
mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
nJ0	vector with a length of two to specifiy starting values for level-1 and level-2 sample sizes.
ncase	number of cases to show in the output.
gm	grid multiplier to increase the range of sample size search for each level.
constrain	one of the followings can be constrained at a specified cost or value: "cost", "power", or "mdes".
optimizer	algorithm to find optimal sample sizes given total cost, power, or MDES. Available algorithms: "auglag_cobyla", "auglag_lbfgs", "auglag_mma", or "auglag_slsqp".
rho2	proportion of variance in the outcome explained by level-2 units.
P	proportion of level-2 units randomly assigned to treatment.
Q	proportion of level-1 units within moderator subgroup. NULL implies continuous moderator.
g1	number of covariates at level-1 excluding moderator.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.

Value

fun	function name.
parms	list of parameters used in the function.
nloptr	list of nloptr log and output.
round.optim	solution after rounding. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).
integer.optim	best integer solutions around round.optim solution. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).

See Also

mdes.mod1n.cra2r2, power.mod1n.cra2r2

Examples

```
## Not run:

     optimal.mod1n.cra2r2(cn=1, cJ=10, cost=560,
                          constrain="cost", rho2=.20)


## End(Not run)
```

`optimal.mod1n.cra3r3` *COSA Solver for Non-randomly Varying Moderator Effect in Three-Level Cluster Random Assignment Designs, Treatment at Level-3 and Moderator at Level-1*

Description

`optimal.mod1n.cra3r3` finds constrained optimal sample allocation (COSA) solutions for non-randomly varying moderator effect in designs with three-levels where level-3 units are randomly assigned to treatment and control groups and moderator is at level-1. COSA can be found in the following forms, (i) under budgetary constraints given marginal costs per unit, (ii) under power contraints given marginal costs per unit, (iii) under MDES contraints given marginal costs per unit, and (iv) under sample size contraints for one or more levels along with any of the i,ii, or iii options.

Usage

```
optimal.mod1n.cra3r3(cn=0, cJ=0, cK=0, cost=NULL,
                n=NULL, J=NULL, K=NULL,
                power=.80, mdes=.25, alpha=.05, two.tail=TRUE,
                nJK0=c(10,10,10), ncase=10, gm=2,
                constrain="power", optimizer="auglag_cobyla",
                rho2, rho3, P=.50, Q=NULL, g1=0, R12=0)
```

Arguments

`cn`	marginal cost per level-1 unit.
`cJ`	marginal cost per level-2 unit.
`cK`	marginal cost per level-3 unit.
`cost`	total cost or budget.
`n`	harmonic mean of level-1 units across level-2 units (or simple average).
`J`	harmonic mean of level-2 units across level-3 units (or simple average).
`K`	level-3 sample size.
`power`	statistical power (1 - type II error).
`mdes`	minimum detectable effect size.
`alpha`	probability of type I error.
`two.tail`	logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing.
`nJK0`	vector with a length of three to specifiy starting values for level-1, level-2, and level-3 sample sizes.

ncase	number of cases to show in the output.
gm	grid multiplier to increase the range of sample size search for each level.
constrain	one of the followings can be constrained at a specified cost or value: "cost", "power", or "mdes".
optimizer	algorithm to find optimal sample sizes given total cost, power, or MDES. Available algorithms: "auglag_cobyla", "auglag_lbfgs", "auglag_mma", or "auglag_slsqp".
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
P	proportion of level-3 units randomly assigned to treatment.
Q	proportion of level-1 units within moderator subgroup. NULL implies continuous moderator.
g1	number of covariates at level-1 excluding moderator.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.

Value

fun	function name.
parms	list of parameters used in the function.
nloptr	list of nloptr log and output.
round.optim	solution after rounding. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).
integer.optim	best integer solutions around round.optim solution. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).

See Also

mdes.mod1n.cra3r3, power.mod1n.cra3r3

Examples

```
## Not run:

     optimal.mod1n.cra3r3(cn=1, cJ=10, cK=100, n=10, constrain="mdes",
                          rho2=.20, rho3=.10)


## End(Not run)
```

---

optimal.mod1r.cra2r2 *COSA Solver for Randomly Varying Moderator Effect in Two-Level Cluster Random Assignment Design, Treatment at level-2 and Moderator at Level-1*

Description

optimal.mod1r.cra2r2 finds constrained optimal sample allocation (COSA) solutions for randomly varying moderator effect in designs with two-levels where level-2 units are randomly assigned to treatment and control groups and moderator is at level-1. COSA can be found in the following forms, (i) under budgetary constraints given marginal costs per unit, (ii) under power contraints given marginal costs per unit, (iii) under MDES contraints given marginal costs per unit, and (iv) under sample size contraints for one or more levels along with any of the i,ii, or iii options.

Usage

```
optimal.mod1r.cra2r2(cn=0, cJ=0, cost=NULL, n=NULL, J=NULL,
                power=.80, mdes=.25, alpha=.05, two.tail=TRUE,
                nJ0=c(10,10), ncase=10, gm=2,
                constrain="power", optimizer="auglag_cobyla",
                rho2, omega2, P=.50, Q=NULL, g1=0, R12=0, RT22=0)
```

Arguments

cn	marginal cost per level-1 unit.
cJ	marginal cost per level-2 unit.
cost	total cost or budget.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	level-2 sample size.
power	statistical power (1 - type II error).
mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
nJ0	vector with a length of two to specifiy starting values for level-1 and level-2 sample sizes.
ncase	number of cases to show in the output.
gm	grid multiplier to increase the range of sample size search for each level.
constrain	one of the followings can be constrained at a specified cost or value: "cost", "power", or "mdes".
optimizer	algorithm to find optimal sample sizes given total cost, power, or MDES. Available algorithms: "auglag_cobyla", "auglag_lbfgs", "auglag_mma", or "auglag_slsqp".
rho2	proportion of variance in the outcome explained by level-2 units.
omega2	moderator effect heterogeneity (unconditional) across level-2 units.
P	proportion of level-2 units randomly assigned to treatment.
Q	proportion of level-1 units within moderator subgroup. NULL implies continuous moderator.
g1	number of covariates at level-1 excluding moderator.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
RT22	proportion of level-2 variance in the moderator effect explained by level-2 covariates.

Value

`fun`	function name.
`parms`	list of parameters used in the function.
`nloptr`	list of `nloptr` log and output.
`round.optim`	solution after rounding. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).
`integer.optim`	best integer solutions around `round.optim` solution. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).

See Also

`mdes.mod1r.cra2r2`, `power.mod1r.cra2r2`

Examples

```
## Not run:

     optimal.mod1r.cra2r2(cn=1, cJ=10, cost=560,
                          constrain="cost", rho2=.20, omega=.20)


## End(Not run)
```

`optimal.mod1r.cra3r3` *COSA Solver for Randomly Varying Moderator Effect in Three-Level Cluster Random Assignment Designs, Treatment at Level-3 and Moderator at Level-1*

Description

`optimal.mod1r.cra3r3` finds constrained optimal sample allocation (COSA) solutions for randomly varying moderator effect in designs with three-levels where level-3 units are randomly assigned to treatment and control groups and moderator is at level-1. COSA can be found in the following forms, (i) under budgetary constraints given marginal costs per unit, (ii) under power contraints given marginal costs per unit, (iii) under MDES contraints given marginal costs per unit, and (iv) under sample size contraints for one or more levels along with any of the i,ii, or iii options.

Usage

```
optimal.mod1r.cra3r3(cn=0, cJ=0, cK=0, cost=NULL,
                n=NULL, J=NULL, K=NULL,
                power=.80, mdes=.25, alpha=.05, two.tail=TRUE,
                nJK0=c(10,10,10), ncase=10, gm=2,
                constrain="power", optimizer="auglag_cobyla",
                rho2, rho3, omega2, omega3, P=.50, Q=NULL,
                g1=0, R12=0, RT22=0, RT32=0)
```

Arguments

cn	marginal cost per level-1 unit.
cJ	marginal cost per level-2 unit.
cK	marginal cost per level-3 unit.
cost	total cost or budget.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	level-3 sample size.
power	statistical power (1 - type II error).
mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
nJK0	vector with a length of three to specifiy starting values for level-1, level-2, and level-3 sample sizes.
ncase	number of cases to show in the output.
gm	grid multiplier to increase the range of sample size search for each level.
constrain	one of the followings can be constrained at a specified cost or value: "cost", "power", or "mdes".
optimizer	algorithm to find optimal sample sizes given total cost, power, or MDES. Available algorithms: "auglag_cobyla", "auglag_lbfgs", "auglag_mma", or "auglag_slsqp".
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
omega2	moderator effect heterogeneity (unconditional) across level-2 units.
omega3	moderator effect heterogeneity (unconditional) across level-3 units.
P	proportion of level-3 units randomly assigned to treatment.
Q	proportion of level-1 units within moderator subgroup. NULL implies continuous moderator.
g1	number of covariates at level-1 excluding moderator.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
RT22	proportion of level-2 variance in the moderator effect explained by level-2 covariates.
RT32	proportion of level-3 variance in the moderator effect explained by level-3 covariates.

Value

fun	function name.
parms	list of parameters used in the function.
nloptr	list of nloptr log and output.
round.optim	solution after rounding. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).
integer.optim	best integer solutions around round.optim solution. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).

See Also

mdes.mod1r.cra3r3, power.mod1r.cra3r3

Examples

```
## Not run:

      optimal.mod1r.cra3r3(cn=1, cJ=10, cK=100, n=10, constrain="mdes",
                           rho2=.20, rho3=.10, omega2=.10, omega3=.10)


## End(Not run)
```

`optimal.mod2.cra2r2` *COSA Solver for Moderator Effect in Two-Level Cluster Random Assignment Design, Treatment and Moderator at Level-2*

Description

`optimal.mod2.cra2r2` finds constrained optimal sample allocation (COSA) solutions for moderator effect in designs with two-levels where level-2 units are randomly assigned to treatment and control groups and moderator is at level-2. COSA can be found in the following forms, (i) under budgetary constraints given marginal costs per unit, (ii) under power contraints given marginal costs per unit, (iii) under MDES contraints given marginal costs per unit, and (iv) under sample size contraints for one or more levels along with any of the i,ii, or iii options.

Usage

```
optimal.mod2.cra2r2(cn=0, cJ=0, cost=NULL, n=NULL, J=NULL,
                power=.80, mdes=.25, alpha=.05, two.tail=TRUE,
                nJ0=c(10,10), ncase=10, gm=2,
                constrain="power", optimizer="auglag_cobyla",
                rho2, P=.50, Q=NULL, g2=0, R12=0, R22=0)
```

Arguments

`cn`	marginal cost per level-1 unit.
`cJ`	marginal cost per level-2 unit.
`cost`	total cost or budget.
`n`	harmonic mean of level-1 units across level-2 units (or simple average).
`J`	level-2 sample size.
`power`	statistical power (1 - type II error).
`mdes`	minimum detectable effect size.
`alpha`	probability of type I error.
`two.tail`	logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing.
`nJ0`	vector with a length of two to specifiy starting values for level-1 and level-2 sample sizes.

ncase	number of cases to show in the output.
gm	grid multiplier to increase the range of sample size search for each level.
constrain	one of the followings can be constrained at a specified cost or value: "cost", "power", or "mdes".
optimizer	algorithm to find optimal sample sizes given total cost, power, or MDES. Available algorithms: "auglag_cobyla", "auglag_lbfgs", "auglag_mma", or "auglag_slsqp".
rho2	proportion of variance in the outcome explained by level-2 units.
P	proportion of level-2 units randomly assigned to treatment.
Q	proportion of level-1 units within moderator subgroup. NULL implies continuous moderator.
g2	number of covariates at level-2 excluding moderator.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.

Value

fun	function name.
parms	list of parameters used in the function.
nloptr	list of nloptr log and output.
round.optim	solution after rounding. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).
integer.optim	best integer solutions around round.optim solution. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).

See Also

mdes.mod2.cra2r2, power.mod2.cra2r2

Examples

```
## Not run:

     optimal.mod2.cra2r2(cn=1, cJ=10, cost=560,
                         constrain="cost", rho2=.20)

## End(Not run)
```

---

optimal.mod2n.cra3r3 *COSA Solver for Non-Randomly Varying Moderator Effect in Three-Level Cluster Random Assignment Designs, Treatment at Level-3 and Moderator at Level-2*

Description

optimal.mod2n.cra3r3 finds constrained optimal sample allocation (COSA) solutions for non-randomly varying moderator effect in designs with three-levels where level-3 units are randomly assigned to treatment and control groups and moderator is at level-2. COSA can be found in the following forms, (i) under budgetary constraints given marginal costs per unit, (ii) under power contraints given marginal costs per unit, (iii) under MDES contraints given marginal costs per unit, and (iv) under sample size contraints for one or more levels along with any of the i,ii, or iii options.

Usage

```
optimal.mod2n.cra3r3(cn=0, cJ=0, cK=0, cost=NULL,
                n=NULL, J=NULL, K=NULL,
                power=.80, mdes=.25, alpha=.05, two.tail=TRUE,
                nJK0=c(10,10,10), ncase=10, gm=2,
                constrain="power", optimizer="auglag_cobyla",
                rho2, rho3, P=.50, Q=NULL, g2=0, R12=0, R22=0)
```

Arguments

cn	marginal cost per level-1 unit.
cJ	marginal cost per level-2 unit.
cK	marginal cost per level-3 unit.
cost	total cost or budget.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	level-3 sample size.
power	statistical power (1 - type II error).
mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
nJK0	vector with a length of three to specifiy starting values for level-1, level-2, and level-3 sample sizes.
ncase	number of cases to show in the output.
gm	grid multiplier to increase the range of sample size search for each level.
constrain	one of the followings can be constrained at a specified cost or value: "cost", "power", or "mdes".
optimizer	algorithm to find optimal sample sizes given total cost, power, or MDES. Available algorithms: "auglag_cobyla", "auglag_lbfgs", "auglag_mma", or "auglag_slsqp".
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
P	proportion of level-3 units randomly assigned to treatment.
Q	proportion of level-1 units within moderator subgroup. NULL implies continuous moderator.
g2	number of covariates at level-2 excluding moderator.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.

Value

fun	function name.
parms	list of parameters used in the function.
nloptr	list of nloptr log and output.

round.optim	solution after rounding. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).
integer.optim	best integer solutions around round.optim solution. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).

See Also

mdes.mod2n.cra3r3, power.mod2n.cra3r3

Examples

```
## Not run:

      optimal.mod2n.cra3r3(cn=1, cJ=10, cK=100, n=10, constrain="mdes",
                           rho2=.20, rho3=.10)


## End(Not run)
```

---

optimal.mod2r.cra3r3 *COSA Solver for Randomly Varying Moderator Effect in Three-Level Cluster Random Assignment Designs, Treatment at Level-3 and Moderator at Level-2*

Description

optimal.mod2r.cra3r3 finds constrained optimal sample allocation (COSA) solutions for randomly varying moderator effect in designs with three-levels where level-3 units are randomly assigned to treatment and control groups and moderator is at level-2. COSA can be found in the following forms, (i) under budgetary constraints given marginal costs per unit, (ii) under power contraints given marginal costs per unit, (iii) under MDES contraints given marginal costs per unit, and (iv) under sample size contraints for one or more levels along with any of the i,ii, or iii options.

Usage

```
optimal.mod2r.cra3r3(cn=0, cJ=0, cK=0, cost=NULL,
                n=NULL, J=NULL, K=NULL,
                power=.80, mdes=.25, alpha=.05, two.tail=TRUE,
                nJK0=c(10,10,10), ncase=10, gm=2,
                constrain="power", optimizer="auglag_cobyla",
                rho2, rho3, omega3, P=.50, Q=NULL,
                g2=0, R12=0, R22=0, RT32=0)
```

Arguments

cn	marginal cost per level-1 unit.
cJ	marginal cost per level-2 unit.
cK	marginal cost per level-3 unit.
cost	total cost or budget.
n	harmonic mean of level-1 units across level-2 units (or simple average).

J	harmonic mean of level-2 units across level-3 units (or simple average).
K	level-3 sample size.
power	statistical power (1 - type II error).
mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
nJK0	vector with a length of three to specifiy starting values for level-1, level-2, and level-3 sample sizes.
ncase	number of cases to show in the output.
gm	grid multiplier to increase the range of sample size search for each level.
constrain	one of the followings can be constrained at a specified cost or value: "cost", "power", or "mdes".
optimizer	algorithm to find optimal sample sizes given total cost, power, or MDES. Available algorithms: "auglag_cobyla", "auglag_lbfgs", "auglag_mma", or "auglag_slsqp".
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
omega3	moderator effect heterogeneity (unconditional) across level-3 units.
P	proportion of level-3 units randomly assigned to treatment.
Q	proportion of level-1 units within moderator subgroup. NULL implies continuous moderator.
g2	number of covariates at level-2 excluding moderator.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
RT32	proportion of level-3 variance in the moderator effect explained by level-3 covariates.

Value

fun	function name.
parms	list of parameters used in the function.
nloptr	list of nloptr log and output.
round.optim	solution after rounding. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).
integer.optim	best integer solutions around round.optim solution. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).

See Also

mdes.mod2r.cra3r3, power.mod2r.cra3r3

Examples

```
## Not run:

      optimal.mod2r.cra3r3(cn=1, cJ=10, cK=100, n=10, constrain="mdes",
                           rho2=.20, rho3=.10, omega3=.10)


## End(Not run)
```

`optimal.mod3.cra3r3` *COSA Solver for Moderator Effect in Three-Level Cluster Random Assignment Designs, Treatment and Moderator at Level-3*

Description

`optimal.mod3.cra3r3` finds constrained optimal sample allocation (COSA) solutions for moderator effect in designs with three-levels where level-3 units are randomly assigned to treatment and control groups and moderator is at level-3. COSA can be found in the following forms, (i) under budgetary constraints given marginal costs per unit, (ii) under power contraints given marginal costs per unit, (iii) under MDES contraints given marginal costs per unit, and (iv) under sample size contraints for one or more levels along with any of the i,ii, or iii options.

Usage

```
optimal.mod3.cra3r3(cn=0, cJ=0, cK=0, cost=NULL,
               n=NULL, J=NULL, K=NULL,
               power=.80, mdes=.25, alpha=.05, two.tail=TRUE,
               nJK0=c(10,10,10), ncase=10, gm=2,
               constrain="power", optimizer="auglag_cobyla",
               rho2, rho3, P=.50, Q=NULL,
               g3=0, R12=0, R22=0, R32=0)
```

Arguments

`cn`	marginal cost per level-1 unit.
`cJ`	marginal cost per level-2 unit.
`cK`	marginal cost per level-3 unit.
`cost`	total cost or budget.
`n`	harmonic mean of level-1 units across level-2 units (or simple average).
`J`	harmonic mean of level-2 units across level-3 units (or simple average).
`K`	level-3 sample size.
`power`	statistical power (1 - type II error).
`mdes`	minimum detectable effect size.
`alpha`	probability of type I error.
`two.tail`	logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing.
`nJK0`	vector with a length of three to specifiy starting values for level-1, level-2, and level-3 sample sizes.

ncase	number of cases to show in the output.
gm	grid multiplier to increase the range of sample size search for each level.
constrain	one of the followings can be constrained at a specified cost or value: "cost", "power", or "mdes".
optimizer	algorithm to find optimal sample sizes given total cost, power, or MDES. Available algorithms: "auglag_cobyla", "auglag_lbfgs", "auglag_mma", or "auglag_slsqp".
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
P	proportion of level-3 units randomly assigned to treatment.
Q	proportion of level-1 units within moderator subgroup. NULL implies continuous moderator.
g3	number of covariates at level-3 excluding moderator.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
R32	proportion of level-3 variance in the outcome explained by level-3 covariates.

Value

fun	function name.
parms	list of parameters used in the function.
nloptr	list of nloptr log and output.
round.optim	solution after rounding. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).
integer.optim	best integer solutions around round.optim solution. MDES is calculated at the specified power (default .80), and power is calculated at the specified MDES (default .25).

See Also

mdes.mod3.cra3r3, power.mod3.cra3r3

Examples

```
## Not run:

      optimal.mod3.cra3r3(cn=1, cJ=10, cK=100, n=10, constrain="mdes",
                          rho2=.20, rho3=.10)


## End(Not run)
```

optimal.to.mdes *Optimal to MDES*

Description

optimal.to.mdes converts an object returned from optimal function into an object returned from mdes function.

Usage

```
optimal.to.mdes(x)
```

Arguments

x	an object returned from one of the optimal functions.

Details

optimal.to.mdes converts an object returned from optimal function into an object returned from mdes function by passing parameters through mdes function.

See Also

optimal.to.power

Examples

```
## Not run:

    # object returned from optimal function
    design1 <- optimal.bira2r1(cn=1, cJ=10, cost=560,
                               constrain="cost", rho2=.20, omega2=.50)
    # convert the object into an object returned from mdes function
    design2 <- optimal.to.mdes(design1)


## End(Not run)
```

optimal.to.power *Optimal to Power*

Description

optimal.to.power converts an object returned from optimal function into an object returned from power function.

Usage

```
optimal.to.power(x)
```

Arguments

x	an object returned from one of the optimal functions.

Details

optimal.to.power converts an object returned from optimal function into an object returned from power function by passing parameters through power function.

See Also

optimal.to.mdes

Examples

```
## Not run:

    # object returned from optimal function
    design1 <- optimal.bira2r1(cn=1, cJ=10, cost=560,
                               constrain="cost", rho2=.20, omega2=.50)
    # convert the object into an object returned from power function
    design2 <- optimal.to.power(design1)


## End(Not run)
```

---

power.bcra3f2 *Statistical Power Calculator for Three-Level Fixed Effects Blocked Cluster Random Assignment Design, Treatment at Level-2*

Description

power.bcra3f2 calculates statistical power for designs with three-levels where level-2 units are randomly assigned to treatment and control groups within level-3 units (fixed blocks).

Usage

```
power.bcra3f2(mdes=.25, alpha=.05, two.tail=TRUE,
     rho2,
     P=.50, g2=0, R12=0, R22=0,
     n, J, K, ...)
```

Arguments

mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
P	average proportion of level-2 units randomly assigned to treatment within level-3 units.
g2	number of covariates at level-2.

R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	level-3 sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in power calculation.
ncp	noncentrality parameter.
power	statistical power $(1 - \beta)$.

See Also

mdes.bcra3f2, optimal.bcra3f2

Examples

```
## Not run:

    power.bcra3f2(rho2=.10,
                  n=20, J=44, K=5)


## End(Not run)
```

---

power.bcra3r2 *Statistical Power Calculator for Three-Level Random Effects Blocked Cluster Random Assignment Designs, Treatment at Level-2*

Description

power.bcra3r2 calculates statistical power for designs with three-levels where level-2 units are randomly assigned to treatment and control groups within level-3 units (random blocks).

Usage

```
power.bcra3r2(mdes=.25, alpha=.05, two.tail=TRUE,
     rho2, rho3, omega3,
     P=.50, g3=0, R12=0, R22=0, RT32=0,
     n, J, K, ...)
```

Arguments

mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
omega3	treatment effect heterogeneity as ratio of treatment effect variance among level-3 units to the residual variance at level-3.
P	average proportion of level-2 units randomly assigned to treatment within level-3 units.
g3	number of covariates at level-3.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
RT32	proportion of treatment effect variance among level-3 units explained by level-3 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	level-3 sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in power calculation.
ncp	noncentrality parameter.
power	statistical power $(1 - \beta)$.

See Also

mdes.bcra3r2, optimal.bcra3r2

Examples

```
## Not run:

    power.bcra3r2(rho3=.13, rho2=.10, omega3=.40,
                  n=10, J=6, K=24)


## End(Not run)
```

power.bcra4f3 *Statistical Power Calculator for Four-Level Fixed Effects Blocked Cluster Random Assignment Designs, Treatment at Level-3*

Description

power.bcra4f3 calculates statistical power for designs with four-levels where level-3 units are randomly assigned to treatment and control groups within level-4 units (fixed blocks).

Usage

```
power.bcra4f3(mdes=.25, alpha=.05, two.tail=TRUE,
     rho2, rho3,
     P=.50, R12=0, R22=0, R32=0, g3=0,
     n, J, K, L, ...)
```

Arguments

mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
P	average proportion of level-3 units randomly assigned to treatment within level-4 units.
g3	number of covariates at level-3.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
R32	proportion of level-3 variance in the outcome explained by level-3 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	harmonic mean of level-3 units across level-4 units (or simple average).
L	number of level-4 units.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in power calculation.
ncp	noncentrality parameter.
power	statistical power $(1 - \beta)$.

See Also

mdes.bcra4f3, optimal.bcra4f3

Examples

```
## Not run:

    power.bcra4f3(rho3=.15, rho2=.15,
                  n=10, J=4, K=4, L=15)


## End(Not run)
```

---

power.bcra4r2 *Statistical Power Calculator for Four-Level Random Effects Block Random Assignment Designs, Treatment at Level-2*

Description

power.bcra4r2 calculates statistical power for designs with four-levels where level-2 units are randomly assigned to treatment and control groups within level-3 units (random blocks).

Usage

```
power.bcra4r2(mdes=.25, alpha=.05, two.tail=TRUE,
     rho2, rho3, rho4, omega3, omega4,
     P=.50, R12=0, R22=0, RT32=0, RT42=0, g4=0,
     n, J, K, L, ...)
```

Arguments

mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
rho4	proportion of variance in the outcome explained by level-4 units.
omega3	treatment effect heterogeneity as ratio of treatment effect variance among level-3 units to the residual variance at level-3.
omega4	treatment effect heterogeneity as ratio of treatment effect variance among level-4 units to the residual variance at level-4.
P	average proportion of level-2 units randomly assigned to treatment within level-3 units.
g4	number of covariates at level-4.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
RT32	proportion of treatment effect variance among level-3 units explained by level-3 covariates.
RT42	proportion of treatment effect variance among level-4 units explained by level-4 covariates.

n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	harmonic mean of level-3 units across level-4 units (or simple average).
L	number of level-4 units.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in power calculation.
ncp	noncentrality parameter.
power	statistical power $(1 - \beta)$.

See Also

mdes.bcra4r2, optimal.bcra4r2

Examples

```
## Not run:

    power.bcra4r2(rho4=.05, rho3=.15, rho2=.15,
                  omega4=.50, omega3=.50,
                  n=10, J=4, L=27, K=4)

## End(Not run)
```

---

power.bcra4r3 *Statistical Power Calculator for Four-Level Random Effects Blocked Cluster Random Assignment Designs, Treatment at Level-3*

Description

power.bcra4r3 calculates statistical power for designs with four-levels where level-3 units are randomly assigned to treatment and control groups within level-4 units (random blocks).

Usage

```
power.bcra4r3(mdes=.25, alpha=.05, two.tail=TRUE,
     rho2, rho3, rho4, omega4,
     P=.50, R12=0, R22=0, R32=0, RT42=0, g4=0,
     n, J, K, L, ...)
```

Arguments

mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
rho4	proportion of variance in the outcome explained by level-4 units.
omega4	treatment effect heterogeneity as ratio of treatment effect variance among level-4 units to the residual variance at level-4.
P	average proportion of level-3 units randomly assigned to treatment within level-4 units.
g4	number of covariates at level-4.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
R32	proportion of level-3 variance in the outcome explained by level-3 covariates.
RT42	proportion of treatment effect variance among level-4 units explained by level-4 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	harmonic mean of level-3 units across level-4 units (or simple average).
L	number of level-4 units.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in power calculation.
ncp	noncentrality parameter.
power	statistical power $(1 - \beta)$.

See Also

mdes.bcra4r3, optimal.bcra4r3

Examples

```
## Not run:

    power.bcra4r3(rho4=.05, rho3=.15, rho2=.15,
                  omega4=.50,
                  n=10, J=4, L=27, K=4)

## End(Not run)
```

power.bira2c1 *Statistical Power Calculator for Two-Level Constant Effects Blocked Individual Random Assignment Designs, Treatment at Level-1*

Description

power.bira2c1 calculates statistical power for designs with two-levels where level-1 units are randomly assigned to treatment and control groups within level-2 units (school intercepts only).

Usage

```
power.bira2c1(mdes=.25, alpha=.05, two.tail=TRUE,
              P=.50, g1=0, R12=0,
              n, J, ...)
```

Arguments

mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
P	average proportion of level-1 units randomly assigned to treatment within level-2 units.
g1	number of covariates at level-1.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	level-2 sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in power calculation.
ncp	noncentrality parameter.
power	statistical power $(1 - \beta)$.

See Also

mdes.bira2c1, optimal.bira2c1

Examples

```
## Not run:

   power.bira2c1(n=55, J=3)


## End(Not run)
```

power.bira2f1 *Statistical Power Calculator for Two-Level Fixed Effects Blocked Individual Random Assignment Designs, Treatment at Level-1*

Description

power.bira2f1 calculates statistical power for designs with two-levels where level-1 units are randomly assigned to treatment and control groups within level-2 units (fixed blocks).

Usage

```
power.bira2f1(mdes=.25, alpha=.05, two.tail=TRUE,
              P=.50, g1=0, R12=0,
              n, J, ...)
```

Arguments

mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
P	average proportion of level-1 units randomly assigned to treatment within level-2 units.
g1	number of covariates at level-1.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	level-2 sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in power calculation.
ncp	noncentrality parameter.
power	statistical power $(1 - \beta)$.

See Also

mdes.bira2f1, optimal.bira2f1

Examples

```
## Not run:

   power.bira2f1(n=55, J=3)


## End(Not run)
```

power.bira2r1 *Statistical Power Calculator for Two-Level Random Effects Blocked Individual Random Assignment Designs, Individuals Randomized within Blocks*

Description

power.bira2r1 calculates statistical power for designs with two-levels where level-1 units are randomly assigned to treatment and control groups within level-2 units (random blocks).

Usage

```
power.bira2r1(mdes=.25, alpha=.05, two.tail=TRUE,
              rho2, omega2,
              g2=0, P=.50, R12=0, RT22=0,
              n, J, ...)
```

Arguments

mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
omega2	treatment effect heterogeneity as ratio of treatment effect variance among level-2 units to the residual variance at level-2.
P	average proportion of level-1 units randomly assigned to treatment within level-2 units.
g2	number of covariates at level-2.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
RT22	proportion of treatment effect variance among level-2 units explained by level-2 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	level-2 sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in power calculation.
ncp	noncentrality parameter.
power	statistical power $(1 - \beta)$.

See Also

mdes.bira2r1, optimal.bira2r1

Examples

```
## Not run:

   power.bira2r1(rho2=.35, omega2=.10,
               n=83, J=480)


## End(Not run)
```

`power.bira3r1` *Statistical Power Calculator for Three-Level Random Effects Blocked Individual Random Assignment Design, Individuals Randomized within Blocks*

Description

`power.bira3r1` calculates statistical power for designs with three-levels where level-1 units are randomly assigned to treatment and control groups within level-2 units (random blocks).

Usage

```
power.bira3r1(mdes=.25, alpha=.05, two.tail=TRUE,
              rho2, rho3, omega2, omega3,
              P=.50, R12=0, RT22=0, RT32=0, g3=0,
              n, J, K, ...)
```

Arguments

`mdes`	minimum detectable effect size.
`alpha`	probability of type I error.
`two.tail`	logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing.
`rho2`	proportion of variance in the outcome explained by level-2 units.
`rho3`	proportion of variance in the outcome explained by level-3 units.
`omega2`	treatment effect heterogeneity as ratio of treatment effect variance among level-2 units to the residual variance at level-2.
`omega3`	treatment effect heterogeneity as ratio of treatment effect variance among level-3 units to the residual variance at level-3.
`P`	average proportion of level-1 units randomly assigned to treatment within level-2 units.
`g3`	number of covariates at level-3.
`R12`	proportion of level-1 variance in the outcome explained by level-1 covariates.
`RT22`	proportion of treatment effect variance among level-2 units explained by level-2 covariates.
`RT32`	proportion of treatment effect variance among level-3 units explained by level-3 covariates.
`n`	harmonic mean of level-1 units across level-2 units (or simple average).

J	harmonic mean of level-2 units across level-3 units (or simple average).
K	level-3 sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in power calculation.
ncp	noncentrality parameter.
power	statistical power $(1 - \beta)$.

See Also

mdes.bira3r1, optimal.bira3r1

Examples

```
## Not run:

    power.bira3r1(rho3=.20, rho2=.15, omega3=.10, omega2=.10,
                  n=69, J=10, K=100)

## End(Not run)
```

---

power.bira4r1 *Statistical Power Calculator for Four-Level Random Effects Blocked Individual Random Assignment Design, Individuals Randomized within Blocks*

Description

power.bira4r1 calculates statistical power for designs with four-levels where level-1 units are randomly assigned to treatment and control groups within level-2 units (random blocks).

Usage

```
power.bira4r1(mdes=.25, alpha=.05, two.tail=TRUE,
     rho2, rho3, rho4, omega2, omega3, omega4,
     P=.50, R12=0, RT22=0, RT32=0, RT42=0, g4=0,
     n, J, K, L, ...)
```

Arguments

mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.

rho4	proportion of variance in the outcome explained by level-4 units.
omega2	treatment effect heterogeneity as ratio of treatment effect variance among level-2 units to the residual variance at level-2.
omega3	treatment effect heterogeneity as ratio of treatment effect variance among level-3 units to the residual variance at level-3.
omega4	treatment effect heterogeneity as ratio of treatment effect variance among level-4 units to the residual variance at level-4.
P	average proportion of level-1 units randomly assigned to treatment within level-2 units.
g4	number of covariates at level-4.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
RT22	proportion of treatment effect variance among level-2 units explained by level-2 covariates.
RT32	proportion of treatment effect variance among level-3 units explained by level-3 covariates.
RT42	proportion of treatment effect variance among level-4 units explained by level-4 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	harmonic mean of level-3 units across level-4 units (or simple average).
L	number of level-4 units.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in power calculation.
ncp	noncentrality parameter.
power	statistical power $(1 - \beta)$.

See Also

mdes.bira4r1, optimal.bira4r1

Examples

```
## Not run:

    power.bira4r1(rho4=.05, rho3=.15, rho2=.15,
                  omega4=.50, omega3=.50, omega2=.50,
                  n=10, J=4, L=27, K=4)

## End(Not run)
```

power.cra2r2 *Statistical Power Calculator for Main Effect in Two-Level Cluster Random Assignment Design, Treatment at Level-2*

Description

power.cra2r2 calculates statistical power for main effect in designs with two-levels where level-2 units are randomly assigned to treatment and control groups.

Usage

```
power.cra2r2(mdes=.25, alpha=.05, two.tail=TRUE,
                  rho2, g2=0, P=.50, R12=0, R22=0,
                  n, J, ...)
```

Arguments

mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
P	proportion of level-2 units randomly assigned to treatment.
g2	number of covariates at level-2.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	level-2 sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in power calculation.
ncp	noncentrality parameter.
power	statistical power $(1 - \beta)$.

See Also

mdes.cra2r2, optimal.cra2r2

Examples

```
## Not run:

   power.cra2r2(rho2=.20,
                    n=4, J=20)


## End(Not run)
```

`power.cra3r3` *Statistical Power Calculator for Main Effect in Three-Level Cluster Random Assignment Designs, Treatment at Level-3*

Description

`power.cra3r3` calculates statistical power for main effect in designs with three-levels where level-3 units are randomly assigned to treatment and control groups.

Usage

```
power.cra3r3(mdes=.25, alpha=.05, two.tail=TRUE,
             rho2, rho3,
             P=.50, g3=0, R12=0, R22=0, R32=0,
             n, J, K, ...)
```

Arguments

`mdes`	minimum detectable effect size.
`alpha`	probability of type I error.
`two.tail`	logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing.
`rho2`	proportion of variance in the outcome explained by level-2 units.
`rho3`	proportion of variance in the outcome explained by level-3 units.
`P`	proportion of level-3 units randomly assigned to treatment.
`g3`	number of covariates at level-3.
`R12`	proportion of level-1 variance in the outcome explained by level-1 covariates.
`R22`	proportion of level-2 variance in the outcome explained by level-2 covariates.
`R32`	proportion of level-3 variance in the outcome explained by level-3 covariates.
`n`	harmonic mean of level-1 units across level-2 units (or simple average).
`J`	harmonic mean of level-2 units across level-3 units (or simple average).
`K`	level-3 sample size.
`...`	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in power calculation.
ncp	noncentrality parameter.
power	statistical power $(1 - \beta)$.

See Also

mdes.cra3r3, optimal.cra3r3

Examples

```
## Not run:

    power.cra3r3(rho3=.06, rho2=.17,
                        n=15, J=3, K=60)


## End(Not run)
```

---

power.cra4r4 *Statistical Power Calculator for Four-Level Cluster Random Assignment Designs, Treatment at Level-4*

Description

power.cra4r4 calculates statistical power for designs with four-levels where level-4 units are randomly assigned to treatment and control groups.

Usage

```
power.cra4r4(mdes=.25, alpha=.05, two.tail=TRUE,
     rho2, rho3, rho4,
     P=.50, R12=0, R22=0, R32=0, R42=0, g4=0,
     n, J, K, L, ...)
```

Arguments

mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
rho4	proportion of variance in the outcome explained by level-4 units.
P	proportion of level-4 units randomly assigned to treatment.
g4	number of covariates at level-4.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.

R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
R32	proportion of level-3 variance in the outcome explained by level-3 covariates.
R42	proportion of level-4 variance in the outcome explained by level-4 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	harmonic mean of level-3 units across level-4 units (or simple average).
L	number of level-4 units.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in power calculation.
ncp	noncentrality parameter.
power	statistical power $(1 - \beta)$.

See Also

mdes.cra4r4, optimal.cra4r4

Examples

```
## Not run:

    power.cra4r4(rho4=.05, rho3=.05, rho2=.10,
                 n=10, J=2, K=3, L=20)


## End(Not run)
```

---

power.ira1r1 *Statistical Power Calculator for Individual Random Assignment Designs, Completely Randomized Controlled Trials*

Description

power.ira1r1 calculates statistical power for completely randomized controlled trials where individuals are randomly assigned to treatment or control groups.

Usage

```
power.ira1r1(mdes=.25, alpha=.05, two.tail=TRUE,
             P=.50, g1=0, R12=0,
             n, ...)
```

Arguments

mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
P	proportion of units randomly assigned to treatment.
g1	number of covariates.
R12	proportion of variance in the outcome explained by covariates.
n	sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in power calculation.
ncp	noncentrality parameter.
power	statistical power $(1 - \beta)$.

See Also

mdes.ira1r1, optimal.ira1r1

Examples

```
## Not run:

   power.ira1r1(n=55)


## End(Not run)
```

---

power.mod1n.cra2r2 *Statistical Power Calculator for for Non-Randomly Varying Moderator Effect in Two-Level Cluster Random Assignment Design, Treatment at Level-2, Moderator at Level-1*

Description

power.mod1n.cra2r2 calculates statistical power for non-randomly varying moderator effect in designs with two-levels where level-2 units are randomly assigned to treatment and control groups and moderator is at level-1.

Usage

```
power.mod1n.cra2r2(mdes=.25, alpha=.05, two.tail=TRUE,
                   rho2, P=.50, Q=NULL, g1=0, R12=0,
                   n, J, ...)
```

Arguments

mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
P	proportion of level-2 units randomly assigned to treatment.
Q	proportion of level-1 units within moderator subgroup. NULL implies continuous moderator.
g1	number of covariates at level-1 excluding moderator.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	level-2 sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in power calculation.
ncp	noncentrality parameter.
power	statistical power $(1 - \beta)$.

See Also

mdes.mod1n.cra2r2, optimal.mod1n.cra2r2

Examples

```
## Not run:

   power.mod1n.cra2r2(rho2=.20,
                      n=4, J=20)


## End(Not run)
```

---

power.mod1n.cra3r3 *Statistical Power Calculator for Non-Randomly Varying Moderator Effect in Three-Level Cluster Random Assignment Design, Treatment at Level-3, Moderator at Level-1*

Description

power.mod1n.cra3r3 calculates statistical power for non-randomly varying moderator effect in designs with three-levels where level-3 units are randomly assigned to treatment and control groups and moderator is at level-1.

Usage

```
power.mod1n.cra3r3(mdes=.25, alpha=.05, two.tail=TRUE,
                   rho2, rho3, P=.50, Q=NULL, g1=0, R12=0,
                   n, J, K, ...)
```

Arguments

mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
P	proportion of level-3 units randomly assigned to treatment.
Q	proportion of level-1 units within moderator subgroup. NULL implies continuous moderator.
g1	number of covariates at level-1 excluding moderator.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	level-3 sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in power calculation.
ncp	noncentrality parameter.
power	statistical power $(1 - \beta)$.

See Also

mdes.mod1n.cra3r3, optimal.mod1n.cra3r3

Examples

```
## Not run:

    power.mod1n.cra3r3(rho3=.06, rho2=.17,
                       n=15, J=3, K=60)


## End(Not run)
```

`power.mod1r.cra2r2` *Statistical Power Calculator for Randomly Varying Moderator Effect in Two-Level Cluster Random Assignment Design, Treatment at Level-2, Moderator at Level-1*

Description

`power.mod1r.cra2r2` calculates statistical power for randomly varying moderator effect in designs with two-levels where level-2 units are randomly assigned to treatment and control groups and moderator is at level-1.

Usage

```
power.mod1r.cra2r2(mdes=.25, alpha=.05, two.tail=TRUE,
                   rho2, omega2, RT22=0, P=.50, Q=NULL, g1=0, R12=0,
                   n, J, ...)
```

Arguments

`mdes`	minimum detectable effect size.
`alpha`	probability of type I error.
`two.tail`	logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing.
`rho2`	proportion of variance in the outcome explained by level-2 units.
`omega2`	moderator effect heterogeneity (unconditional) across level-2 units.
`P`	proportion of level-2 units randomly assigned to treatment.
`Q`	proportion of level-1 units within moderator subgroup. `NULL` implies continuous moderator.
`g1`	number of covariates at level-1 excluding moderator.
`R12`	proportion of level-1 variance in the outcome explained by level-1 covariates.
`RT22`	proportion of level-2 variance in the moderator effect explained by level-2 covariates.
`n`	harmonic mean of level-1 units across level-2 units (or simple average).
`J`	level-2 sample size.
`...`	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

`fun`	function name.
`parms`	list of parameters used in power calculation.
`ncp`	noncentrality parameter.
`power`	statistical power $(1 - \beta)$.

See Also

`mdes.mod1r.cra2r2`, `optimal.mod1r.cra2r2`

Examples

```
## Not run:

   power.mod1r.cra2r2(rho2=.2, omega2=.2, RT22=.2, n=4, J=20)


## End(Not run)
```

`power.mod1r.cra3r3` *Statistical Power Calculator for Randomly Varying Moderator Effect in Three-Level Cluster Random Assignment Designs, Treatment at Level-3, Moderator at Level-1*

Description

`power.mod1r.cra3r3` calculates statistical power for randomly varying moderator effect in designs with three-levels where level-3 units are randomly assigned to treatment and control groups and moderator is at level-1.

Usage

```
power.mod1r.cra3r3(mdes=.25, alpha=.05, two.tail=TRUE,
                   rho2, rho3, omega2, omega3, P=.50, Q=NULL,
                   g1=0, R12=0, RT22=0, RT32=0,
                   n, J, K, ...)
```

Arguments

`mdes`	minimum detectable effect size.
`alpha`	probability of type I error.
`two.tail`	logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing.
`rho2`	proportion of variance in the outcome explained by level-2 units.
`rho3`	proportion of variance in the outcome explained by level-3 units.
`omega2`	moderator effect heterogeneity (unconditional) across level-2 units.
`omega3`	moderator effect heterogeneity (unconditional) across level-3 units.
`P`	proportion of level-3 units randomly assigned to treatment.
`Q`	proportion of level-1 units within moderator subgroup. `NULL` implies continuous moderator.
`g1`	number of covariates at level-1 excluding moderator.
`R12`	proportion of level-1 variance in the outcome explained by level-1 covariates.
`RT22`	proportion of level-2 variance in the moderator effect explained by level-2 covariates.
`RT32`	proportion of level-3 variance in the moderator effect explained by level-3 covariates.
`n`	harmonic mean of level-1 units across level-2 units (or simple average).
`J`	harmonic mean of level-2 units across level-3 units (or simple average).
`K`	level-3 sample size.
`...`	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in power calculation.
ncp	noncentrality parameter.
power	statistical power $(1 - \beta)$.

See Also

mdes.mod1r.cra3r3, optimal.mod1r.cra3r3

Examples

```
## Not run:

    power.mod1r.cra3r3(rho3=.05, rho2=.12, omega2=.08, omega3=.07,
                       P=.4, Q=.7, g1=1, R12=.20, RT22=0, RT32=0,
                       n=20, J=4, K=60)


## End(Not run)
```

---

power.mod2.cra2r2 *Statistical Power Calculator for Moderator Effect in Two-Level Cluster Random Assignment Design, Treatment and Moderator at Level-2*

Description

power.mod2.cra2r2 calculates statistical power for moderator effect in designs with two-levels where level-2 units are randomly assigned to treatment and control groups and moderator is at level-2.

Usage

```
power.mod2.cra2r2(mdes=.25, alpha=.05, two.tail=TRUE,
                  rho2, P=.50, Q=NULL, g2=0, R12=0, R22=0,
                  n, J, ...)
```

Arguments

mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
P	proportion of level-2 units randomly assigned to treatment.
Q	proportion of level-1 units within moderator subgroup. NULL implies continuous moderator.
g2	number of covariates at level-2 excluding moderator.

R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	level-2 sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in power calculation.
ncp	noncentrality parameter.
power	statistical power $(1 - \beta)$.

See Also

mdes.mod2.cra2r2, optimal.mod2.cra2r2

Examples

```
## Not run:

   power.mod2.cra2r2(rho2=.20, n=4, J=20, Q=.5)


## End(Not run)
```

---

power.mod2n.cra3r3 *Statistical Power Calculator for Non-Randomly Varying Moderator Effect in Three-Level Cluster Random Assignment Design, Treatment at Level-3, Moderator at Level-2*

Description

power.mod2n.cra3r3 calculates statistical power for non-randomly varying moderator effect in designs with three-levels where level-3 units are randomly assigned to treatment and control groups and moderator is at level-2.

Usage

```
power.mod2n.cra3r3(mdes=.25, alpha=.05, two.tail=TRUE,
                   rho2, rho3, P=.50, Q=NULL, g2=0, R12=0, R22=0,
                   n, J, K, ...)
```

Arguments

mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
P	proportion of level-3 units randomly assigned to treatment.
Q	proportion of level-1 units within moderator subgroup. NULL implies continuous moderator.
g2	number of covariates at level-2 excluding moderator.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	level-3 sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in power calculation.
ncp	noncentrality parameter.
power	statistical power $(1 - \beta)$.

See Also

mdes.mod2n.cra3r3, optimal.mod2n.cra3r3

Examples

```
## Not run:

     power.mod2n.cra3r3(rho3=.10, rho2=.10, Q=.5, g2=1,
                        R12=.30, R22=.40,  n=20, J=4, K=60)


## End(Not run)
```

`power.mod2r.cra3r3` *Statistical Power Calculator for Randomly Varying Moderator Effect in Three-Level Cluster Random Assignment Designs, Treatment at Level-3, Moderator at Level-2*

Description

`power.mod2r.cra3r3` calculates statistical power for randomly varying moderator effect in designs with three-levels where level-3 units are randomly assigned to treatment and control groups and moderator is at level-2.

Usage

```
power.mod2r.cra3r3(mdes=.25, alpha=.05, two.tail=TRUE,
                rho2, rho3, omega3, P=.50, Q=NULL, g2=0, R12=0, R22=0, RT32=0,
                    n, J, K, ...)
```

Arguments

`mdes`	minimum detectable effect size.
`alpha`	probability of type I error.
`two.tail`	logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing.
`rho2`	proportion of variance in the outcome explained by level-2 units.
`rho3`	proportion of variance in the outcome explained by level-3 units.
`omega3`	moderator effect heterogeneity (unconditional) across level-3 units.
`P`	proportion of level-3 units randomly assigned to treatment.
`Q`	proportion of level-1 units within moderator subgroup. `NULL` implies continuous moderator.
`g2`	number of covariates at level-2 excluding moderator.
`R12`	proportion of level-1 variance in the outcome explained by level-1 covariates.
`R22`	proportion of level-2 variance in the outcome explained by level-2 covariates.
`RT32`	proportion of level-3 variance in the moderator effect explained by level-3 covariates.
`n`	harmonic mean of level-1 units across level-2 units (or simple average).
`J`	harmonic mean of level-2 units across level-3 units (or simple average).
`K`	level-3 sample size.
`...`	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

`fun`	function name.
`parms`	list of parameters used in power calculation.
`ncp`	noncentrality parameter.
`power`	statistical power $(1 - \beta)$.

See Also

mdes.mod2r.cra3r3, optimal.mod2r.cra3r3

Examples

```
## Not run:

    power.mod2r.cra3r3(rho3=.10, rho2=.10, omega3=.05, Q=.5, g2=1,
                       R12=.30, R22=.40, RT32=0,  n=20, J=4, K=60)


## End(Not run)
```

---

power.mod3.cra3r3 *Statistical Power Calculator for Moderator Effect in Three-Level Cluster Random Assignment Designs, Treatment and Moderator at Level-3*

Description

power.mod3.cra3r3 calculates statistical power for moderator effect in designs with three-levels where level-3 units are randomly assigned to treatment and control groups and moderator is at level-3.

Usage

```
power.mod3.cra3r3(mdes=.25, alpha=.05, two.tail=TRUE,
                  rho2, rho3, P=.50, Q=NULL, g3=0, R12=0, R22=0, R32=0,
                  n, J, K, ...)
```

Arguments

mdes	minimum detectable effect size.
alpha	probability of type I error.
two.tail	logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.
rho2	proportion of variance in the outcome explained by level-2 units.
rho3	proportion of variance in the outcome explained by level-3 units.
P	proportion of level-3 units randomly assigned to treatment.
Q	proportion of level-1 units within moderator subgroup. NULL implies continuous moderator.
g3	number of covariates at level-3 excluding moderator.
R12	proportion of level-1 variance in the outcome explained by level-1 covariates.
R22	proportion of level-2 variance in the outcome explained by level-2 covariates.
R32	proportion of level-3 variance in the outcome explained by level-3 covariates.
n	harmonic mean of level-1 units across level-2 units (or simple average).
J	harmonic mean of level-2 units across level-3 units (or simple average).
K	level-3 sample size.
...	to handle extra parameters passed from other functions, do not define any additional parameters.

Value

fun	function name.
parms	list of parameters used in power calculation.
ncp	noncentrality parameter.
power	statistical power $(1 - \beta)$.

See Also

mdes.mod3.cra3r3, optimal.mod3.cra3r3

Examples

```
## Not run:

    power.mod3.cra3r3(rho3=.1, rho2=.1, omega3=.05, Q=.5,
                  g3=1, R12=.3, R22=.4, R32=.5, n=20, J=4, K=60)


## End(Not run)
```

---

power.to.mdes *power to MDES*

Description

power.to.mdes converts an object returned from power function into an object returned from mdes function.

Usage

```
power.to.mdes(x)
```

Arguments

x	an object returned from one of the power functions.

Details

power.to.mdes converts an object returned from power function into an object returned from mdes function by passing parameters through mdes function.

See Also

optimal.to.mdes

Examples

```
## Not run:

  # object returned from power function
  design1 <- power.bira2r1(mdes=.15, rho2=.35, omega2=.10, n=83, J=10)
  # convert the object into an object returned from mdes function
  design2 <- power.to.mdes(design1)


## End(Not run)
```

PowerUpR-deprecated *Deprecated and Defunct functions in 'PowerUpR'*

Description

Deprecated functions are provided for compatibility purposes and will be defunct in the next 'PowerUpR' release. Defunct functions are no longer supported.

Format

Deprecated functions are no longer documented, please use documentation for new functions.

Details

Deprecated and defunct functions and their replacement are as follows:

- `mrss.ira1r1` is deprecated, use `optimal.ira1r1`
- `mrss.cra2r2` is deprecated, use `optimal.cra2r2`
- `mrss.cra3r3` is deprecated, use `optimal.cra3r3`
- `mrss.cra4r4` is deprecated, use `optimal.cra4r4`
- `mrss.bcra3f2` is deprecated, use `optimal.bcra3f2`
- `mrss.bcra3r2` is deprecated, use `optimal.bcra3r2`
- `mrss.bcra4r2` is deprecated, use `optimal.bcra4r2`
- `mrss.bcra4f3` is deprecated, use `optimal.bcra4f3`
- `mrss.bcra4r2` is deprecated, use `optimal.bcra4r2`
- `mrss.bira4r1` is deprecated, use `optimal.bira4r1`
- `mrss.bira3r1` is deprecated, use `optimal.bira3r1`
- `mrss.bira2c1` is deprecated, use `optimal.bira2c1`
- `mrss.bira2f1` is deprecated, use `optimal.bira2f1`
- `mrss.bira2r1` is deprecated, use `optimal.bira2r1`
- `mrss.to.mdes` is defunct, use `optimal.to.mdes`
- `mrss.to.power` is defunct, use `optimal.to.power`
- `t1t2.error` is defunct, there is no replacement function
- `plot.pars` is defunct, there is no replacement function

Index