Package 'cosa'

December 16, 2018

Type Package

Title Bound Constrained Optimal Sample Allocation

Version 1.2.1

Date 2018-12-16

Description Implements bound constrained optimization of sample sizes in power analysis of multilevel regression discontinuity studies (RDDs) and multilevel randomized controlled trials (RCTs) with continuous outcomes. Functions for statistical power and minimum detectable effect size calculations are also provided. Bulus, M. (2017). Design Considerations in Three-level Regression Discontinuity Studies (Doctoral dissertation). University of Missouri, Columbia, MO.

Imports nloptr(>= 1.0.4)

Suggests knitr, rmarkdown

VignetteBuilder knitr

License GPL (>= 3)

NeedsCompilation no

Author Metin Bulus [aut, cre, cph],
Nianbo Dong [aut, cph]

Maintainer Metin Bulus <bulusmetin@gmail.com>

Repository CRAN

Date/Publication 2018-12-16 18:40:10 UTC

R topics documented:

`cosa-package` *Bound Constrained Optimal Design of Multilevel RDDs and RCTs*

Description

Bound Constrained Optimal Sample Allocation (BCOSA) functions are designed to optimize sample sizes at one or more levels subject to budget, statistical power, or effect size constraints. BCOSA can be found in the following forms; (i) under budgetary constraints given marginal costs per unit while minimizing the sampling variance of the treatment effect (or, alternatively, while maximizing statistical power), (ii) under statistical power or effect size (ES) constraints while minimizing the total cost, and (iii) under sample size constraints for one or more levels along with (i) or (ii). Specifying `rhots = 0` produces results equivalent to corresponding random assignment designs, which means there is no relationship between the treatment [random] and the score variable. Therefore, BCOSA functions also allows optimization of proportion of treatment allocation (`p`) under unequal marginal costs when primary constraint is on the total cost. Different starting values and algorithms may produce different results especially when marginal cost information is not provided and sample sizes at two or more levels and `p` are optimized. Comparing several algorithms may faciliate decisions regarding sample sizes and `p`.

Designs available in **cosa** package:

Design	*Total Levels*	*Treatment Level*	*Top Level*
`ird1r1`	1	1	random
`bird2r1`	2	1	random
`bird2f1`	2	1	fixed
`bird3r1`	3	1	random
`bird4r1`	4	1	random
`crd2r2`	2	2	random
`bcrd3f2`	3	2	fixed
`bcrd3r2`	3	2	random
`bcrd4r2`	4	2	random
`crd3r3`	3	3	random
`bcrd4f3`	4	3	fixed
`bcrd4r3`	4	3	random
`crd4r4`	4	4	random

`ird`: individual-level regression discontinuity. `bird`: blocked individual-level regression discontinuity. `crd`: cluster-level regression discontinuity. `bcrd`: blocked cluster-level regression discontinuity.

Design parameters follows a sequential order. Numbers at the end of a sequential parameter refers to the corresponding level. For example `rho2` is the proportion of variance in the outcome between

level 2 units, rho3 is the proportion of variance in the outcome between level 3 units. Similiarly, r21 is the proportion of the variance in the outcome explained by level 1 covariates, r22 is the proportion of the variance in the outcome explained by level 2 covariates and so on. Similiar naming conventions applies to other design parameters.

Author(s)

Metin Bulus <bulusmetin@gmail.com>

---

bcrd3r2 *Blocked Cluster-level Regression Discontinuity (Three-level Design, Discontinuity at Level 2)*

Description

Use mdes.bcrd3r2() to calculate minimum detectable effect size, power.bcrd3r2() to calculate statistical power, and cosa.bcrd3r2() for constrained optimal sample allocation.

Usage

```
cosa.bcrd3r2(cn1 = 0, cn2 = 0, cn3 = 0, cost = NULL,
             n1 = NULL, n2 = NULL, n3 = NULL, p = NULL,
             n0 = c(10, 3, 100 + g3), p0 = .499, constrain = "power",
             round = TRUE, max.power = FALSE,
             local.solver = c("LBFGS", "SLSQP", "MMA", "COBYLA"),
             rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
             power = .80, es = .25, alpha = .05, two.tailed = TRUE,
             rho2, rho3, omega3, g3 = 0, r21 = 0, r22 = 0, r2t3 = 0)

mdes.bcrd3r2(power = .80, alpha = .05, two.tailed = TRUE,
             rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
             rho2, rho3, omega3, r21 = 0, r22 = 0, r2t3 = 0,
             g3 = 0, p = .50, n1, n2, n3)

power.bcrd3r2(es = .25, alpha = .05, two.tailed = TRUE,
              rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
              rho2, rho3, omega3, r21 = 0, r22 = 0, r2t3 = 0,
              g3 = 0, p = .50, n1, n2, n3)
```

Arguments

cn1	marginal cost per level 1 unit in treatment and control conditions, e.g. c(10, 5).
cn2	marginal cost per level 2 unit in treatment and control conditions, e.g. c(50, 20).
cn3	marginal cost per level 3 unit.
cost	total cost or budget.
n1	average number of level 1 units per level 2 unit.

n2	average number of level 2 units per level 3 unit.
n3	number of level 3 units.
p	proportion of level 2 units in treatment condition.
n0	vector of starting values for n1, n2, n3 (positional). Starting values are replaced with averages when sample sizes are constrained by bounds.
p0	starting value for p when rhots = 0 and p = NULL. Starting value is replaced with average when p is constrained by bounds.
power	statistical power (1 - β).
es	effect size (Cohen's d).
alpha	probability of type I error (α).
two.tailed	logical; TRUE for two-tailed hypothesis testing.
constrain	character; "cost", "power", or "mdes".
round	logical; TRUE for rounded COSA solution.
max.power	logical; TRUE for maximizing power instead of minimizing variance.
local.solver	subset of c("LBFGS", "SLSQP", "MMA", "COBYLA")
rhots	correlation between the treatment and the scoring variable. Specify rhots = 0 to obtain results equivalent to random assignment designs.
k1	left truncation point (in standard deviation units from full normal distribution mean), ignored when rhots is not NULL or dists = "uniform".
k2	right truncation point (in standard deviation units from full normal distribution mean), ignored when rhots is not NULL or dists = "uniform".
dists	distribution of the scoring variable; "normal" or "uniform". By default, dists = "normal" specification implies a truncated normal distribution with k1 = -6 and k2 = 6.
rho2	proportion of variance in the outcome between level 2 units (unconditional ICC2).
rho3	proportion of variance in the outcome between level 3 units (unconditional ICC3).
omega3	ratio of the treatment effect variance between level 3 units to the variance in the outcome between level 3 units.
g3	number of covariates at level 3.
r21	proportion of level 1 variance in the outcome explained by level 1 covariates.
r22	proportion of level 1 variance in the outcome explained by level 2 covariates.
r2t3	proportion of treatment effect variance between level 3 units explained by level 3 covariates.

Value

parms	list of parameters used in the function.
df	degrees of freedom.
sse	standardized standard error.
cosa	constrained optimal sample allocation.
mdes	minimum detectable effect size and (1 - α)% confidence limits.
power	statistical power (1 - β)

Examples

```
# cost constrained - optimize n2 and n3
cosa.bcrd3r2(constrain = "cost", cost = 20000,
             cn1 = c(5, 3), cn2 = 10, cn3 = 30,
             es = .20, rho2 = .20, rho3 = .10, omega3 = .10,
             r21 = .20, r22 = .20, r2t3 = .20,
             g3 = 1, p = .50, n1 = 25)

# minimum detectable effect size
mdes.bcrd3r2(power = .80, rho2 = .20, rho3 = .10, omega3 = .10,
             r21 = .20, r22 = .30, r2t3 = .40,
             g3 = 1, p = .50, n1 = 13.39, n2 = 3, n3 = 277)

# statistical power
power.bcrd3r2(es = .20, rho2 = .20, rho3 = .10, omega3 = .10,
              r21 = .20, r22 = .30, r2t3 = .40,
              g3 = 1, p = .50, n1 = 13.39, n2 = 3, n3 = 277)
```

---

bcrd4r2 *Blocked Cluster-level Regression Discontinuity (Four-level Design, Discontinuity at Level 2)*

Description

Use `mdes.bcrd4r2()` to calculate minimum detectable effect size, `power.bcrd4r2()` to calculate statistical power, and use `cosa.bcrd4r2()` for constrained optimal sample allocation.

Usage

```
cosa.bcrd4r2(cn1 = 0, cn2 = 0, cn3 = 0, cn4 = 0, cost = NULL,
             n1 = NULL, n2 = NULL, n3 = NULL, n4 = NULL, p = NULL,
             n0 = c(10, 3, 100, 5 + g4), p0 = .499, constrain = "power",
             round = TRUE, max.power = FALSE,
             local.solver = c("LBFGS", "SLSQP", "MMA", "COBYLA"),
             rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
             power = .80, es = .25, alpha = .05, two.tailed = TRUE,
             rho2, rho3, rho4, omega3, omega4,
             g4 = 0, r21 = 0, r22 = 0, r2t3 = 0, r2t4 = 0)

mdes.bcrd4r2(power = .80, alpha = .05, two.tailed = TRUE,
             rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
             rho2, rho3, rho4, omega3, omega4,
             r21 = 0, r22 = 0, r2t3 = 0, r2t4 = 0, g4 = 0,
             p = .50, n1, n2, n3, n4)

power.bcrd4r2(es = .25, alpha = .05, two.tailed = TRUE,
              rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
              rho2, rho3, rho4, omega3, omega4,
```

```
r21 = 0, r22 = 0, r2t3 = 0, r2t4 = 0, g4 = 0,
p = .50, n1, n2, n3, n4)
```

Arguments

cn1	marginal cost per level 1 unit in treatment and control conditions, e.g. c(10, 5).
cn2	marginal cost per level 2 unit in treatment and control conditions, e.g. c(50, 20).
cn3	marginal cost per level 3 unit.
cn4	marginal cost per level 4 unit.
cost	total cost or budget.
n1	average number of level 1 units per level 2 unit.
n2	average number of level 2 units per level 3 unit.
n3	average number of level 3 units per level 4 unit.
n4	number of level 4 units.
p	proportion of level 2 units in treatment condition.
n0	vector of starting values for n1, n2, n3, n4 (positional). Starting values are replaced with averages when sample sizes are constrained by bounds.
p0	starting value for p when rhots = 0 and p = NULL. Starting value is replaced with average when p is constrained by bounds.
power	statistical power (1 - β).
es	effect size (Cohen's d).
alpha	probability of type I error (α).
two.tailed	logical; TRUE for two-tailed hypothesis testing.
constrain	character; "cost", "power", or "mdes".
round	logical; TRUE for rounded COSA solution.
max.power	logical; TRUE for maximizing power instead of minimizing variance.
local.solver	subset of c("LBFGS", "SLSQP", "MMA", "COBYLA").
rhots	correlation between the treatment and the scoring variable. Specify rhots = 0 to obtain results equivalent to random assignment designs.
k1	left truncation point (in standard deviation units from full normal distribution mean), ignored when rhots is not NULL or dists = "uniform".
k2	right truncation point (in standard deviation units from full normal distribution mean), ignored when rhots is not NULL or dists = "uniform".
dists	distribution of the scoring variable; "normal" or "uniform". By default, dists = "normal" specification implies a truncated normal distribution with k1 = -6 and k2 = 6.
rho2	proportion of variance in the outcome between level 2 units (unconditional ICC2).
rho3	proportion of variance in the outcome between level 3 units (unconditional ICC3).
rho4	proportion of variance in the outcome between level 4 units (unconditional ICC4).
omega3	ratio of the treatment effect variance between level 3 units to the variance in the outcome between level 3 units.

omega4	ratio of the treatment effect variance between level 4 units to the variance in the outcome between level 4 units.
g4	number of covariates at level 4.
r21	proportion of level 1 variance in the outcome explained by level 1 covariates.
r22	proportion of level 1 variance in the outcome explained by level 2 covariates.
r2t3	proportion of treatment effect variance between level 3 units explained by level 3 covariates.
r2t4	proportion of treatment effect variance between level 4 units explained by level 4 covariates.

Value

parms	list of parameters used in the function.
df	degrees of freedom.
sse	standardized standard error.
cosa	constrained optimal sample allocation.
mdes	minimum detectable effect size and (1 - α)% confidence limits.
power	statistical power (1 - β)

Examples

```
# cost constrained - optimize n3 and n4
cosa.bcrd4r2(constrain = "cost", cost = 100000,
             cn1 = c(5, 3), cn2 = 10, cn3 = 30, cn4 = 50,
             es = .20, rho2 = .20, rho3 = .10, rho4 = .05,
             omega3 = .10, omega4 = .20,
             g4 = 1,r21 = .20, r22 = .20, r2t3 = .20, r2t4 = .20,
             p = .50, n1 = 25, n2 = 3, n3 = NULL, n4 = NULL)

# minimum detectable effect size
mdes.bcrd4r2(power = .80, rho2 = .20, rho3 = .10, rho4 = .05,
             omega3 = .10, omega4 = .20,
             r21 = .20, r22 = .30, r2t3 = .40, r2t4 = .50,
             g4 = 1, p = .50, n1 = 10, n2 = 3, n3 = 50.92, n4 = 13)

# statistical power
power.bcrd4r2(es = .20, rho2 = .20, rho3 = .10, rho4 = .05,
              omega3 = .10, omega4 = .20,
              r21 = .20, r22 = .30, r2t3 = .40, r2t4 = .50,
              g4 = 1, p = .50, n1 = 10, n2 = 3, n3 = 50.92, n4 = 13)
```

bcrd4r3 *Blocked Cluster-level Regression Discontinuity (Four-level Design, Discontinuity at Level 3)*

Description

Use `mdes.bcrd4r3()` to calculate minimum detectable effect size, `power.bcrd4r3()` to calculate statistical power, and `cosa.bcrd4r3()` for constrained optimal sample allocation.

Usage

```
cosa.bcrd4r3(cn1 = 0, cn2 = 0, cn3 = 0, cn4 = 0, cost = NULL,
             n1 = NULL, n2 = NULL, n3 = NULL, n4 = NULL, p = NULL,
             n0 = c(10, 3, 100, 5 + g4), p0 = .499, constrain = "power",
             round = TRUE, max.power = FALSE,
             local.solver = c("LBFGS", "SLSQP", "MMA", "COBYLA"),
             rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
             power = .80, es = .25, alpha = .05, two.tailed = TRUE,
             rho2, rho3, rho4, omega4,
             g4 = 0, r21 = 0, r22 = 0, r23 = 0, r2t4 = 0)

mdes.bcrd4r3(power = .80, alpha = .05, two.tailed = TRUE,
             rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
             rho2, rho3, rho4, omega4,
             r21 = 0, r22 = 0, r23 = 0, r2t4 = 0, g4 = 0,
             p = .50, n1, n2, n3, n4)

power.bcrd4r3(es = .25, alpha = .05, two.tailed = TRUE,
              rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
              rho2, rho3, rho4, omega4,
              r21 = 0, r22 = 0, r23 = 0, r2t4 = 0, g4 = 0,
              p = .50, n1, n2, n3, n4)
```

Arguments

`cn1`	marginal cost per level 1 unit in treatment and control conditions, e.g. `c(10, 5)`.
`cn2`	marginal cost per level 2 unit in treatment and control conditions, e.g. `c(50, 20)`.
`cn3`	marginal cost per level 3 unit in treatment and control conditions, e.g. `c(80, 50)`.
`cn4`	marginal cost per level 4 unit.
`cost`	total cost or budget.
`n1`	average number of level 1 units per level 2 unit.
`n2`	average number of level 2 units per level 3 unit.
`n3`	average number of level 3 units per level 4 unit.
`n4`	number of level 4 units.

p	proportion of level 3 units in treatment condition.
n0	vector of starting values for n1, n2, n3, n4 (positional). Starting values are replaced with averages when sample sizes are constrained by bounds.
p0	starting value for p when rhots = 0 and p = NULL. Starting value is replaced with average when p is constrained by bounds.
power	statistical power (1 - β).
es	effect size (Cohen's d).
alpha	probability of type I error (α).
two.tailed	logical; TRUE for two-tailed hypothesis testing.
constrain	character; "cost", "power", or "mdes".
round	logical; TRUE for rounded COSA solution.
max.power	logical; TRUE for maximizing power instead of minimizing variance.
local.solver	subset of c("LBFGS", "SLSQP", "MMA", "COBYLA").
rhots	correlation between the treatment and the scoring variable. Specify rhots = 0 to obtain results equivalent to random assignment designs.
k1	left truncation point (in standard deviation units from full normal distribution mean), ignored when rhots is not NULL or dists = "uniform".
k2	right truncation point (in standard deviation units from full normal distribution mean), ignored when rhots is not NULL or dists = "uniform".
dists	distribution of the scoring variable; "normal" or "uniform". By default, dists = "normal" specification implies a truncated normal distribution with k1 = -6 and k2 = 6.
rho2	proportion of variance in the outcome between level 2 units (unconditional ICC2).
rho3	proportion of variance in the outcome between level 3 units (unconditional ICC3).
rho4	proportion of variance in the outcome between level 4 units (unconditional ICC4).
omega4	ratio of the treatment effect variance between level 4 units to the variance in the outcome between level 4 units.
g4	number of covariates at level 4.
r21	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.
r23	proportion of level 3 variance in the outcome explained by level 3 covariates.
r2t4	proportion of treatment effect variance between level 4 units explained by level 4 covariates.

Value

parms	list of parameters used in the function.
df	degrees of freedom.
sse	standardized standard error.
cosa	constrained optimal sample allocation.
mdes	minimum detectable effect size and (1 - α)% confidence limits.
power	statistical power (1 - β)

Examples

```
# cost constrained - optimize n3 and n4
cosa.bcrd4r3(constrain = "cost", cost = 50000,
             cn1 = 5, cn2 = 10, cn3 = 30, cn4 = 50,
             es = .20, rho2 = .30, rho3 = .15, rho4 = .10, omega4 = .50,
             r21 = .50, r22 = .30, r23 = .20, r2t4 = .20,
             g4 = 1, p = .50, n1 = 25, n2 = 3, n3 = NULL, n4 = NULL)

# minimum detectable effect size
mdes.bcrd4r3(power = .80, rho2 = .20, rho3 = .10, rho4 = .05, omega4 = .20,
             r21 = .20, r22 = .30, r23 = .40, r2t4 = .50,
             g4 = 1, p = .50, n1 = 10, n2 = 3, n3 = 50.92, n4 = 13)

# statistical power
power.bcrd4r3(es = .20, rho2 = .20, rho3 = .10, rho4 = .05, omega4 = .20,
              r21 = .20, r22 = .30, r23 = .40, r2t4 = .50,
              g4 = 1, p = .50, n1 = 10, n2 = 3, n3 = 50.92, n4 = 13)
```

---

bird2r1 *Blocked Individual-level Regression Discontinuity (Two-level Design, Discontinuity at Level 1)*

Description

Use `mdes.bird2r1()` to calculate minimum detectable effect size, `power.bird2r1()` to calculate statistical power, and `cosa.bird2r1()` for constrained optimal sample allocation.

Usage

```
cosa.bird2r1(cn1 = 0, cn2 = 0, cost = NULL,
             n1 = NULL, n2 = NULL, p = NULL,
             n0 = c(10, 100 + g2), p0 = .499, constrain = "power",
             round = TRUE, max.power = FALSE,
             local.solver = c("LBFGS", "SLSQP", "MMA", "COBYLA"),
             rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
             power = .80, es = .25, alpha = .05, two.tailed = TRUE,
             rho2, omega2, g2 = 0, r21 = 0, r2t2 = 0)

mdes.bird2r1(power = .80, alpha = .05, two.tailed = TRUE,
             rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
             rho2, omega2, r21 = 0, r2t2 = 0,
             g2 = 0, p = .50, n1, n2)

power.bird2r1(es = .25, alpha = .05, two.tailed = TRUE,
              rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
              rho2, omega2, r21 = 0, r2t2 = 0,
              g2 = 0, p = .50, n1, n2)
```

Arguments

cn1	marginal cost per level 1 unit in treatment and control conditions, e.g. c(10, 5).
cn2	marginal cost per level 2 unit.
cost	total cost or budget.
n1	average number of level 1 units per level 2 units.
n2	number of level 2 units.
p	proportion of level 1 units in treatment condition.
n0	vector of starting values for n1, n2 (positional). Starting values are replaced with averages when sample sizes are constrained by bounds.
p0	starting value for p when rhots = 0 and p = NULL. Starting value is replaced with average when p is constrained by bounds.
power	statistical power (1 - β).
es	effect size (Cohen's d).
alpha	probability of type I error (α).
two.tailed	logical; TRUE for two-tailed hypothesis testing.
constrain	character; "cost", "power", or "mdes".
round	logical; TRUE for rounded COSA solution.
max.power	logical; TRUE for maximizing power instead of minimizing variance.
local.solver	subset of c("LBFGS", "SLSQP", "MMA", "COBYLA")
rhots	correlation between the treatment and the scoring variable. Specify rhots = 0 to obtain results equivalent to random assignment designs.
k1	left truncation point (in standard deviation units from full normal distribution mean), ignored when rhots is not NULL or dists = "uniform".
k2	right truncation point (in standard deviation units from full normal distribution mean), ignored when rhots is not NULL or dists = "uniform".
dists	character; distribution of the scoring variable, "normal" or "uniform". By default, dists = "normal" specification implies a truncated normal distribution with k1 = -6 and k2 = 6.
rho2	proportion of variance in the outcome between level 2 units (unconditional ICC2).
omega2	ratio of the treatment effect variance between level 2 units to the variance in the outcome between level 2 units.
g2	number of covariates at level 2.
r21	proportion of level 1 variance in the outcome explained by level 1 covariates.
r2t2	proportion of treatment effect variance between level 2 units explained by level 2 covariates.

Value

parms	list of parameters used in the function.
df	degrees of freedom.
sse	standardized standard error.
cosa	constrained optimal sample allocation.
mdes	minimum detectable effect size and (1 - α)% confidence limits.
power	statistical power (1 - β)

Examples

```
# cost constrained - optimize n1 and n2
cosa.bird2r1(constrain = "cost", cost = 5000,
             cn1 = c(5, 4), cn2 = 10,
             es = .20, rho2 = .20, omega2 = .20,
             r21 = .20, r2t2 = .20, g2 = 1,
             p = .50, n1 = NULL, n2 = NULL)

# minimum detectable effect size
mdes.bird2r1(power = .80, rho2 = .20, omega2 = .20,
             r21 = .20, r2t2 = .30, g2 = 1, p = .50,
             n1 = 26.96, n2 = 22)

# statistical power
power.bird2r1(es = .20, rho2 = .20, omega2 = .20,
              r21 = .20, r2t2 = .30, g2 = 1, p = .50,
              n1 = 26.96, n2 = 22)
```

---

bird3r1 *Blocked Individual-level Regression Discontinuity (Three-level Design, Discontinuity at Level 1)*

Description

Use `mdes.bird3r1()` to calculate minimum detectable effect size, `power.bird3r1()` to calculate statistical power, and `cosa.bird3r1()` for constrained optimal sample allocation.

Usage

```
cosa.bird3r1(cn1 = 0, cn2 = 0, cn3 = 0, cost = NULL,
             n1 = NULL, n2 = NULL, n3 = NULL, p = NULL,
             n0 = c(10, 3, 100 + g3), p0 = .499, constrain = "power",
             round = TRUE, max.power = FALSE,
             local.solver = c("LBFGS", "SLSQP", "MMA", "COBYLA"),
             rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
             power = .80, es = .25, alpha = .05, two.tailed = TRUE,
             rho2, rho3, omega2, omega3,
```

```
              g3 = 0, r21 = 0, r2t2 = 0, r2t3 = 0)

mdes.bird3r1(power = .80, alpha = .05, two.tailed = TRUE,
             rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
             rho2, rho3, omega2, omega3,
             r21 = 0, r2t2 = 0, r2t3 = 0, g3 = 0,
             p = .50, n1, n2, n3)

power.bird3r1(es = .25, alpha = .05, two.tailed = TRUE,
              rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
              rho2, rho3, omega2, omega3,
              r21 = 0, r2t2 = 0, r2t3 = 0, g3 = 0,
              p = .50, n1, n2, n3)
```

Arguments

cn1	marginal cost per level 1 unit in treatment and control conditions, e.g. c(10, 5).
cn2	marginal cost per level 2 unit.
cn3	marginal cost per level 3 unit.
cost	total cost or budget.
n1	average number of level 1 units per level 2 unit.
n2	average number of level 2 units per level 3 unit.
n3	number of level 3 units.
p	proportion of level 1 units in treatment condition.
n0	vector of starting values for n1, n2, n3 (positional). Starting values are replaced with averages when sample sizes are constrained by bounds.
p0	starting value for p when rhots = 0 and p = NULL. Starting value is replaced with average when p is constrained by bounds.
power	statistical power (1 - β).
es	effect size (Cohen's d).
alpha	probability of type I error (α).
two.tailed	logical; TRUE for two-tailed hypothesis testing.
constrain	character; "cost", "power", or "mdes".
round	logical; TRUE for rounded COSA solution.
max.power	logical; TRUE for maximizing power instead of minimizing variance.
local.solver	subset of c("LBFGS", "SLSQP", "MMA", "COBYLA")
rhots	correlation between the treatment and the scoring variable. Specify rhots = 0 to obtain results equivalent to random assignment designs.
k1	left truncation point (in standard deviation units from full normal distribution mean), ignored when rhots is not NULL or dists = "uniform".
k2	right truncation point (in standard deviation units from full normal distribution mean), ignored when rhots is not NULL or dists = "uniform".

dists	distribution of the scoring variable; "normal" or "uniform". By default, dists = "normal" specification implies a truncated normal distribution with k1 = -6 and k2 = 6.
rho2	proportion of variance in the outcome between level 2 units (unconditional ICC2).
rho3	proportion of variance in the outcome between level 3 units (unconditional ICC3).
omega2	ratio of the treatment effect variance between level 2 units to the variance in the outcome between level 2 units.
omega3	ratio of the treatment effect variance between level 3 units to the variance in the outcome between level 3 units.
g3	number of covariates at level 3.
r21	proportion of level 1 variance in the outcome explained by level 1 covariates.
r2t2	proportion of treatment effect variance between level 2 units explained by level 2 covariates.
r2t3	proportion of treatment effect variance between level 3 units explained by level 3 covariates.

Value

parms	list of parameters used in the function.
df	degrees of freedom.
sse	standardized standard error.
cosa	constrained optimal sample allocation.
mdes	minimum detectable effect size and (1 - α)% confidence limits.
power	statistical power (1 - β)

Examples

```
# cost constrained - optimize n2 and n3
cosa.bird3r1(constrain = "cost", cost = 5000,
             cn1 = c(5, 3), cn2 = 10, cn3 = 30,
             es = .20, rho2 = .20, rho3 = .10, omega2 = .20, omega3 = .10,
             r21 = .20, r2t2 = .20, r2t3 = .20, g3 = 1, p = .50,
             n1 = 25, n2 = NULL, n3 = NULL)

# minimum detectable effect size
mdes.bird3r1(power = .80, rho2 = .20, rho3 = .10, omega2 = .20, omega3 = .10,
             r21 = .20, r2t2 = .30, r2t3 = .40, g3 = 1, p = .50,
             n1 = 28.06, n2 = 3, n3 = 16)

# statistical power
power.bird3r1(es = .20, rho2 = .20, rho3 = .10, omega2 = .20, omega3 = .10,
              r21 = .20, r2t2 = .30, r2t3 = .40, g3 = 1, p = .50,
              n1 = 28.06, n2 = 3, n3 = 16)
```

bird4r1 *Blocked Individual-level Regression Discontinuity (Four-level Design, Discontinuity at Level 1)*

Description

Use `mdes.bird4r1()` to calculate minimum detectable effect size, `power.bird4r1()` to calculate statistical power, and `cosa.bird4r1()` for constrained optimal sample allocation.

Usage

```
cosa.bird4r1(cn1 = 0, cn2 = 0, cn3 = 0, cn4 = 0, cost = NULL,
             n1 = NULL, n2 = NULL, n3 = NULL, n4 = NULL, p = NULL,
             n0 = c(10, 3, 100, 5 + g4), p0 = .499, constrain = "power",
             round = TRUE, max.power = FALSE,
             local.solver = c("LBFGS", "SLSQP", "MMA", "COBYLA"),
             rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
             power = .80, es = .25, alpha = .05, two.tailed = TRUE,
             rho2, rho3, rho4, omega2, omega3, omega4,
             g4 = 0, r21 = 0, r2t2 = 0, r2t3 = 0, r2t4 = 0)

mdes.bird4r1(power = .80, alpha = .05, two.tailed = TRUE,
             rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
             rho2, rho3, rho4, omega2, omega3, omega4,
             r21 = 0, r2t2 = 0, r2t3 = 0, r2t4 = 0, g4 = 0,
             p = .50, n1, n2, n3, n4)

power.bird4r1(es = .25, alpha = .05, two.tailed = TRUE,
              rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
              rho2, rho3, rho4, omega2, omega3, omega4,
              r21 = 0, r2t2 = 0, r2t3 = 0, r2t4 = 0, g4 = 0,
              p = .50, n1, n2, n3, n4)
```

Arguments

`cn1`	marginal cost per level 1 unit in treatment and control conditions.
`cn2`	marginal cost per level 2 unit.
`cn3`	marginal cost per level 3 unit.
`cn4`	marginal cost per level 4 unit.
`cost`	total cost or budget.
`n1`	average number of level 1 units per level 2 unit.
`n2`	average number of level 2 units per level 3 unit.
`n3`	average number of level 3 units per level 4 unit.
`n4`	number of level 4 units.

p	proportion of level 1 units in treatment condition.
n0	vector of starting values for n1, n2, n3, n4 (positional). Starting values are replaced with averages when sample sizes are constrained by bounds.
p0	starting value for p when rhots = 0 and p = NULL. Starting value is replaced with average when p is constrained by bounds.
power	statistical power (1 - β).
es	effect size (Cohen's d).
alpha	probability of type I error (α).
two.tailed	logical; TRUE for two-tailed hypothesis testing.
constrain	character; "cost", "power", or "mdes".
round	logical; TRUE for rounded COSA solution.
max.power	logical; TRUE for maximizing power instead of minimizing variance.
local.solver	subset of c("LBFGS", "SLSQP", "MMA", "COBYLA").
rhots	correlation between the treatment and the scoring variable. Specify rhots = 0 to obtain results equivalent to random assignment designs.
k1	left truncation point (in standard deviation units from full normal distribution mean), ignored when rhots is not NULL or dists = "uniform".
k2	right truncation point (in standard deviation units from full normal distribution mean), ignored when rhots is not NULL or dists = "uniform".
dists	distribution of the scoring variable; "normal" or "uniform". By default, dists = "normal" specification implies a truncated normal distribution with k1 = -6 and k2 = 6.
rho2	proportion of variance in the outcome between level 2 units (unconditional ICC2).
rho3	proportion of variance in the outcome between level 3 units (unconditional ICC3).
rho4	proportion of variance in the outcome between level 4 units (unconditional ICC4).
omega2	ratio of the treatment effect variance between level 2 units to the variance in the outcome between level 2 units.
omega3	ratio of the treatment effect variance between level 3 units to the variance in the outcome between level 3 units.
omega4	ratio of the treatment effect variance between level 4 units to the variance in the outcome between level 4 units.
g4	number of covariates at level 4.
r21	proportion of level 1 variance in the outcome explained by level 1 covariates.
r2t2	proportion of treatment effect variance between level 2 units explained by level 2 covariates.
r2t3	proportion of treatment effect variance between level 3 units explained by level 3 covariates.
r2t4	proportion of treatment effect variance between level 4 units explained by level 4 covariates.

Value

parms	list of parameters used in the function.
df	degrees of freedom.
sse	standardized standard error.
cosa	constrained optimal sample allocation.
mdes	minimum detectable effect size and (1 - α)% confidence limits.
power	statistical power (1 - β)

Examples

```
# cost constrained - optimize n3 and n4
cosa.bird4r1(constrain = "cost", cost = 20000,
             cn1 = c(5, 3), cn2 = 10, cn3 = 30, cn4 = 50,
             es = .20, rho2 = .20, rho3 = .10, rho4 = .05,
             omega2 = .20, omega3 = .10, omega4 = .10,
             r21 = .20, r2t2 = .20, r2t3 = .20, r2t4 = .20,
             g4 = 1, p = .50, n1 = 25, n2 = 3, n3 = NULL, n4 = NULL)

# minimum detectable effect size
mdes.bird4r1(power = .80, rho2 = .20, rho3 = .10, rho4 = .05,
             omega2 = .20, omega3 = .10, omega4 = .10,
             r21 = .20, r2t2 = .30, r2t3 = .40, r2t4 = .50,
             g4 = 1, p = .50, n1 = 11.74, n2 = 3, n3 = 50, n4 = 6)

# statistical power
power.bird4r1(es = .20, rho2 = .20, rho3 = .10, rho4 = .05,
              omega2 = .20, omega3 = .10, omega4 = .10,
              r21 = .20, r2t2 = .30, r2t3 = .40, r2t4 = .50,
              g4 = 1, p = .50, n1 = 11.74, n2 = 3, n3 = 50, n4 = 6)
```

---

crd2r2 *Cluster-level Regression Discontinuity (Two-level Design, Discontinuity at Level 2, w/ or w/o Strata or Fixed Blocks)*

Description

Use `mdes.crd2r2()` to calculate minimum detectable effect size, `power.crd2r2()` to calculate statistical power, and `cosa.crd2r2()` for constrained optimal sample allocation.

If higher level strata or fixed blocks exist, use `mdes.bcrd3f2()` to calculate minimum detectable effect size, `power.bcrd3f2()` to calculate statistical power, and `cosa.bcrd3f2()` for constrained optimal sample allocation.

Usage

```
cosa.crd2r2(cn1 = 0, cn2 = 0, cost = NULL,
            n1 = NULL, n2 = NULL, p = NULL,
            n0 = c(10, 100 + g2), p0 = .499, constrain = "power",
            round = TRUE, max.power = FALSE,
            local.solver = c("LBFGS", "SLSQP", "MMA", "COBYLA"),
            rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
            power = .80, es = .25, alpha = .05, two.tailed = TRUE,
            rho2, g2 = 0, r21 = 0, r22 = 0)

cosa.bcrd3f2(cn1 = 0, cn2 = 0, cn3 = 0, cost = NULL,
            n1 = NULL, n2 = NULL, n3 = NULL, p = NULL,
            n0 = c(10, 100 + g2, 5), p0 = .499, constrain = "power",
            round = TRUE, max.power = FALSE,
            local.solver = c("LBFGS", "SLSQP", "MMA", "COBYLA"),
            rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
            power = .80, es = .25, alpha = .05, two.tailed = TRUE,
            rho2, g2 = 0, r21 = 0, r22 = 0)

mdes.crd2r2(power = .80, alpha = .05, two.tailed = TRUE,
            rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
            rho2, r21 = 0, r22 = 0, g2 = 0,
            p = .50, n1, n2)

mdes.bcrd3f2(power = .80, alpha = .05, two.tailed = TRUE,
            rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
            rho2, r21 = 0, r22 = 0, g2 = 0,
            p = .50, n1, n2, n3)

power.crd2r2(es = .25, alpha = .05, two.tailed = TRUE,
             rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
             rho2, r21 = 0, r22 = 0, g2 = 0,
             p = .50, n1, n2)

power.bcrd3f2(es = .25, alpha = .05, two.tailed = TRUE,
             rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
             rho2, r21 = 0, r22 = 0, g2 = 0,
             p = .50, n1, n2, n3)
```

Arguments

cn1	marginal cost per level 1 unit in treatment and control conditions, e.g. `c(10, 5)`.
cn2	marginal cost per level 2 unit in treatment and control conditions, e.g. `c(50, 30)`.
cn3	marginal cost per stratum or fixed block.
cost	total cost or budget.
n1	average number of level 1 units per level 2 unit.
n2	number of level 2 units (per stratum or block, if exists).

n3	number of stratum or fixed blocks.
p	proportion of level 2 units in treatment condition.
n0	vector of starting values for n1, n2 or n1, n2, n3 (positional). Starting values are replaced with averages when sample sizes are constrained by bounds.
p0	starting value for p when rhots = 0 and p = NULL. Starting value is replaced with average when p is constrained by bounds.
power	statistical power (1 - β).
es	effect size (Cohen's d).
alpha	probability of type I error (α).
two.tailed	logical; TRUE for two-tailed hypothesis testing.
constrain	character; "cost", "power", or "mdes".
round	logical; TRUE for rounded COSA solution.
max.power	logical; TRUE for maximizing power instead of minimizing variance.
local.solver	subset of c("LBFGS", "SLSQP", "MMA", "COBYLA")
rhots	correlation between the treatment and the scoring variable. Specify rhots = 0 to obtain results equivalent to random assignment designs.
k1	left truncation point (in standard deviation units from full normal distribution mean), ignored when rhots is not NULL or dists = "uniform".
k2	right truncation point (in standard deviation units from full normal distribution mean), ignored when rhots is not NULL or dists = "uniform".
dists	character; distribution of the scoring variable, "normal" or "uniform". By default, dists = "normal" specification implies a truncated normal distribution with k1 = -6 and k2 = 6.
rho2	proportion of variance in the outcome between level 2 units (unconditional ICC2).
g2	number of covariates at level 2.
r21	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

parms	list of parameters used in the function.
df	degrees of freedom.
sse	standardized standard error.
cosa	constrained optimal sample allocation.
mdes	minimum detectable effect size and (1 - α)% confidence limits.
power	statistical power (1 - β)

Examples

```
# cost constrained - optimize n1 and n2
cosa.crd2r2(constrain = "cost", cost = 50000,
            cn1 = 5, cn2 = c(10, 8),
            es = .20, rho2 = .20, r21 = .20, r22 = .30,
            g2 = 1, p = .50, n1 = NULL, n2 = NULL)

# minimum detectable effect size
mdes.crd2r2(power = .80, rho2 = .20, r21 = .20, r22 = .30,
            g2 = 1, p = .50, n1 = 16.04, n2 = 570)

# statistical power
power.crd2r2(es = .20, rho2 = .20, r21 = .20, r22 = .30,
             g2 = 1, p = .50, n1 = 16.04, n2 = 570)
```

---

crd3r3 *Cluster-level Regression Discontinuity (Three-level Design, Discontinuity at Level 3, w/ or w/o Strata or Fixed Blocks)*

Description

Use `mdes.crd3r3()` to calculate minimum detectable effect size, `power.crd3r3()` to calculate statistical power, and `cosa.crd3r3()` for constrained optimal sample allocation.

If higher level strata or fixed blocks exist, use `mdes.bcrd4f3()` to calculate minimum detectable effect size, `power.bcrd4f3()` to calculate statistical power, and `cosa.bcrd4f3()` for constrained optimal sample allocation.

Usage

```
cosa.crd3r3(cn1 = 0, cn2 = 0, cn3 = 0, cost = NULL,
            n1 = NULL, n2 = NULL, n3 = NULL, p = NULL,
            n0 = c(10, 3, 100 + g3), p0 = .499, constrain = "power",
            round = TRUE, max.power = FALSE,
            local.solver = c("LBFGS", "SLSQP", "MMA", "COBYLA"),
            rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
            power = .80, es = .25, alpha = .05, two.tailed = TRUE,
            rho2, rho3, g3 = 0, r21 = 0, r22 = 0, r23 = 0)

cosa.bcrd4f3(cn1 = 0, cn2 = 0, cn3 = 0, cn4 = 0, cost = NULL,
            n1 = NULL, n2 = NULL, n3 = NULL, n4 = NULL, p = NULL,
            n0 = c(10, 3, 100 + g3, 5), p0 = .499, constrain = "power",
            round = TRUE, max.power = FALSE,
            local.solver = c("LBFGS", "SLSQP", "MMA", "COBYLA"),
            rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
            power = .80, es = .25, alpha = .05, two.tailed = TRUE,
            rho2, rho3, g3 = 0, r21 = 0, r22 = 0, r23 = 0)
```

```
mdes.crd3r3(power = .80, alpha = .05, two.tailed = TRUE,
            rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
            rho2, rho3, r21 = 0, r22 = 0, r23 = 0, g3 = 0,
            p = .50, n1, n2, n3)

mdes.bcrd4f3(power = .80, alpha = .05, two.tailed = TRUE,
            rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
            rho2, rho3, r21 = 0, r22 = 0, r23 = 0, g3 = 0,
            p = .50, n1, n2, n3, n4)

power.crd3r3(es = .25, alpha = .05, two.tailed = TRUE,
             rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
             rho2, rho3, r21 = 0, r22 = 0, r23 = 0, g3 = 0,
             p = .50, n1, n2, n3)

power.bcrd4f3(es = .25, alpha = .05, two.tailed = TRUE,
             rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
             rho2, rho3, r21 = 0, r22 = 0, r23 = 0, g3 = 0,
             p = .50, n1, n2, n3, n4)
```

Arguments

cn1	marginal cost per level 1 unit in treatment and control conditions, e.g. c(10, 5).
cn2	marginal cost per level 2 unit in treatment and control conditions, e.g. c(50, 30).
cn3	marginal cost per level 3 unit in treatment and control conditions, e.g. c(80, 50).
cn4	marginal cost per stratum or fixed block.
cost	total cost or budget.
n1	average number of level 1 units per level 2 unit.
n2	average number of level 2 units per level 3 unit.
n3	number of level 3 units(per stratum or block, if exists).
n4	number of stratum or fixed blocks.
p	proportion of level 3 units in treatment condition.
n0	vector of starting values for n1, n2, n3 or n1, n2, n3, n4 (positional). Starting values are replaced with averages when sample sizes are constrained by bounds.
p0	starting value for p when rhots = 0 and p = NULL. Starting value is replaced with average when p is constrained by bounds.
power	statistical power (1 - β).
es	effect size (Cohen's d).
alpha	probability of type I error (α).
two.tailed	logical; TRUE for two-tailed hypothesis testing.
constrain	character; "cost", "power", or "mdes".
round	logical; TRUE for rounded COSA solution.

max.power	logical; TRUE for maximizing power instead of minimizing variance.
local.solver	subset of c("LBFGS", "SLSQP", "MMA", "COBYLA")
rhots	correlation between the treatment and the scoring variable. Specify rhots = 0 to obtain results equivalent to random assignment designs.
k1	left truncation point (in standard deviation units from full normal distribution mean), ignored when rhots is not NULL or dists = "uniform".
k2	right truncation point (in standard deviation units from full normal distribution mean), ignored when rhots is not NULL or dists = "uniform".
dists	distribution of the scoring variable; "normal" or "uniform". By default, dists = "normal" specification implies a truncated normal distribution with k1 = -6 and k2 = 6.
rho2	proportion of variance in the outcome between level 2 units (unconditional ICC2).
rho3	proportion of variance in the outcome between level 3 units (unconditional ICC3).
g3	number of covariates at level 3.
r21	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.
r23	proportion of level 3 variance in the outcome explained by level 3 covariates.

Value

parms	list of parameters used in the function.
df	degrees of freedom.
sse	standardized standard error.
cosa	constrained optimal sample allocation.
mdes	minimum detectable effect size and (1 - α)% confidence limits.
power	statistical power (1 - β)

Examples

```
# cost constrained - optimize n2 and n3
cosa.crd3r3(constrain = "cost", cost = 100000,
            cn1 = 5, cn2 = c(10, 5), cn3 = c(30, 10),
            es = .20, rho2 = .20, rho3 = .10,
            r21 = .20, r22 = .30, r23 = .40,
            g3 = 1, p = .50, n1 = 25, n2 = NULL, n3 = NULL)

# minimum detectable effect size
mdes.crd3r3(power = .80, rho2 = .20, rho3 = .10,
            r21 = .20, r22 = .30, r23 = .40,
            g3 = 1, p = .50, n1 = 19.39, n2 = 3, n3 = 300)

# statistical power
power.crd3r3(es = .20, rho2 = .20, rho3 = .10,
             r21 = .20, r22 = .30, r23 = .40,
             g3 = 1, p = .50, n1 = 19.39, n2 = 3, n3 = 300)
```

crd4r4 *Cluster-level Regression Discontinuity (Four-level Design, Discontinuity at Level 1)*

Description

Use `mdes.crd4r4()` to calculate minimum detectable effect size, `power.crd4r4()` to calculate statistical power, and `cosa.crd4r4()` for constrained optimal sample allocation.

Usage

```
cosa.crd4r4(cn1 = 0, cn2 = 0, cn3 = 0, cn4 = 0, cost = NULL,
           n1 = NULL, n2 = NULL, n3 = NULL, n4 = NULL, p = NULL,
           n0 = c(10, 3, 100, 5 + g4), p0 = .499, constrain = "power",
           round = TRUE, max.power = FALSE,
           local.solver = c("LBFGS", "SLSQP", "MMA", "COBYLA"),
           rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
           power = .80, es = .25, alpha = .05, two.tailed = TRUE,
           rho2, rho3, rho4, g4 = 0, r21 = 0, r22 = 0, r23 = 0, r24 = 0)

mdes.crd4r4(power = .80, alpha = .05, two.tailed = TRUE,
            rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
            rho2, rho3, rho4, r21 = 0, r22 = 0, r23 = 0, r24 = 0, g4 = 0,
            p = .50, n1, n2, n3, n4)

power.crd4r4(es = .25, alpha = .05, two.tailed = TRUE,
            rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
            rho2, rho3, rho4, r21 = 0, r22 = 0, r23 = 0, r24 = 0, g4 = 0,
            p = .50, n1, n2, n3, n4)
```

Arguments

`cn1`	marginal cost per level 1 unit in treatment and control conditions, e.g. `c(10, 5)`.
`cn2`	marginal cost per level 2 unit in treatment and control conditions, e.g. `c(50, 30)`.
`cn3`	marginal cost per level 3 unit in treatment and control conditions, e.g. `c(80, 50)`.
`cn4`	marginal cost per level 4 unit in treatment and control conditions, e.g. `c(100, 40)`.
`cost`	total cost or budget.
`n1`	average number of level 1 units per level 2 unit.
`n2`	average number of level 2 units per level 3 unit.
`n3`	average number of level 3 units per level 4 unit.
`n4`	number of level 4 units.
`p`	proportion of level 4 units in treatment condition.

n0	vector of starting values for n1, n2, n3, n4 (positional). Starting values are replaced with averages when sample sizes are constrained by bounds.
p0	starting value for p when rhots = 0 and p = NULL. Starting value is replaced with average when p is constrained by bounds.
power	statistical power (1 - β).
es	effect size (Cohen's d).
alpha	probability of type I error (α).
two.tailed	logical; TRUE for two-tailed hypothesis testing.
constrain	character; "cost", "power", or "mdes".
round	logical; TRUE for rounded COSA solution.
max.power	logical; TRUE for maximizing power instead of minimizing variance.
local.solver	subset of c("LBFGS", "SLSQP", "MMA", "COBYLA").
rhots	correlation between the treatment and the scoring variable. Specify rhots = 0 to obtain results equivalent to random assignment designs.
k1	left truncation point (in standard deviation units from full normal distribution mean), ignored when rhots is not NULL or dists = "uniform".
k2	right truncation point (in standard deviation units from full normal distribution mean), ignored when rhots is not NULL or dists = "uniform".
dists	distribution of the scoring variable; "normal" or "uniform". By default, dists = "normal" specification implies a truncated normal distribution with k1 = -6 and k2 = 6.
rho2	proportion of variance in the outcome between level 2 units (unconditional ICC2).
rho3	proportion of variance in the outcome between level 3 units (unconditional ICC3).
rho4	proportion of variance in the outcome between level 4 units (unconditional ICC4).
g4	number of covariates at level 4.
r21	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.
r23	proportion of level 3 variance in the outcome explained by level 3 covariates.
r24	proportion of level 4 variance in the outcome explained by level 4 covariates.

Value

parms	list of parameters used in the function.
df	degrees of freedom.
sse	standardized standard error.
cosa	constrained optimal sample allocation.
mdes	minimum detectable effect size and (1 - α)% confidence limits.
power	statistical power (1 - β)

Examples

```
# cost constrained - optimize n3 and n4
cosa.crd4r4(constrain = "cost", cost = 50000,
            cn1 = 2, cn2 = c(10, 8), cn3 = 20, cn4 = c(50,40),
            es = .20, rho2 = .20, rho3 = .10, rho4 = .05,
            r21 = .20, r22 = .30, r23 = .40, r24 = .50,
            g4 = 1, p = .50, n1 = 25, n2 = 3, n3 = NULL, n4 = NULL)

# minimum detectable effect size
mdes.crd4r4(power = .80, rho2 = .20, rho3 = .10, rho4 = .05,
            r21 = .20, r22 = .30, r23 = .40, r24 = .50,
            g4 = 1, p = .50, n1 = 10, n2 = 3, n3 = 51.7, n4 = 10)

# statistical power
power.crd4r4(es = .20, rho2 = .20, rho3 = .10, rho4 = .05,
             r21 = .20, r22 = .30, r23 = .40, r24 = .50,
             g4 = 1, p = .50, n1 = 10, n2 = 3, n3 = 51.7, n4 = 10)
```

`ird1r1` *Simple Individual-level Regression Discontinuity (w/ or w/o Strata or Fixed Blocks)*

Description

Use `mdes.ird1r1()` to calculate minimum detectable effect size and `power.ird1r1()` to calculate statistical power.

If higher level strata or fixed blocks exist, use `mdes.bird2f1()` to calculate minimum detectable effect size, `power.bird2f1()` to calculate statistical power, and `cosa.bird2f1()` for constrained optimal sample allocation.

Usage

```
cosa.bird2f1(cn1 = 0,  cn2 = 0, cost = NULL, n1 = NULL, n2 = NULL, p = NULL,
             n0 = c(400 + g1, 5), p0 = .499, constrain = "power",
             round = TRUE, max.power = FALSE,
             local.solver = c("LBFGS", "SLSQP", "MMA", "COBYLA"),
             rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
             power = .80, es = .25, alpha = .05, two.tailed = TRUE,
             g1 = 0, r21 = 0)

mdes.ird1r1(power = .80, alpha = .05, two.tailed = TRUE,
            rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
            r21 = 0, g1 = 0, p = .50, n1)

mdes.bird2f1(power = .80, alpha = .05, two.tailed = TRUE,
             rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
```

```
              r21 = 0, g1 = 0, p = .50, n1, n2)

power.ird1r1(es = .25, alpha = .05, two.tailed = TRUE,
             rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
             r21 = 0, g1 = 0, p = .50, n1)

power.bird2f1(es = .25, alpha = .05, two.tailed = TRUE,
              rhots = NULL, k1 = -6, k2 = 6, dists = "normal",
              r21 = 0, g1 = 0, p = .50, n1, n2)
```

Arguments

cn1	marginal cost per unit in treatment and control conditions, e.g. `c(10, 5)`.
cn2	marginal cost per stratum or fixed block.
cost	total cost or budget.
n1	sample size (per stratum or block, if exists).
n2	number of stratum or fixed blocks.
p	proportion of units in treatment condition.
n0	starting value for `n1` or `n1, n2`. Starting value is replaced with average when sample size is constrained by bounds.
p0	starting value for `p` when `rhots = 0` and `p = NULL`. Starting value is replaced with average when `p` is constrained by bounds.
power	statistical power (1 - β).
es	effect size (Cohen's d).
alpha	probability of type I error (α).
two.tailed	logical; `TRUE` for two-tailed hypothesis testing.
constrain	character; `"cost"`, `"power"`, or `"mdes"`.
round	logical; `TRUE` for rounded COSA solution.
max.power	logical; `TRUE` for maximizing power instead of minimizing variance.
local.solver	subset of `c("LBFGS", "SLSQP", "MMA", "COBYLA")`
rhots	correlation between the treatment and the scoring variable. Specify `rhots = 0` to obtain results equivalent to random assignment designs.
k1	left truncation point (in standard deviation units from full normal distribution mean), ignored when `rhots` is not `NULL` or `dists = "uniform"`.
k2	right truncation point (in standard deviation units from full normal distribution mean), ignored when `rhots` is not `NULL` or `dists = "uniform"`.
dists	character; distribution of the scoring variable, `"normal"` or `"uniform"`. By default, `dists = "normal"` specification implies a truncated normal distribution with `k1 = -6` and `k2 = 6`.
g1	number of covariates.
r21	proportion of variance in the outcome explained by covariates.

Value

parms	list of parameters used in the function.
df	degrees of freedom.
sse	standardized standard error.
cosa	constrained optimal sample allocation.
mdes	minimum detectable effect size and (1 - α)% confidence limits.
power	statistical power (1 - β)

Examples

```
# minimum detectable effect size
mdes.ird1r1(power = .80, r2 = .20, g = 1, p = .50, n = 500)

# statistical power
power.ird1r1(es = .373, r2 = .20, g = 1, p = .50, n = 500)
```

---

plot *Power and MDES Curves*

Description

Plots statistical power or minimum detectable effect size curves with (1-α)x100 % confidence interval for the design of interest.

Usage

```
## S3 method for class 'power'
plot(x, ypar = "mdes",  xpar = NULL,
          xlim = NULL, ylim = NULL,
          xlab = NULL, ylab = NULL,
          main = NULL, sub = NULL,
          locate = FALSE, benchmark = NULL, ...)

## S3 method for class 'mdes'
plot(x, ypar = "mdes",  xpar = NULL,
          xlim = NULL, ylim = NULL,
          xlab = NULL, ylab = NULL,
          main = NULL, sub = NULL,
          locate = FALSE, benchmark = NULL, ...)

## S3 method for class 'cosa'
plot(x, ypar = "mdes",  xpar = NULL,
          xlim = NULL, ylim = NULL,
          xlab = NULL, ylab = NULL,
          main = NULL, sub = NULL,
          locate = FALSE, benchmark = NULL, ...)
```

Arguments

x	an object returned from functions in **cosa** package.
ypar	character; "mdes" or "power" on y axis .
xpar	character; one of the sample sizes on x axis.
xlim	limits for xpar.
ylim	limits for ypar.
xlab	x axis label.
ylab	y axis label.
main	title for the plot.
sub	subtitle for the plot.
locate	logical; TRUE locates parameter values for design x on the plot.
benchmark	benchmark line.
...	other graphical parameters to pass to plot.new().

Examples

```
d1 <- mdes.bcrd3r2(rho2 = .10, rho3 = .20, omega3 = .30,
                   n1 = 20, n2 = 44, n3 = 50)
plot(d1, xpar = "n3", xlim = c(30, 100))
```

Index