Package ‘odr’

September 16, 2021

Type Package

Title Optimal Design and Statistical Power of Multilevel Randomized Trials

Version 1.2.1

Date 2021-9-15

Description Calculate the optimal sample allocation that produces the highest statistical power for experimental studies under a budget constraint, perform power analyses with and without accommodating cost structures of sampling, and calculate the relative efficiency between two sample allocations.

The references for the proposed methods are:

Language en-US

Depends R (>= 3.3.0), stats (>= 3.0.0), graphics (>= 3.0.0), base(>= 3.0.0)

License GPL-3

Encoding UTF-8

RoxygenNote 7.1.1

Suggests rmarkdown, knitr, markdown
This package is to help researchers design cost-efficient experimental studies assessing main treatment effects with adequate statistical precision by (a) solving optimal sample allocations, (b) comparing design precision and efficiency between different sample allocations, and (c) explicitly accommodating costs and budget in power analyses.
Details

The package covers seven types of experiments aiming to detect main and mediation effects. These experiments are individual randomized controlled trials (RCTs), two-, three-, and four-level cluster-randomized trials (CRTs), two-, three-, and four-level multisite randomized trials (MRTs), and two-level CRTs investigating mediation effects with group-level mediators. There are two categorical functions for each type of experiments and a uniform function for all types of experiments. The two categorical functions are 'od' and 'power'. The 'od' function can calculate the optimal sample allocation with and without constraint for each type of experiments. The 'power' function by default can calculate required budget (and required sample size) for desired power, minimum detectable effect size (MDES) under a fixed budget, statistical power under a fixed budget. The 'power' function also can perform conventional power analyses (e.g., required sample size, power, MDES calculation). The uniform function 're' (or 'rpe') is to compare the relative (precision and) efficiency between two designs with different sample allocations.

Author(s)

Zuchao Shen, Ben Kelcey
Maintainer: Zuchao Shen zuchao.shen@gmail.com

---

gen.design.pars

Generate optimal design parameters using ant colony optimization

Description

This function can generate a set of optimal design parameters based on given distributions of the rank of optimization target (or budget).

Usage

gen.design.pars(
  dist.mean,  # List of means - coordinates
  dist.rank,  # Rank of the archived values of objective function
  n.of.ants,  # Number of ants used in each iteration after the initialization of power analysis for calculating required budget, default value is 10.
  nl,         # Neighborhood of the search area
  q = 1e-04,  # Minimum detectable effect size
  n.of.archive = 100,  # Number of archives to store
  xi = 0.5  # Control parameter for the algorithm
)

Arguments

dist.mean   List of means - coordinates

dist.rank   Rank of the archived values of objective function

n.of.ants   Number of ants used in each iteration after the initialization of power analysis for calculating required budget, default value is 10.

nl          Neighborhood of the search area
od.1

q  Locality of the search (0,1), default is 0.0001.
n.of.archive  Size of the solution archive, default is 100.
xi  Convergence pressure (0, Inf), suggested: (0, 1), default is 0.5.

Value

Generated optimal design parameter value(s) (i.e., a matrix with n.of.ants rows and n.of.design.pars columns)

References


We thank Dr. Krzysztof Socha for providing us the original code (http://iridia.ulb.ac.be/supp/IridiaSupp2008-001/) for this function.

---

od.1  Optimal sample allocation calculation for single-level experiments detecting main effects

Description

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

Usage

```r
od.1(
p = NULL,
r12 = NULL,
c1 = NULL,
c1t = NULL,
m = NULL,
plots = TRUE,
plim = NULL,
varlim = NULL,
plab = NULL,
varlab = NULL,
vartitle = NULL,
verbose = TRUE
)
```
Arguments

- **p** The proportion of individuals to be assigned to treatment.
- **r12** The proportion of outcome variance explained by covariates.
- **c1** The cost of sampling one unit in control condition.
- **c1t** The cost of sampling one unit in treatment condition.
- **m** Total budget, default value is the total costs of sampling 60 individuals across treatment conditions.
- **plots** Logical, provide variance plots if TRUE, otherwise not; default value is TRUE.
- **plim** The plot range for p, default value is c(0, 1).
- **varlim** The plot range for variance, default value is c(0, 0.05).
- **plab** The plot label for p, default value is "Proportion of Individuals in Treatment: p".
- **varlab** The plot label for variance, default value is "Variance".
- **vartitle** The title of variance plot, default value is NULL.
- **verbose** Logical; print the value of p if TRUE, otherwise not; default value is TRUE.

Value

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

Examples

```r
# Unconstrained optimal design #---------
myod1 <- od.1(r12 = 0.5, c1 = 1, c1t = 5, varlim = c(0, 0.2))
myod1$out # output

# Constrained p, no calculation performed #---------
myod2 <- od.1(r12 = 0.5, c1 = 1, c1t = 5, varlim = c(0, 0.2), p = 0.5)
myod2$out

# Relative efficiency (RE)
myre <- re(od = myod1, subod= myod2)
myre$re # RE = 0.87

# When sampling costs are equal, a balanced design with p = 0.5 is the best #---------
myod3 <- od.1(r12 = 0.5, c1 = 1, c1t = 1, varlim = c(0, 0.2))
myod3$out # output
```
**Optimal sample allocation calculation for two-level CRTs detecting main effects**

**Description**

The optimal design of two-level cluster randomized trials (CRTs) detecting main effects is to calculate the optimal sample allocation that minimizes the variance of a treatment effect under a fixed budget, which is approximately the optimal sample allocation that maximizes statistical power under a fixed budget. The optimal design parameters include the level-1 sample size per level-2 unit ($n$) and the proportion of level-2 clusters/groups to be assigned to treatment ($p$). This function solves the optimal $n$ and/or $p$ with and without constraints.

**Usage**

```r
od.2(n = NULL, p = NULL, icc = NULL, r12 = NULL, r22 = NULL, c1 = NULL, c2 = NULL, c1t = NULL, c2t = NULL, m = NULL, plots = TRUE, plot.by = NULL, nlim = NULL, plim = NULL, varlim = NULL, nlab = NULL, plab = NULL, varlab = NULL, vartitle = NULL, verbose = TRUE)
```

**Arguments**

- **n**: The level-1 sample size per level-2 unit.
- **p**: The proportion of level-2 clusters/units to be assigned to treatment.
- **icc**: The unconditional intraclass correlation coefficient (ICC) in population or in each treatment condition.
- **r12**: The proportion of level-1 variance explained by covariates.
- **r22**: The proportion of level-2 variance explained by covariates.
The cost of sampling one level-1 unit in control condition.

The cost of sampling one level-2 unit in control condition.

The cost of sampling one level-1 unit in treatment condition.

The cost of sampling one level-2 unit in treatment condition.

Total budget, default value is the total costs of sampling 60 level-2 units across treatment conditions.

Logical, provide variance plots if TRUE, otherwise not; default value is TRUE.

Plot the variance by n and/or p; default value is plot.by = list(n = "n", p = "p").

The plot range for n, default value is c(2, 50).

The plot range for p, default value is c(0, 1).

The plot range for variance, default value is c(0, 0.05).

The plot label for n, default value is "Level-1 Sample Size: n".

The plot label for p, default value is "Proportion Level-2 Units in Treatment: p".

The plot label for variance, default value is "Variance".

The title of variance plot, default value is NULL.

Logical; print the values of n and p if TRUE, otherwise not; default value is TRUE.

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


Examples

# Unconstrained optimal design #--------
myod1 <- od.2(icc = 0.2, r12 = 0.5, r22 = 0.5, c1 = 1, c2 = 5, c1t = 1, c2t = 50, 
varlim = c(0.01, 0.02))
myod1$out # output

# Plot by p
myod1 <- od.2(icc = 0.2, r12 = 0.5, r22 = 0.5, c1 = 1, c2 = 5, c1t = 1, c2t = 50, 
varlim = c(0.01, 0.02), plot.by = list(p = 'p'))

# Constrained optimal design with n = 20 #-------
myod2 <- od.2(icc = 0.2, r12 = 0.5, r22 = 0.5, c1 = 1, c2 = 5, c1t = 1, c2t = 50, 
n = 20, varlim = c(0.005, 0.025))
myod2$out

# Relative efficiency (RE)
myre <- re(od = myod1, subod= myod2)
myre$re # RE = 0.88
# Constrained optimal design with p = 0.5 #---------
myod3 <- od.2(icc = 0.2, r12 = 0.5, r22 = 0.5, c1 = 1, c2 = 5, c1t = 1, c2t = 50, p = 0.5, varlim = c(0.005, 0.025))
myod3$out

# Relative efficiency (RE)
myre <- re(od = myod1, subod = myod3)
myre$re # RE = 0.90

# Constrained n and p, no calculation performed #---------
myod4 <- od.2(icc = 0.2, r12 = 0.5, r22 = 0.5, c1 = 1, c2 = 5, c1t = 1, c2t = 50, n = 20, p = 0.5, varlim = c(0.005, 0.025))
myod4$out

# Relative efficiency (RE)
myre <- re(od = myod1, subod = myod4)
myre$re # RE = 0.83

---

od.2.221

Optimal sample allocation calculation for two-level CRTs probing mediation effects with cluster-level mediators

Description

The optimal design of two-level cluster randomized trials (CRTs) probing mediation effects with cluster-level mediators, for the Sobel test, is to calculate the optimal sample allocation that minimizes the variance of a mediation effect under a fixed budget. For the joint significance test, it is to identify the optimal sample allocation that requires the minimum budget to achieve certain power level. The optimal design parameters include the level-1 sample size per level-2 unit (n) and the proportion of level-2 clusters/groups to be assigned to treatment (p). This function solves the optimal n and/or p with and without a constraint.

Usage

od.2.221(
  a = NULL,
  b = NULL,
  icc = NULL,
  c1 = NULL,
  c1t = NULL,
  c2 = NULL,
  c2t = NULL,
  m = NULL,
  r2m = 0,
  r12 = 0,
  r22 = 0,
  q.a = 0,
  q.b = 0,
  test = "joint",
)
n = NULL,
p = NULL,
iter = 100,
tol = 1e-11,
power.joint = 0.8,
d.p = c(0.1, 0.5),
d.n = c(1, 50),
sig.level = 0.05,
two.tailed = TRUE,
plots = TRUE,
plot.by = NULL,
n.rep = 1000,
nlim = c(0, 50),
plim = c(0, 1),
varlim = c(0, 0.05),
Jlim = c(4, 1e+05),
nlab = NULL,
plab = NULL,
varlab = NULL,
vartitle = NULL,
verbose = TRUE,
max.value = Inf,
max.iter = 150,
e = 1e-10,
n.of.ants = 10,
n.of.archive = 50,
q = 1e-04,
xi = 0.5
)

Arguments

a  The treatment effect on the mediator.
b  The within treatment correlation between the outcome and the mediator.
icc The unconditional intraclass correlation coefficient (ICC) in population or in each treatment condition.
c1  The cost of sampling one level-1 unit in control condition.
c1t  The cost of sampling one level-1 unit in treatment condition.
c2  The cost of sampling one level-2 unit in control condition.
c2t  The cost of sampling one level-2 unit in treatment condition.
m  Total budget.
r2m  The proportion of within treatment mediator variance explained by covariates.
r12  The proportion of within treatment individual-level outcome variance explained by covariates.
r22  The proportion of within treatment group-level outcome variance explained by covariates and the mediator.
q.a The number of covariates in the outcome model (except the treatment indicator and the mediator).
q.b The number of covariates in the mediator model (except the treatment indicator).
test The type of test will be used to detect mediation effects. Default is the joint significance test (i.e., test = "joint"). Other choices are the Sobel test and Monte Carlo confidence interval test by specifying the argument as test = "sobel" or test = "mcci".

n The level-1 sample size per level-2 unit.
p The proportion of level-2 clusters/units to be assigned to treatment.
iter number of iteration used for solving roots in Sobel test.
tol convergence tolerance.

power.joint Statistical power specified for the joint significance test, default is .80.
d.p The initial sampling domains for p. Default is c(0.03, 0.97).
d.n The initial sampling domain for n. Default is c(0.5, 500).
sig.level Significance level or type I error rate, default value is 0.05.
two.tailed Logical; two-tailed tests if TRUE, otherwise one-tailed tests; default value is TRUE.

plots Logical, provide variance plots if TRUE, otherwise not; default value is TRUE.
plot.by Plot the variance by n and/or p; default value is plot.by = list(n = "n", p = "p").
n.rep Number of replications in MCCI power calculation.
nlim The plot range for n, default value is c(2, 50).
pelim The plot range for p, default value is c(0, 1).
varlim The plot range for variance, default value is c(0, 0.05).
Jlim The range for J to search for a numerical solution. Default is c(4, 10e4).
nlab The plot label for n, default value is "Level-1 Sample Size: n".
pplab The plot label for p, default value is "Proportion Level-2 Units in Treatment: p".
varlab The plot label for variance, default value is "Variance".
vartitle The title of variance plot, default value is NULL.
verbose Print out evaluation process if TRUE, default is TRUE.

max.value Maximal value of optimization when used as the stopping criterion. Default is infinite.
max.iter Maximal number of function evaluations when used as the stopping criterion.
e Maximum error value used when solution quality used as the stopping criterion, default is 1e-10.
n.of.ants Number of ants used in each iteration after the initialization of power analysis for calculating required budget, default value is 10.

n.of.archive Size of the solution archive, default is 100.
q Locality of the search (0,1), default is 0.0001.
xi Convergence pressure (0, Inf), suggested: (0, 1), default is 0.5.
Value

Unconstrained or constrained optimal sample allocation (n and p). The function also returns the variance of a mediation effect or statistical power, function name, design type, and parameters used in the calculation.

od.2m

Optimal sample allocation calculation for two-level MRTs detecting main effects

Description

The optimal design of two-level multisite randomized trials (MRTs) detecting main effects is to calculate the sample allocation that minimizes the variance of a treatment effect under a fixed budget, which is approximately the optimal sample allocation that maximizes statistical power under a fixed budget. The optimal design parameters include the level-one sample size per site (n) and the proportion of level-one unit to be assigned to treatment (p). This function solves the optimal n and/or p with and without a constraint.

Usage

od.2m(
  n = NULL,
  p = NULL,
  icc = NULL,
  r12 = NULL,
  r22m = NULL,
  c1 = NULL,
  c2 = NULL,
  c1t = NULL,
  omega = NULL,
  m = NULL,
  plots = TRUE,
  plot.by = NULL,
  nlim = NULL,
  plim = NULL,
  varlim = NULL,
  nlab = NULL,
  plab = NULL,
  varlab = NULL,
  vartitle = NULL,
  verbose = TRUE,
  iter = 100,
  tol = 1e-10
)

Arguments

- **n**: The level-1 sample size per level-2 unit.
- **p**: The proportion of level-4 clusters/units to be assigned to treatment.
- **icc**: The unconditional intraclass correlation coefficient (ICC) in population or in each treatment condition.
- **r12**: The proportion of level-1 variance explained by covariates.
- **r22m**: The proportion of variance of site-specific treatment effect explained by covariates.
- **c1**: The cost of sampling one level-1 unit in control condition.
- **c2**: The cost of sampling one level-2 unit in control condition.
- **c1t**: The cost of sampling one level-1 unit in treatment condition.
- **omega**: The standardized variance of site-specific treatment effect.
- **m**: Total budget, default is the total costs of sampling 60 sites.
- **plots**: Logical, provide variance plots if TRUE, otherwise not; default value is TRUE.
- **plot.by**: Plot the variance by n and/or p; default value is plot.by = list(n = "n", p = "p").
- **nlim**: The plot range for n, default value is c(2, 50).
- **plim**: The plot range for p, default value is c(0, 1).
- **varlim**: The plot range for variance, default value is c(0, 0.05).
- **nlab**: The plot label for n, default value is "Level-1 Sample Size: n".
- **plab**: The plot label for p, default value is "Proportion Level-1 Units in Treatment: p".
- **varlab**: The plot label for variance, default value is "Variance".
- **vartitle**: The title of variance plot, default value is NULL.
- **verbose**: Logical; print the values of n and p if TRUE, otherwise not; default value is TRUE.
- **iter**: Number of iterations; default value is 100.
- **tol**: Tolerance for convergence; default value is 1e-10.

Value

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

References

Examples

# Unconstrained optimal design #---------
myod1 <- od.2m(icc = 0.2, omega = 0.02, r12 = 0.5, r22m = 0.5, 
c1 = 1, c2 = 10, c1t = 10, 
varlim = c(0, 0.005))
myod1$out # n = 20, p = 0.37
# Plots by p
myod1 <- od.2m(icc = 0.2, omega = 0.02, 
r12 = 0.5, r22m = 0.5, 
c1 = 1, c2 = 10, c1t = 10, 
varlim = c(0, 0.005), plot.by = list(p = 'p'))

# Constrained optimal design with p = 0.5 #---------
myod2 <- od.2m(icc = 0.2, omega = 0.02, 
r12 = 0.5, r22m = 0.5, 
c1 = 1, c2 = 10, c1t = 10, 
varlim = c(0, 0.005), p = 0.5)
myod2$out
# Relative efficiency (RE)
myre <- re(od = myod1, subod= myod2)
myre$re # RE = 0.86

# Constrained optimal design with n = 5 #---------
myod3 <- od.2m(icc = 0.2, omega = 0.02, 
r12 = 0.5, r22m = 0.5, c1 = 1, c2 = 10, 
c1t = 10, varlim = c(0, 0.005), n = 5)
myod3$out
# Relative efficiency (RE)
myre <- re(od = myod1, subod= myod3)
myre$re # RE = 0.79

# Constrained n and p, no calculation performed #---------
myod4 <- od.2m(icc = 0.2, omega = 0.02, r12 = 0.5, r22m = 0.5, 
c1 = 1, c2 = 10, c1t = 10, 
varlim = c(0, 0.005), p = 0.5, n = 10)
myod4$out
# Relative efficiency (RE)
myre <- re(od = myod1, subod= myod4)
myre$re # RE = 0.84

---

**od.3**

*Optimal sample allocation calculation for three-level CRTs detecting main effects*

---

**Description**

The optimal design of three-level cluster randomized trials (CRTs) is to calculate the optimal sample allocation that minimizes the variance of treatment effect under fixed budget, which is approximately the optimal sample allocation that maximizes statistical power under a fixed budget. The
optimal design parameters include the level-1 sample size per level-2 unit \( (n) \), the level-2 sample size per level-3 unit \( (J) \), and the proportion of level-3 clusters/groups to be assigned to treatment \( (p) \). This function solves the optimal \( n \), \( J \) and/or \( p \) with and without constraints.

Usage

\[
\text{od.3}(n = \text{NULL}, J = \text{NULL}, p = \text{NULL}, icc2 = \text{NULL}, icc3 = \text{NULL}, r12 = \text{NULL}, r22 = \text{NULL}, r32 = \text{NULL}, c1 = \text{NULL}, c2 = \text{NULL}, c3 = \text{NULL}, c1t = \text{NULL}, c2t = \text{NULL}, c3t = \text{NULL}, m = \text{NULL}, plots = \text{TRUE}, plot.by = \text{NULL}, nlim = \text{NULL}, Jlim = \text{NULL}, plim = \text{NULL}, varlim = \text{NULL}, nlab = \text{NULL}, Jlab = \text{NULL}, plab = \text{NULL}, varlab = \text{NULL}, vartitle = \text{NULL}, verbose = \text{TRUE}, \text{iter} = 100, \text{tol} = 1e-10)
\]

Arguments

- \( n \) The level-1 sample size per level-2 unit.
- \( J \) The level-2 sample size per level-3 unit.
- \( p \) The proportion of level-3 clusters/units assigned to treatment.
- \( \text{icc2} \) The unconditional intraclass correlation coefficient (ICC) at level 2.
- \( \text{icc3} \) The unconditional intraclass correlation coefficient (ICC) at level 3.
- \( \text{r12} \) The proportion of level-1 variance explained by covariates.
- \( \text{r22} \) The proportion of level-2 variance explained by covariates.
The proportion of level-3 variance explained by covariates.
c1 The cost of sampling one level-1 unit in control condition.
c2 The cost of sampling one level-2 unit in control condition.
c3 The cost of sampling one level-3 unit in control condition.
c1t The cost of sampling one level-1 unit in treatment condition.
c2t The cost of sampling one level-2 unit in treatment condition.
c3t The cost of sampling one level-3 unit in treatment condition.
m Total budget, default is the total costs of sampling 60 level-3 units across treatment conditions.
plot.s Logical, provide variance plots if TRUE, otherwise not; default value is TRUE.
plot.by Plot the variance by n, J and/or p; default is plot.by = list(n = "n", J = "J", p = "p").
nlim The plot range for n, default value is c(2, 50).
Jlim The plot range for J, default value is c(2, 50).
plim The plot range for p, default value is c(0, 1).
varlim The plot range for variance, default value is c(0, 0.05).
nlab The plot label for n, default value is "Level-1 Sample Size: n".
Jlab The plot label for J, default value is "Level-2 Sample Size: J".
plab The plot label for p, default is "Proportion Level-3 Units in Treatment: p".
varlab The plot label for variance, default value is "Variance".
vartitle The title of variance plot, default value is NULL.
verbose Logical; print the values of n, J, and p if TRUE, otherwise not; default is TRUE.
iter Number of iterations; default value is 100.
tol Tolerance for convergence; default value is 1e-10.

Value

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

References


Examples

# Unconstrained optimal design #--------
myod1 <- od.3(icc2 = 0.2, icc3 = 0.1, r12 = 0.5, r22 = 0.5, r32 = 0.5,
c1 = 1, c2 = 5, c3 = 25, c1t = 1, c2t = 50, c3t = 250,
varlim = c(0.005, 0.025))
myod1$out # output
# Plots by p and J
od.3m

Optimal sample allocation calculation for three-level MRTs detecting main effects

Description

The optimal design of three-level multisite randomized trials (MRTs) is to calculate the optimal sample allocation that minimizes the variance of treatment effect under fixed budget, which is approximately the optimal sample allocation that maximizes statistical power under a fixed budget. The optimal design parameters include the level-1 sample size per level-2 unit (n), the level-2 sample size per level-3 unit (J), and the proportion of level-2 unit to be assigned to treatment (p). This function solves the optimal n, J and/or p with and without constraints.

Usage

od.3m(
  n = NULL,
J = NULL,
p = NULL,
icc2 = NULL,
icc3 = NULL,
r12 = NULL,
r22 = NULL,
r32m = NULL,
c1 = NULL,
c2 = NULL,
c3 = NULL,
c1t = NULL,
c2t = NULL,
omega = NULL,
m = NULL,
plots = TRUE,
plot.by = NULL,
nlim = NULL,
Jlim = NULL,
plim = NULL,
varlim = NULL,
nlab = NULL,
Jlab = NULL,
plab = NULL,
varlab = NULL,
vartitle = NULL,
verbose = TRUE,
iter = 100,
tol = 1e-10
)

Arguments

n  The level-1 sample size per level-2 unit.
J  The level-2 sample size per level-3 unit.
p  The proportion of level-4 clusters/units to be assigned to treatment.
icc2 The unconditional intraclass correlation coefficient (ICC) at level 2.
icc3 The unconditional intraclass correlation coefficient (ICC) at level 3.
r12 The proportion of level-1 variance explained by covariates.
r22 The proportion of level-2 variance explained by covariates.
r32m The proportion of variance of site-specific treatment effect explained by covariates.
c1  The cost of sampling one level-1 unit in control condition.
c2  The cost of sampling one level-2 unit in control condition.
c3  The cost of sampling one level-3 unit in control condition.
c1t The cost of sampling one level-1 unit in treatment condition.
c2t  The cost of sampling one level-2 unit in treatment condition.
omega  The standardized variance of site-specific treatment effect.
m  Total budget, default is the total costs of sampling 60 level-3 units.
plots  Logical, provide variance plots if TRUE, otherwise not; default value is TRUE.
plot.by  Plot the variance by n, J and/or p; default value is plot.by = list(n = "n", J = "J", p = "p").
nlim  The plot range for n, default value is c(2, 50).
Jlim  The plot range for J, default value is c(2, 50).
plim  The plot range for p, default value is c(0, 1).
varlim  The plot range for variance, default value is c(0, 0.05).
nlab  The plot label for n, default value is "Level-1 Sample Size: n".
Jlab  The plot label for J, default value is "Level-2 Sample Size: J".
plab  The plot label for p, default value is "Proportion Level-2 Units in Treatment: p".
varlab  The plot label for variance, default value is "Variance".
vartitle  The title of variance plot, default value is NULL.
verbose  Logical; print the values of n, J, and p if TRUE, otherwise not; default value is TRUE.
iter  Number of iterations; default value is 100.
tol  Tolerance for convergence; default value is 1e-10.

Value
Unconstrained or constrained optimal sample allocation (n, J, and p). The function also returns the variance of the treatment effect, function name, design type, and parameters used in the calculation.

References

Examples
# Unconstrained optimal design #--------
myod1 <- od.3m(iccc2 = 0.2, icc3 = 0.1, omega = 0.02,
               r12 = 0.5, r22 = 0.5, r32m = 0.5,
               c1 = 1, c2 = 5,
               c1t = 1, c2t = 200, c3 = 200,
               varlim = c(0, 0.005))
myod1$out # output
# Plots by p and J
myod1 <- od.3m(iccc2 = 0.2, icc3 = 0.1, omega = 0.02,
               r12 = 0.5, r22 = 0.5, r32m = 0.5,
               c1 = 1, c2 = 5,
               c1t = 1, c2t = 200, c3 = 200,
               varlim = c(0, 0.005), plot.by = list(p = 'p', J = 'J'))
od.4

Optimal sample allocation calculation for four-level CRTs detecting main effects

Description

The optimal design of four-level cluster randomized trials (CRTs) is to calculate the optimal sample allocation that minimizes the variance of treatment effect under fixed budget, which is approximately the optimal sample allocation that maximizes statistical power under a fixed budget. The optimal design parameters include the level-1 sample size per level-2 unit (n), the level-2 sample size per level-3 unit (J), the level-3 sample size per level-4 unit (K), and the proportion of level-4 clusters/groups to be assigned to treatment (p). This function solves the optimal n, J, K and/or p with and without constraints.
Usage

od.4(
  n = NULL,
  J = NULL,
  K = NULL,
  p = NULL,
  icc2 = NULL,
  icc3 = NULL,
  icc4 = NULL,
  r12 = NULL,
  r22 = NULL,
  r32 = NULL,
  r42 = NULL,
  c1 = NULL,
  c2 = NULL,
  c3 = NULL,
  c4 = NULL,
  c1t = NULL,
  c2t = NULL,
  c3t = NULL,
  c4t = NULL,
  m = NULL,
  plots = TRUE,
  plot.by = NULL,
  nlim = NULL,
  Jlim = NULL,
  Klim = NULL,
  plim = NULL,
  varlim = NULL,
  nlab = NULL,
  Jlab = NULL,
  Klab = NULL,
  plab = NULL,
  varlab = NULL,
  vartitle = NULL,
  verbose = TRUE,
  iter = 100,
  tol = 1e-10
)

Arguments

n  The level-1 sample size per level-2 unit.
J  The level-2 sample size per level-3 unit.
K  The level-3 sample size per level-4 unit.
p  The proportion of level-4 clusters/units to be assigned to treatment.
icc2 The unconditional intraclass correlation coefficient (ICC) at level 2.
The unconditional intraclass correlation coefficient (ICC) at level 3.

The unconditional intraclass correlation coefficient (ICC) at level 4.

The proportion of level-1 variance explained by covariates.

The proportion of level-2 variance explained by covariates.

The proportion of level-3 variance explained by covariates.

The proportion of level-4 variance explained by covariates.

The cost of sampling one level-1 unit in control condition.

The cost of sampling one level-2 unit in control condition.

The cost of sampling one level-3 unit in control condition.

The cost of sampling one level-4 unit in control condition.

The cost of sampling one level-1 unit in treatment condition.

The cost of sampling one level-2 unit in treatment condition.

The cost of sampling one level-3 unit in treatment condition.

The cost of sampling one level-4 unit in treatment condition.

Total budget, default value is the total costs of sampling 60 level-4 units across treatment conditions.

Logical, provide variance plots if TRUE, otherwise not; default value is TRUE.

Plot the variance by n, J, K and/or p; default value is plot.by = list(n = "n", J = "J", K = "K", p = "p").

The plot range for n, default value is c(2, 50).

The plot range for J, default value is c(2, 50).

The plot range for K, default value is c(2, 50).

The plot range for p, default value is c(0, 1).

The plot range for variance, default value is c(0, 0.05).

The plot label for n, default value is "Level-1 Sample Size: n".

The plot label for J, default value is "Level-2 Sample Size: J".

The plot label for K, default value is "Level-3 Sample Size: K".

The plot label for p, default value is "Proportion Level-4 Units in Treatment: p".

The plot label for variance, default value is "Variance".

The title of variance plot, default value is NULL.

Logical; print the values of n, J, K, and p if TRUE, otherwise not; default value is TRUE.

Number of iterations; default value is 100.

Tolerance for convergence; default value is 1e-10.

Value

Unconstrained or constrained optimal sample allocation (n, J, K, and p). The function also returns the variance of the treatment effect, function name, design type, and parameters used in the calculation.
Examples

# Unconstrained optimal design #---------
myod1 <- od.4(icc2 = 0.2, icc3 = 0.1, icc4 = 0.05,
      r12 = 0.5, r22 = 0.5, r32 = 0.5, r42 = 0.5,
      c1 = 1, c2 = 5, c3 = 25, c4 = 125,
      c1t = 1, c2t = 50, c3t = 250, c4t = 2500,
      varlim = c(0, 0.01))
myod1$out # output

# Plots by p and K
myod1 <- od.4(icc2 = 0.2, icc3 = 0.1, icc4 = 0.05,
      r12 = 0.5, r22 = 0.5, r32 = 0.5, r42 = 0.5,
      c1 = 1, c2 = 5, c3 = 25, c4 = 125,
      c1t = 1, c2t = 50, c3t = 250, c4t = 2500,
      varlim = c(0, 0.01), plot.by = list(p = 'p', K = 'K'))

# Constrained optimal design with p = 0.5 #--------
myod2 <- od.4(icc2 = 0.2, icc3 = 0.1, icc4 = 0.05, p = 0.5,
      r12 = 0.5, r22 = 0.5, r32 = 0.5, r42 = 0.5,
      c1 = 1, c2 = 5, c3 = 25, c4 = 125,
      c1t = 1, c2t = 50, c3t = 250, c4t = 2500,
      varlim = c(0, 0.01))
myod2$out

# Relative efficiency (RE)
myre <- re(od = myod1, subod= myod2)
myre$re # RE = 0.78

# Constrained optimal design with K = 20 #--------
myod3 <- od.4(icc2 = 0.2, icc3 = 0.1, icc4 = 0.05, K = 20,
      r12 = 0.5, r22 = 0.5, r32 = 0.5, r42 = 0.5,
      c1 = 1, c2 = 5, c3 = 25, c4 = 125,
      c1t = 1, c2t = 50, c3t = 250, c4t = 2500,
      varlim = c(0, 0.01))
myod3$out

# Relative efficiency (RE)
myre <- re(od = myod1, subod= myod3)
myre$re # RE = 0.67

# Constrained n, J, K and p, no calculation performed #--------
myod4 <- od.4(icc2 = 0.2, icc3 = 0.1, icc4 = 0.05,
      r12 = 0.5, n = 10, J = 10, K = 20, p = 0.5,
      r22 = 0.5, r32 = 0.5, r42 = 0.5,
      c1 = 1, c2 = 5, c3 = 25, c4 = 125,
      c1t = 1, c2t = 50, c3t = 250, c4t = 2500,
      varlim = c(0, 0.01))
myod4$out

# Relative efficiency (RE)
myre <- re(od = myod1, subod= myod4)
myre$re # RE = 0.27
**od.4m**

*Optimal sample allocation calculation for four-level MRTs detecting main effects*

**Description**

The optimal design of four-level multisite randomized trials (MRTs) is to calculate the optimal sample allocation that minimizes the variance of treatment effect under fixed budget, which is approximately the optimal sample allocation that maximizes statistical power under a fixed budget. The optimal design parameters include the level-1 sample size per level-2 unit (n), the level-2 sample size per level-3 unit (J), the level-3 sample size per level-4 unit (K), and the proportion of level-3 units to be assigned to treatment (p). This function solves the optimal n, J, K and/or p with and without constraints.

**Usage**

```r
od.4m(
  n = NULL,
  J = NULL,
  K = NULL,
  p = NULL,
  icc2 = NULL,
  icc3 = NULL,
  icc4 = NULL,
  r12 = NULL,
  r22 = NULL,
  r32 = NULL,
  r42m = NULL,
  c1 = NULL,
  c2 = NULL,
  c3 = NULL,
  c4 = NULL,
  c1t = NULL,
  c2t = NULL,
  c3t = NULL,
  omega = NULL,
  m = NULL,
  plots = TRUE,
  plot.by = NULL,
  nlim = NULL,
  Jlim = NULL,
  Klim = NULL,
  plim = NULL,
  varlim = NULL,
  nlab = NULL,
  Jlab = NULL,
  Klab = NULL,
)```

Arguments

- **n**: The level-1 sample size per level-2 unit.
- **J**: The level-2 sample size per level-3 unit.
- **K**: The level-3 sample size per level-4 unit.
- **p**: The proportion of level-3 units to be assigned to treatment.
- **icc2**: The unconditional intraclass correlation coefficient (ICC) at level 2.
- **icc3**: The unconditional intraclass correlation coefficient (ICC) at level 3.
- **icc4**: The unconditional intraclass correlation coefficient (ICC) at level 4.
- **r12**: The proportion of level-1 variance explained by covariates.
- **r22**: The proportion of level-2 variance explained by covariates.
- **r32**: The proportion of level-3 variance explained by covariates.
- **r42m**: The proportion of variance of site-specific treatment effect explained by covariates.
- **c1**: The cost of sampling one level-1 unit in control condition.
- **c2**: The cost of sampling one level-2 unit in control condition.
- **c3**: The cost of sampling one level-3 unit in control condition.
- **c4**: The cost of sampling one level-4 unit.
- **c1t**: The cost of sampling one level-1 unit in treatment condition.
- **c2t**: The cost of sampling one level-2 unit in treatment condition.
- **c3t**: The cost of sampling one level-3 unit in treatment condition.
- **omega**: The standardized variance of site-specific treatment effect.
- **m**: Total budget, default is the total costs of sampling 60 level-4 units.
- **plots**: Logical, provide variance plots if TRUE, otherwise not; default value is TRUE.
- **plot.by**: Plot the variance by n, J, K and/or p; default value is plot.by = list(n = "n", J = "J", K = "K", p = "p").
- **nlim**: The plot range for n, default value is c(2, 50).
- **Jlim**: The plot range for J, default value is c(2, 50).
- **Klim**: The plot range for K, default value is c(2, 50).
- **plim**: The plot range for p, default value is c(0, 1).
- **varlim**: The plot range for variance, default value is c(0, 0.05).
- **nlab**: The plot label for n, default value is "Level-1 Sample Size: n".
The plot label for J, default value is "Level-2 Sample Size: J".

The plot label for K, default value is "Level-3 Sample Size: K".

The plot label for p, default value is "Proportion Level-3 Units in Treatment: p".

The plot label for variance, default value is "Variance".

The title of variance plot, default value is NULL.

Logical; print the values of n, J, K, and p if TRUE, otherwise not; default value is TRUE.

Number of iterations; default value is 100.

Tolerance for convergence; default value is 1e-10.

Unconstrained or constrained optimal sample allocation (n, J, K, and p). The function also returns the variance of the treatment effect, function name, design type, and parameters used in the calculation.

Examples

# Unconstrained optimal design #--------
myod1 <- od.4m(icc2 = 0.2, icc3 = 0.1, icc4 = 0.05, omega = 0.02,
    r12 = 0.5, r22 = 0.5, r32 = 0.5, r42m = 0.5,
    c1 = 1, c2 = 5, c3 = 25,
    c1t = 1, c2t = 50, c3t = 250, c4 = 500,
    varlim = c(0, 0.005))
myod1$out # output

# Plots by p and K
myod1 <- od.4m(icc2 = 0.2, icc3 = 0.1, icc4 = 0.05, omega = 0.02,
    r12 = 0.5, r22 = 0.5, r32 = 0.5, r42m = 0.5,
    c1 = 1, c2 = 5, c3 = 25,
    c1t = 1, c2t = 50, c3t = 250, c4 = 500,
    varlim = c(0, 0.005), plot.by = list(p = 'p', K = 'K'))

# Constrained optimal design with p = 0.5 #--------
myod2 <- od.4m(icc2 = 0.2, icc3 = 0.1, icc4 = 0.05, omega = 0.02,
    r12 = 0.5, r22 = 0.5, r32 = 0.5, r42m = 0.5,
    c1 = 1, c2 = 5, c3 = 25,
    c1t = 1, c2t = 50, c3t = 250, c4 = 500,
    varlim = c(0, 0.005), p = 0.5)
myod2$out

# Relative efficiency (RE)
myre <- re(od = myod1, subod= myod2)
myre$re # RE = 0.88

# Constrained optimal design with J = 20 #--------
myod3 <- od.4m(icc2 = 0.2, icc3 = 0.1, icc4 = 0.05, omega = 0.02,
    r12 = 0.5, r22 = 0.5, r32 = 0.5, r42m = 0.5,
    c1 = 1, c2 = 5, c3 = 25,
    c1t = 1, c2t = 50, c3t = 250, c4 = 500,
    varlim = c(0, 0.005), J = 20)
myod3$out
# Relative efficiency (RE)
myre <- re(od = myod1, subod = myod3)
myre$re # RE = 0.58

# Constrained n, J, K and p, no calculation performed #--------
myod4 <- od.4m(icc2 = 0.2, icc3 = 0.1, icc4 = 0.05, omega = 0.02,
    r12 = 0.5, r22 = 0.5, r32 = 0.5, r42m = 0.5,
    c1 = 1, c2 = 5, c3 = 25,
    c1t = 1, c2t = 50, c3t = 250, c4 = 500,
    varlim = c(0, 0.005), p = 0.5, n = 15, J = 20, K = 5)

myod4$out
# Relative efficiency (RE)
myre <- re(od = myod1, subod = myod4)
myre$re # RE = 0.46

---

power.1  

Budget and/or sample size, power, MDES calculation for single-level experiments detecting main effects

Description

This function can calculate required budget for desired power, power or minimum detectable effect size (MDES) under fixed budget for single-level experiments. It also can perform conventional power analyses (e.g., required sample size, power, and MDES calculation).

Usage

```r
power.1(
  cost.model = TRUE,
  expr = NULL,
  constraint = NULL,
  sig.level = 0.05,
  two.tailed = TRUE,
  d = NULL,
  power = NULL,
  m = NULL,
  n = NULL,
  p = NULL,
  r12 = NULL,
  q = NULL,
  c1 = NULL,
  c1t = NULL,
  dlim = NULL,
  powerlim = NULL,
  nlim = NULL,
  mlim = NULL,
```
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cost.model</td>
<td>Logical; power analyses accommodating costs and budget (e.g., required budget for desired power, power/MDES under fixed budget) if TRUE, otherwise conventional power analyses (e.g., required sample size, power, or MDES calculation); default value is TRUE.</td>
</tr>
<tr>
<td>expr</td>
<td>Returned object from function od.1; default value is NULL; if expr is specified, parameter values of r12, c1, c1t, and p used or solved in function od.1 will be passed to the current function; only the value of p that specified or solved in function od.1 can be overwritten if constraint is specified.</td>
</tr>
<tr>
<td>constraint</td>
<td>Specify the constrained value of p in list format to overwrite that from expr; default value is NULL.</td>
</tr>
<tr>
<td>sig.level</td>
<td>Significance level or type I error rate, default value is 0.05.</td>
</tr>
<tr>
<td>two.tailed</td>
<td>Logical; two-tailed tests if TRUE, otherwise one-tailed tests; default value is TRUE.</td>
</tr>
<tr>
<td>d</td>
<td>Effect size.</td>
</tr>
<tr>
<td>power</td>
<td>Statistical power.</td>
</tr>
<tr>
<td>m</td>
<td>Total budget.</td>
</tr>
<tr>
<td>n</td>
<td>The total sample size.</td>
</tr>
<tr>
<td>p</td>
<td>The proportion of individuals to be assigned to treatment.</td>
</tr>
<tr>
<td>r12</td>
<td>The proportion of outcome variance explained by covariates.</td>
</tr>
<tr>
<td>q</td>
<td>The number of covariates.</td>
</tr>
<tr>
<td>c1</td>
<td>The cost of sampling one unit in control condition.</td>
</tr>
<tr>
<td>c1t</td>
<td>The cost of sampling one unit in treatment condition.</td>
</tr>
<tr>
<td>dlim</td>
<td>The range for searching the root of effect size (d) numerically, default value is c(0, 5).</td>
</tr>
<tr>
<td>powerlim</td>
<td>The range for searching the root of power (power) numerically, default value is c(1e-10, 1 - 1e-10).</td>
</tr>
<tr>
<td>nlim</td>
<td>The range for searching the root of sample size (n) numerically, default value is c(4, 10e10).</td>
</tr>
<tr>
<td>mlim</td>
<td>The range for searching the root of budget (m) numerically, default value is the costs sampling nlim units across treatment conditions or c(4 * ncost, 10e10 * ncost) with ncost = ((1 - p) * c1 + p * c1t).</td>
</tr>
<tr>
<td>rounded</td>
<td>Logical; round p that is from functions od.1 to two decimal places if TRUE, otherwise no rounding; default value is TRUE.</td>
</tr>
</tbody>
</table>

Value

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

# Unconstrained optimal design
myod1 <- od.1(r12 = 0.5, c1 = 1, c1t = 5, varlim = c(0, 0.2))
myod1$out # p = 0.31

# ------- Power analyses by default considering costs and budget -------
# Required budget and sample size
mym.1 <- power.1(expr = myod1, d = 0.2, q = 1, power = 0.8)
mym.1$out # m = 1032
n = 461
# mym.1$par # parameters and their values used for the function
# Or, equivalently, specify every argument in the function
mym.1 <- power.1(d = 0.2, power = 0.8, c1 = 1, c1t = 5,
r12 = 0.5, p = 0.31, q = 1)
# Required budget and sample size with constrained p
mym.2 <- power.1(expr = myod1, d = 0.2, q = 1, power = 0.8,
constraint = list(p = 0.5))
mym.2$out # m = 1183
n = 394

# Power calculation
mypower <- power.1(expr = myod1, q = 1, d = 0.2, m = 1032)
mypower$out # power = 0.80
# Power calculation under constrained p (p = 0.5)
mypower.1 <- power.1(expr = myod1, q = 1, d = 0.2, m = 1032,
constraint = list(p = 0.5))
mypower.1$out # power = 0.74

# MDES calculation
mymdes <- power.1(expr = myod1, q = 1, power = 0.80, m = 1032)
mymdes$out # d = 0.20

# ------- Conventional power analyses with cost.model = FALSE-------
# Required sample size n
myn <- power.1(cost.model = FALSE, expr = myod1, d = 0.2, q = 1, power = 0.8)
myn$out # n = 461
# myn$par # parameters and their values used for the function
# Or, equivalently, specify every argument in the function
myn <- power.1(cost.model = FALSE, d = 0.2, power = 0.8,
r12 = 0.5, p = 0.31, q = 1)
# Power calculation
mypower1 <- power.1(cost.model = FALSE, expr = myod1, n = 461, d = 0.2, q = 1)
mypower1$out # power = 0.80
# MDES calculation
mymdes1 <- power.1(cost.model = FALSE, expr = myod1, n = 461, power = 0.8, q = 1)
mymdes1$out # d = 0.20
Description

This function can calculate required budget for desired power, power or minimum detectable effect size (MDES) under fixed budget for two-level cluster randomized trials (CRTs). It also can perform conventional power analyses (e.g., required sample size, power, and MDES calculation).

Usage

```
power.2(
  cost.model = TRUE,
  expr = NULL,
  constraint = NULL,
  sig.level = 0.05,
  two.tailed = TRUE,
  d = NULL,
  power = NULL,
  m = NULL,
  n = NULL,
  J = NULL,
  p = NULL,
  icc = NULL,
  r12 = NULL,
  r22 = NULL,
  q = NULL,
  c1 = NULL,
  c2 = NULL,
  c1t = NULL,
  c2t = NULL,
  dlim = NULL,
  powerlim = NULL,
  Jlim = NULL,
  mlim = NULL,
  rounded = TRUE
)
```

Arguments

- **cost.model** Logical; power analyses accommodating costs and budget (e.g., required budget for desired power, power/MDES under fixed budget) if TRUE, otherwise conventional power analyses (e.g., required sample size, power, or MDES calculation); default value is TRUE.

- **expr** Returned object from function `od.2`; default is NULL; if `expr` is specified, parameter values of `icc`, `r12`, `r22`, `c1`, `c2`, `c1t`, `c2t`, `n`, and `p` used or solved in function `od.2` will be passed to the current function; only the values of `n` and
p that specified or solved in function od.2 can be overwritten if constraint is specified.

**constraint**
Specify the constrained values of n and/or p in list format to overwrite those from expr; default is NULL.

**sig.level**
Significance level or type I error rate, default value is 0.05.

**two.tailed**
Logical; two-tailed tests if TRUE, otherwise one-tailed tests; default value is TRUE.

**d**
Effect size.

**power**
Statistical power.

**m**
Total budget.

**n**
The level-1 sample size per level-2 unit.

**J**
The total level-2 sample size.

**p**
The proportion of level-2 clusters/units to be assigned to treatment.

**icc**
The unconditional intraclass correlation coefficient (ICC) in population or in each treatment condition.

**r12**
The proportion of level-1 variance explained by covariates.

**r22**
The proportion of level-2 variance explained by covariates.

**q**
The number of level-2 covariates.

**c1**
The cost of sampling one level-1 unit in control condition.

**c2**
The cost of sampling one level-2 unit in control condition.

**c1t**
The cost of sampling one level-1 unit in treatment condition.

**c2t**
The cost of sampling one level-2 unit in treatment condition.

**dlim**
The range for searching the root of effect size (d) numerically, default value is c(0, 5).

**powerlim**
The range for searching the root of power (power) numerically, default value is c(1e-10, 1 - 1e-10).

**Jlim**
The range for searching the root of level-2 sample size (J) numerically, default is c(4, 10e10).

**mlim**
The range for searching the root of budget (m) numerically, default is the costs sampling Jlim level-2 units across treatment conditions or c(4 * Jcost, 10e10 * Jcost), with Jcost = ((1 - p) * (c1 * n + c2) + p * (c1t * n + c2t)).

**rounded**
Logical; round n and p that are from functions od.2 to integer and two decimal places, respectively if TRUE, otherwise no rounding; default value is TRUE.

**Value**
Required budget (and/or required level-2 sample size), statistical power, or MDES depending on the specification of parameters. The function also returns the function name, design type, and parameters used in the calculation.

**References**
Examples

# Unconstrained optimal design
myod1 <- od.2(icc = 0.2, r12 = 0.5, r22 = 0.5, c1 = 1, c2 = 5, c1t = 1, c2t = 50)
myod1$out  # n = 8.9, p = 0.33

# ------- Power analyses by default considering costs and budget -------
# Required budget and sample size
mym.1 <- power.2(expr = myod1, d = 0.2, q = 1, power = 0.8)
#mym.1$par  # parameters and their values used for the function
mym.1$out  # m = 3755, J = 130.2
# Or, equivalently, specify every argument in the function
mym.1 <- power.2(d = 0.2, power = 0.8, icc = 0.2,
c1 = 1, c2 = 5, c1t = 1, c2t = 50,
r12 = 0.5, r22 = 0.5, n = 9, p = 0.33, q = 1)
# Required budget and sample size with constrained p
mym.2 <- power.2(expr = myod1, d = 0.2, q = 1, power = 0.8,
constraint = list(p = 0.5))
#mym.2$par  # parameters and their values used for the function
mym.2$out  # m = 4210, J = 115.3
# Required budget and sample size with constrained p and n
mym.3 <- power.2(expr = myod1, d = 0.2, q = 1, power = 0.8,
constraint = list(p = 0.5, n = 20))
#mym.3$par  # parameters and their values used for the function
mym.3$out  # m = 4568, J = 96.2

# Power calculation
mypower <- power.2(expr = myod1, q = 1, d = 0.2, m = 3755)
mypower$out  # power = 0.80
# Power calculation under constrained p (p = 0.5)
mypower.1 <- power.2(expr = myod1, q = 1, d = 0.2, m = 3755,
constraint = list(p = 0.5))
mypower.1$out  # power = 0.75

# MDES calculation
mymdes <- power.2(expr = myod1, q = 1, power = 0.80, m = 3755)
mymdes$out  # d = 0.20

# ------- Conventional power analyses with cost.model = FALSE-------
# Required J
myJ <- power.2(cost.model = FALSE, expr = myod1, d = 0.2, q = 1, power = 0.8)
#myJ$par  # parameters and their values used for the function
myJ$out  # J = 130.2
# Or, equivalently, specify every argument in the function
myJ <- power.2(cost.model = FALSE, d = 0.2, power = 0.8, icc = 0.2,
r12 = 0.5, r22 = 0.5, n = 9, p = 0.33, q = 1)

# Power calculation
mypower1 <- power.2(cost.model = FALSE, expr = myod1, J = 130, d = 0.2, q = 1)
mypower1$out  # power = 0.80

# MDES calculation
mymdes1 <- power.2(cost.model = FALSE, expr = myod1, J = 130, power = 0.8, q = 1)
mymdes1$out  # d = 0.20
This function can calculate required budget for desired power and power under a fixed budget for experimental studies with group mediators probing mediation effects. It also can perform conventional power analyses (e.g., required sample size and power calculation).

Usage

```r
power.2.221(
  cost.model = TRUE,
  expr = NULL,
  constraint = NULL,
  sig.level = 0.05,
  two.tailed = TRUE,
  a = NULL,
  b = NULL,
  power = NULL,
  m = NULL,
  test = "joint",
  n = NULL,
  p = NULL,
  c1 = NULL,
  c1t = NULL,
  c2 = NULL,
  c2t = NULL,
  r12 = 0,
  r22 = 0,
  r2m = 0,
  icc = NULL,
  J = NULL,
  q = 0,
  q.a = 0,
  q.b = 0,
  powerlim = NULL,
  Jlim = NULL,
  mlim = NULL,
  rounded = TRUE
)
```
Arguments

cost.model Logical; power analyses accommodating costs and budget (e.g., required budget for desired power, power/MDES under fixed budget) if TRUE, otherwise conventional power analyses (e.g., required sample size, power, or MDES calculation); default value is TRUE.

expr returned object from function od.2.221; default value is NULL; if expr is specified, parameter values of a, b, c1, c1t, and p used or solved in function od.2.221 will be passed to the current function; only the values of p and n that specified or solved in function od.2.221 can be overwritten if constraint is specified.

constraint specify the constrained value of p and/or n in a list format to overwrite that/those from expr; default value is NULL.

sig.level Significance level or type I error rate, default value is 0.05.

two.tailed Logical; two-tailed tests if TRUE, otherwise one-tailed tests; default value is TRUE.

a The treatment effect on the mediator.

b The within treatment correlation between the outcome and the mediator.

power Statistical power.

m Total budget.

test The type of test will be used to detect mediation effects. Default is the joint significance test (i.e., test = "joint"). Other choices are the Sobel test and Monte Carlo confidence interval test by specifying the argument as test = "sobel" or test = "mcci".

n The level-1 sample size per level-2 unit.

p The proportion of level-2 clusters/units to be assigned to treatment.

c1 The cost of sampling one level-1 unit in control condition.

c1t The cost of sampling one level-1 unit in treatment condition.

c2 The cost of sampling one level-2 unit in control condition.

c2t The cost of sampling one level-2 unit in treatment condition.

r12 The proportion of level-1 variance explained by covariates.

r22 The proportion of level-2 variance explained by covariates.

r2m The proportion of within treatment mediator variance explained by covariates.

icc The unconditional intraclass correlation coefficient (ICC) in population or in each treatment condition.

J The total level-2 sample size.

q The number of level-2 covariates.

q.a The number of covariates in the outcome model (except the treatment indicator and the mediator).

q.b The number of covariates in the mediator model (except the treatment indicator).

powerlim The range for searching the root of power (power) numerically, default value is c(1e-10, 1 - 1e-10).
The range for searching the root of level-2 sample size ($J$) numerically, default is $c(4, 10e10)$.

The range for searching the root of budget ($m$) numerically, default value is the costs sampling $nlim$ units across treatment conditions or $c(4 \times ncost, 10e10 \times ncost)$ with $ncost = ((1 - p) \times c1 + p \times c1t)$.

Logical; round $n$ and $p$ that are from functions od.2 to integer and two decimal places, respectively if TRUE, otherwise no rounding; default value is TRUE.

Value

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

---

**power.2m**

Budget and/or sample size, power, MDES calculation for two-level MRTs detecting main effects

Description

This function can calculate required budget for desired power, power or minimum detectable effect size (MDES) under fixed budget for two-level multisite randomized trials (MRTs). It also can perform conventional power analyses (e.g., required sample size, power, and MDES calculation).

Usage

```
power.2m(  
  cost.model = TRUE,  
  expr = NULL,  
  constraint = NULL,  
  sig.level = 0.05,  
  two.tailed = TRUE,  
  d = NULL,  
  power = NULL,  
  m = NULL,  
  n = NULL,  
  J = NULL,  
  p = NULL,  
  icc = NULL,  
  r12 = NULL,  
  r22m = NULL,  
  q = NULL,  
  c1 = NULL,  
  c2 = NULL,  
  c1t = NULL,  
  omega = NULL,  
)```
power.2m

dlim = NULL,
powerlim = NULL,
Jlim = NULL,
mlim = NULL,
rounded = TRUE
}

Arguments

cost.model Logical; power analyses accommodating costs and budget (e.g., required budget for desired power, power/MDES under fixed budget) if TRUE, otherwise conventional power analyses (e.g., required sample size, power, or MDES calculation); default value is TRUE.

eexpr Returned objects from function od.2m; default is NULL; if expr is specified, parameter values of icc, r12, r22m, c1, c2, c1t, p, and n used or solved in function od.2m will be passed to current function; only the values of p and n that specified or solved in function od.2m can be overwritten if constraint is specified.

constraint Specify the constrained values of p and/or n in list format to overwrite those from expr; default value is NULL.

sig.level Significance level or type I error rate, default value is 0.05.
two.tailed Logical; two-tailed tests if TRUE, otherwise one-tailed tests; default value is TRUE.

d Effect size.
power Statistical power.
m Total budget.
n The level-1 sample size per level-2 unit.
J The level-2 sample size per level-3 unit.
p The proportion of level-1 units to be assigned to treatment.
icc The unconditional intraclass correlation coefficient (ICC) in population or in each treatment condition.
r12 The proportion of level-1 variance explained by covariates.
r22m The proportion of variance of site-specific treatment effect explained by covariates.
q The number of covariates at level 2.
c1 The cost of sampling one level-1 unit in control condition.
c2 The cost of sampling one level-2 unit.
c1t The cost of sampling one level-1 unit in treatment condition.
omega The standardized variance of site-specific treatment effect.
dlim The range for searching the root of effect size (d) numerically, default value is c(0, 5).
powerlim The range for searching the root of power (power) numerically, default value is c(1e-10, 1 - 1e-10).
Jlim  The range for searching the root of level-2 sample size (J) numerically, default is c(4, 10e10).

mlim  The range for searching the root of budget (m) numerically, default is the costs sampling Jlim level-2 units or c(4 * Jcost, 1e+10 * Jcost) with Jcost = (1 - p) * cl * n + p * clt * n + c2.

rounded  Logical; round the values of p, n/J/K that are from functions od.4 to two decimal places and integer, respectively if TRUE, otherwise no rounding; default value is TRUE.

Value

Required budget (and/or required level-2 sample size), statistical power, or MDES depending on the specification of parameters. The function also returns the function name, design type, and parameters used in the calculation.

References


Examples

# Unconstrained optimal design #--------
myod1 <- od.2m(icc = 0.2, omega = 0.02, r12 = 0.5, r22m = 0.5,
    c1 = 1, c2 = 10, clt = 10,
    varlim = c(0, 0.005))
myod1$out # n = 19.8, p = 0.37

# Required budget and sample size
mym.1 <- power.2m(expr = myod1, d = 0.2, q = 1, power = 0.8)
mym.1$out # m = 2019, J = 20.9
mym.1$par # parameters and their values used for the function

# Or, equivalently, specify every argument in the function
mym.1 <- power.2m(d = 0.2, power = 0.8, q = 1,
    icc = 0.2, omega = 0.02, r12 = 0.5, r22m = 0.5,
    c1 = 1, c2 = 10, clt = 10,
    n = 20, p = 0.37)

# Required budget and sample size with constrained p
mym.2 <- power.2m(expr = myod1, d = 0.2, q = 1, power = 0.8,
    constraint = list(p = 0.5))
mym.2$out # m = 2373, J = 19.8

# Required budget and sample size with constrained p and n
mym.3 <- power.2m(expr = myod1, d = 0.2, q = 1, power = 0.8,
    constraint = list(p = 0.5, n = 5))
mym.3$out # m = 2502, J = 66.7

# Power calculation
mypower <- power.2m(expr = myod1, q = 1, d = 0.2, m = 2019)
mypower$out # power = 0.80

# Power calculation under constrained p (p = 0.5)
**Description**

This function can calculate required budget for desired power, power or minimum detectable effect size (MDES) under fixed budget for three-level cluster randomized trials (CRTs). It also can perform conventional power analyses (e.g., required sample size, power, and MDES calculation).

**Usage**

```
power.3(
  cost.model = TRUE,
  expr = NULL,
  constraint = NULL,
  sig.level = 0.05,
  two.tailed = TRUE,
  d = NULL,
  power = NULL,
  budget = NULL)
```
Arguments

cost.model Logical; power analyses accommodating costs and budget (e.g., required budget for desired power, power/MDES under fixed budget) if TRUE, otherwise conventional power analyses (e.g., required sample size, power, or MDES calculation); default value is TRUE.

eexpr Returned objects from function od.3; default is NULL; if eexpr is specified, parameter values of icc2, icc3, r12, r22, r32, c1, c2, c3, c1t, c2t, c3t, p, n, and J used or solved in function od.3 will be passed to the current function; only the values of p, n, and/or J that specified or solved in function od.3 can be overwritten if constraint is specified.

constraint Specify the constrained values of p, n, and/or J in list format to overwrite those from eexpr; default is NULL.

sig.level Significance level or type I error rate, default value is 0.05.

two.tailed Logical; two-tailed tests if TRUE, otherwise one-tailed tests; default value is TRUE.

d Effect size.

d power Statistical power.

m Total budget.

n The level-1 sample size per level-2 unit.

J The level-2 sample size per level-3 unit.

K The total level-3 sample size.
The proportion of level-3 clusters/units assigned to treatment.

The unconditional intraclass correlation coefficient (ICC) at level 2.

The unconditional intraclass correlation coefficient (ICC) at level 3.

The proportion of level-1 variance explained by covariates.

The proportion of level-2 variance explained by covariates.

The proportion of level-3 variance explained by covariates.

The number of covariates at level 3.

The cost of sampling one level-1 unit in control condition.

The cost of sampling one level-2 unit in control condition.

The cost of sampling one level-3 unit in control condition.

The cost of sampling one level-1 unit in treatment condition.

The cost of sampling one level-2 unit in treatment condition.

The cost of sampling one level-3 unit in treatment condition.

The range for searching the root of effect size ($d$) numerically, default value is $c(0, 5)$.

The range for searching the root of power ($power$) numerically, default value is $c(1e^{-10}, 1 - 1e^{-10})$.

The range for searching the root of level-3 sample size ($K$) numerically, default value is $c(4, 1e+10)$.

The range for searching the root of budget ($m$) numerically, default value is the costs sampling $Klim$ level-3 units across treatment conditions or $c(4 * Kcost, 1e+10 * Kcost)$ with $Kcost = ((1 - p) * (c1 * n * J + c2 * J + c3) + p * (c1t * n * J + c2t * J + c3t))$.

Logical; round the values of $p, n/J$ that are from functions $od.3$ to two decimal places and integer, respectively if TRUE, otherwise no rounding; default value is TRUE.

Required budget (and/or required level-3 sample size), statistical power, or MDES depending on the specification of parameters. The function also returns the function name, design type, and parameters used in the calculation.

References

Examples
# Unconstrained optimal design
myod1 <- od.3(icc2 = 0.2, icc3 = 0.1, r12 = 0.5, r22 = 0.5, r32 = 0.5,
              c1 = 1, c2 = 5, c3 = 25, c1t = 1, c2t = 50, c3t = 250)
myod1$out # output # n = 7.9, J = 3.2, p = 0.28
# Required budget and sample size
mym.1 <- power.3(expr = myod1, d = 0.2, q = 1, power = 0.8)
mym.1$out # m = 16032, K = 97.3
#mym.1$par # parameters and their values used for the function
# Or, equivalently, specify every argument in the function
mym.1 <- power.3(d = 0.2, power = 0.8, q = 1,
icc2 = 0.2, icc3 = 0.1, r12 = 0.5, r22 = 0.5, r32 = 0.5,
c1 = 1, c2 = 5, c3 = 25, c1t = 1, c2t = 50, c3t = 250,
n = 8, J = 3, p = 0.28)

# Required budget and sample size with constrained p
mym.2 <- power.3(expr = myod1, d = 0.2, q = 1, power = 0.8,
constraint = list(p = 0.5))
mym.2$out # m = 19239, K = 78.8

# Required budget and sample size with constrained p and J
mym.3 <- power.3(expr = myod1, d = 0.2, q = 1, power = 0.8,
constraint = list(p = 0.5, J = 20))
mym.3$out # m = 39774, K = 46.9

# Power calculation
mypower <- power.3(expr = myod1, q = 1, d = 0.2, m = 16032)
mypower$out # power = 0.80

# Power calculation under constrained p (p = 0.5)
mypower.1 <- power.3(expr = myod1, q = 1, d = 0.2, m = 16032,
constraint = list(p = 0.5))
mypower.1$out # power = 0.72

# MDES calculation
mymdes <- power.3(expr = myod1, q = 1, power = 0.80, m = 16032)
mymdes$out # d = 0.20

# Conventional power analyses with cost.model = FALSE
# Required sample size
myK <- power.3(cost.model = FALSE, expr = myod1, d = 0.2, q = 1, power = 0.8)
myK$out # K = 97.3
#myK$par # parameters and their values used for the function
# Or, equivalently, specify every argument in the function
myK <- power.3(cost.model = FALSE, d = 0.2, power = 0.8, q = 1,
icc2 = 0.2, icc3 = 0.1, r12 = 0.5, r22 = 0.5, r32 = 0.5,
n = 8, J = 3, p = 0.28)

# Power calculation
mypower1 <- power.3(cost.model = FALSE, expr = myod1, K = 97, d = 0.2, q = 1)
mypower1$out # power = 0.80

# MDES calculation
mymdes1 <- power.3(cost.model = FALSE, expr = myod1, K = 97, power = 0.8, q = 1)
mymdes1$out # d = 0.20
Description

This function can calculate required budget for desired power, power or minimum detectable effect size (MDES) under fixed budget for three-level multisite randomized trials (MRTs). It also can perform conventional power analyses (e.g., required sample size, power, and MDES calculation).

Usage

```r
power.3m(
cost.model = TRUE,
expr = NULL,
constraint = NULL,
sig.level = 0.05,
two.tailed = TRUE,
d = NULL,
power = NULL,
m = NULL,
n = NULL,
J = NULL,
K = NULL,
p = NULL,
icc2 = NULL,
icc3 = NULL,
r12 = NULL,
r22 = NULL,
r32m = NULL,
q = NULL,
c1 = NULL,
c2 = NULL,
c3 = NULL,
c1t = NULL,
c2t = NULL,
omega = NULL,
dlim = NULL,
powerlim = NULL,
Klim = NULL,
mlim = NULL,
rounded = TRUE
)
```

Arguments

- `cost.model` Logical; power analyses accommodating costs and budget (e.g., required budget for desired power, power/MDES under fixed budget) if TRUE, otherwise
conventional power analyses (e.g., required sample size, power, or MDES calculation); default value is TRUE.

- **expr**
  Returned objects from function `od.3m`; default is NULL; if `expr` is specified, parameter values of `icc2`, `icc3`, `r12`, `r22`, `r32m`, `c1`, `c2`, `c3`, `c1t`, `c2t`, `p`, `n`, and `J` used or solved in function `od.3m` will be passed to current function; only the values of `p`, `n`, and/or `J` that specified or solved in function `od.3m` can be overwritten if constraint is specified.

- **constraint**
  Specify the constrained values of `p`, `n`, and/or `J`, in list format to overwrite those from `expr`; default value is NULL.

- **sig.level**
  Significance level or type I error rate, default value is 0.05.

- **two.tailed**
  Logical; two-tailed tests if TRUE, otherwise one-tailed tests; default value is TRUE.

- **d**
  Effect size.

- **power**
  Statistical power.

- **m**
  Total budget.

- **n**
  The level-1 sample size per level-2 unit.

- **J**
  The level-2 sample size per level-3 unit.

- **K**
  The level-3 sample size per level-4 unit.

- **p**
  The proportion of level-2 units to be assigned to treatment.

- **icc2**
  The unconditional intraclass correlation coefficient (ICC) at level 2.

- **icc3**
  The unconditional intraclass correlation coefficient (ICC) at level 3.

- **r12**
  The proportion of level-1 variance explained by covariates.

- **r22**
  The proportion of level-2 variance explained by covariates.

- **r32m**
  The proportion of variance of site-specific treatment effect explained by covariates.

- **q**
  The number of covariates at level 3.

- **c1**
  The cost of sampling one level-1 unit in control condition.

- **c2**
  The cost of sampling one level-2 unit in control condition.

- **c3**
  The cost of sampling one level-3 unit.

- **c1t**
  The cost of sampling one level-1 unit in treatment condition.

- **c2t**
  The cost of sampling one level-2 unit in treatment condition.

- **omega**
  The standardized variance of site-specific treatment effect.

- **dlim**
  The range for searching the root of effect size (`d`) numerically, default value is `c(0, 5)`.

- **powerlim**
  The range for searching the root of power (power) numerically, default value is `c(1e-10, 1 - 1e-10)`.

- **Klim**
  The range for the root of level-3 sample size (`K`) numerically, default value is `c(4, 1e+10)`.

- **mlim**
  The range for searching the root of budget (m) numerically, default is the costs sampling `Klim` level-3 units or `c(4 * Kcost, 1e+10 * Kcost)` with `Kcost = ((1 - p) * (c1 * n * J + c2 * J) + p * (c1t * n * J + c2t * J) + c3.`

- **rounded**
  Logical; round the values of `p`, `n/J/K` that are from functions `od.4` to two decimal places and integer, respectively if TRUE, otherwise no rounding; default value is TRUE.
Value

Required budget (and/or required level-3 sample size), statistical power, or MDES depending on the specification of parameters. The function also returns the function name, design type, and parameters used in the calculation.

References


Examples

# Unconstrained optimal design #---------
myod1 <- od.3m(icc2 = 0.2, icc3 = 0.1, omega = 0.02,
               r12 = 0.5, r22 = 0.5, r32m = 0.5,
               c1 = 1, c2 = 5,
               c1t = 1, c2t = 200, c3 = 200,
               varlim = c(0, 0.005))
myod1$out # n = 13.1, J = 15.3, p = 0.23

# ------- Power analyses by default considering costs and budget -------
# Required budget and sample size
mym.1 <- power.3m(expr = myod1, d = 0.2, q = 1, power = 0.8)
mym.1$out # m = 15491, K = 13.6
# mym.1$par # parameters and their values used for the function
# Or, equivalently, specify every argument in the function
mym.1 <- power.3m(d = 0.2, power = 0.8, q = 1,
                   icc2 = 0.2, icc3 = 0.1, omega = 0.02,
                   r12 = 0.5, r22 = 0.5, r32m = 0.5,
                   c1 = 1, c2 = 5,
                   c1t = 1, c2t = 200, c3 = 200,
                   n = 13, J = 15, p = 0.23)

# Required budget and sample size with constrained p
mym.2 <- power.3m(expr = myod1, d = 0.2, q = 1, power = 0.8,
                   constraint = list(p = 0.5))
mym.2$out # m = 21072, K = 10.9

# Required budget and sample size with constrained p and n
mym.3 <- power.3m(expr = myod1, d = 0.2, q = 1, power = 0.8,
                   constraint = list(p = 0.5, n = 20))
mym.3$out # m = 21252, K = 10.4

# Power calculation
mypower <- power.3m(expr = myod1, q = 1, d = 0.2, m = 15491)
mypower$out # power = 0.80

# Power calculation under constrained p (p = 0.5)
mypower.1 <- power.3m(expr = myod1, q = 1, d = 0.2, m = 15491,
                      constraint = list(p = 0.5))
mypower.1$out # power = 0.62

# MDES calculation
mymdes <- power.3m(expr = myod1, q = 1, power = 0.80, m = 15491)
mymdes$out # d = 0.20
# ---- Conventional power analyses with cost.model = FALSE ----
# Required sample size
myK <- power.3m(cost.model = FALSE, expr = myod1, d = 0.2, q = 1, power = 0.8)
myK$\text{out}$ # K = 13.6
# myK$\text{par}$ # parameters and their values used for the function
# Or, equivalently, specify every argument in the function
myK <- power.3m(cost.model = FALSE, d = 0.2, power = 0.8, q = 1,
                icc2 = 0.2, icc3 = 0.1, omega = 0.02,
                r12 = 0.5, r22 = 0.5, r32m = 0.5,
                c1 = 1, c2 = 5,
                c1t = 1, c2t = 200, c3 = 200,
                n = 13, J = 15, p = 0.23)

# Power calculation
mypower1 <- power.3m(cost.model = FALSE, expr = myod1, K = 13.6, d = 0.2, q = 1)
mypower1$\text{out}$ # power = 0.80

# MDES calculation
mymdes1 <- power.3m(cost.model = FALSE, expr = myod1, K = 13.6, power = 0.8, q = 1)
mymdes1$\text{out}$ # d = 0.20

---

power.4  

Budget and/or sample size, power, MDES calculation for four-level CRTs detecting main effects

Description

This function can calculate required budget for desired power, power or minimum detectable effect size (MDES) under fixed budget for four-level cluster randomized trials (CRTs). It also can perform conventional power analyses (e.g., required sample size, power, and MDES calculation).

Usage

```r
power.4(
  cost.model = TRUE,
  expr = NULL,
  constraint = NULL,
  sig.level = 0.05,
  two.tailed = TRUE,
  d = NULL,
  power = NULL,
  m = NULL,
  n = NULL,
  J = NULL,
  K = NULL,
  L = NULL,
)```
p = NULL,
icc2 = NULL,
icc3 = NULL,
icc4 = NULL,
r12 = NULL,
r22 = NULL,
r32 = NULL,
r42 = NULL,
q = NULL,
c1 = NULL,
c2 = NULL,
c3 = NULL,
c4 = NULL,
c1t = NULL,
c2t = NULL,
c3t = NULL,
c4t = NULL,
dlim = NULL,
powerlim = NULL,
Llim = NULL,
mlim = NULL,
rounded = TRUE
)

Arguments

cost.model Logical; power analyses accommodating costs and budget (e.g., required budget for desired power, power/MDES under fixed budget) if TRUE, otherwise conventional power analyses (e.g., required sample size, power, or MDES calculation); default value is TRUE.

expr Returned objects from function \texttt{od.4}; default value is NULL; if \texttt{expr} is specified, parameter values of \texttt{icc2}, \texttt{icc3}, \texttt{icc4}, \texttt{r12}, \texttt{r22}, \texttt{r32}, \texttt{r42}, \texttt{c1}, \texttt{c2}, \texttt{c3}, \texttt{c4}, \texttt{c1t}, \texttt{c2t}, \texttt{c3t}, \texttt{c4t}, \texttt{p}, \texttt{n}, \texttt{J}, and \texttt{K} used or solved in function \texttt{od.4} will be passed to current function; only the values of \texttt{p}, \texttt{n}, \texttt{J}, and/or \texttt{K} that specified or solved in function \texttt{od.4} can be overwritten if \texttt{constraint} is specified.

constraint Specify the constrained values of \texttt{p}, \texttt{n}, \texttt{J}, and/or \texttt{K} in list format to overwrite those from \texttt{expr}; default value is NULL.

sig.level Significance level or type I error rate, default value is 0.05.

two.tailed Logical; two-tailed tests if TRUE, otherwise one-tailed tests; default value is TRUE.

d Effect size.

power Statistical power.

m Total budget.

n The level-1 sample size per level-2 unit.

J The level-2 sample size per level-3 unit.

K The level-3 sample size per level-4 unit.
L  The total level-4 sample size.
p  The proportion of level-4 clusters/units to be assigned to treatment.
icc2  The unconditional intraclass correlation coefficient (ICC) at level 2.
icc3  The unconditional intraclass correlation coefficient (ICC) at level 3.
icc4  The unconditional intraclass correlation coefficient (ICC) at level 4.
r12  The proportion of level-1 variance explained by covariates.
r22  The proportion of level-2 variance explained by covariates.
r32  The proportion of level-3 variance explained by covariates.
r42  The proportion of level-4 variance explained by covariates.
q  The number of covariates at level 4.
c1  The cost of sampling one level-1 unit in control condition.
c2  The cost of sampling one level-2 unit in control condition.
c3  The cost of sampling one level-3 unit in control condition.
c4  The cost of sampling one level-4 unit in control condition.
c1t  The cost of sampling one level-1 unit in treatment condition.
c2t  The cost of sampling one level-2 unit in treatment condition.
c3t  The cost of sampling one level-3 unit in treatment condition.
c4t  The cost of sampling one level-4 unit in treatment condition.
dlim  The range for searching the root of effect size (d) numerically, default value is c(0, 5).
powerlim  The range for searching the root of power (power) numerically, default value is c(1e-10, 1 - 1e-10).
Llim  The range for searching the root of level-4 sample size (L) numerically, default value is c(4, 1e+10).
mlim  The range for searching the root of budget (m) numerically, default value is c(4 * Lcost, 1e+10 * Lcost) with Lcost = ((1 - p) * (c1 * n * J * K + c2 * J * K + c3 * K + c4) + p * (c1t * n * J * K + c2t * J * K + c3t * K + c4t)).
rounded  Logical; round the values of p, n/J/K that are from functions od.4 to two decimal places and integer, respectively if TRUE, otherwise no rounding; default value is TRUE.

Value

Required budget (and/or required level-4 sample size), statistical power, or MDES depending on the specification of parameters. The function also returns the function name, design type, and parameters used in the calculation.
Examples

# Unconstrained optimal design
myod1 <- od.4(icc2 = 0.2, icc3 = 0.1, icc4 = 0.05, 
          r12 = 0.5, r22 = 0.5, r32 = 0.5, r42 = 0.5, 
          c1 = 1, c2 = 5, c3 = 25, c4 = 125, 
          c1t = 1, c2t = 50, c3t = 250, c4t = 2500)
myod1$out # output # n = 7.1, J = 3.2, K = 4.2, p = 0.23

# ------- Power analyses by default considering costs and budget -------
# Required budget and sample size
mym.1 <- power.4(expr = myod1, d = 0.2, q = 1, power = 0.8)
mym.1$out # m = 71161, L = 57.1
# Or, equivalently, specify every argument in the function
mym.1 <- power.4(d = 0.2, power = 0.8, q = 1,
                 icc2 = 0.2, icc3 = 0.1, icc4 = 0.05, 
                 r12 = 0.5, r22 = 0.5, r32 = 0.5, r42 = 0.5, 
                 c1 = 1, c2 = 5, c3 = 25, c4 = 125, 
                 c1t = 1, c2t = 50, c3t = 250, c4t = 2500, 
                 n = 7, J = 3, K = 4, p = 0.23)

# Required budget and sample size with constrained p (p = 0.5)
mym.2 <- power.4(expr = myod1, d = 0.2, q = 1, power = 0.8, 
                 constraint = list(p = 0.5))
mym.2$out # m = 93508, L = 41.1

# Required budget and sample size with constrained p and K
mym.3 <- power.4(expr = myod1, d = 0.2, q = 1, power = 0.8, 
                 constraint = list(p = 0.5, K = 20))
mym.3$out # m = 157365, L = 25.7

# Power calculation
mypower <- power.4(expr = myod1, q = 1, d = 0.2, m = 71161)
mypower$out # power = 0.80

# Power calculation under constrained p (p = 0.5)
mypower.1 <- power.4(expr = myod1, q = 1, d = 0.2, m = 71161, 
                    constraint = list(p = 0.5))
mypower.1$out # power = 0.68

# MDES calculation
mymdes <- power.4(expr = myod1, q = 1, power = 0.80, m = 71161)
mymdes$out # d = 0.20

# ------- Conventional power analyses with cost.model = FALSE-------
# Required sample size
myL <- power.4(cost.model = FALSE, expr = myod1, d = 0.2, q = 1, power = 0.8)
myL$out # L = 57.1
# Or, equivalently, specify every argument in the function
myL <- power.4(cost.model = FALSE, d = 0.2, power = 0.8, q = 1,
                icc2 = 0.2, icc3 = 0.1, icc4 = 0.05, 
                r12 = 0.5, r22 = 0.5, r32 = 0.5, r42 = 0.5, 
                n = 7, J = 3, K = 4, p = 0.23)
# Power calculation
mypower1 <- power.4(cost.model = FALSE, expr = myod1, L = 57, d = 0.2, q = 1)
mypower1$out # power = 0.80

# MDES calculation
mymdes1 <- power.4(cost.model = FALSE, expr = myod1, L = 57, power = 0.8, q = 1)
mymdes1$out # d = 0.20

---

**power.4m**  
*Budget and/or sample size, power, MDES calculation for four-level MRTs detecting main effects*

---

**Description**

This function can calculate required budget for desired power, power or minimum detectable effect size (MDES) under fixed budget for four-level multisite randomized trials (MRTs). It also can perform conventional power analyses (e.g., required sample size, power, and MDES calculation).

**Usage**

```r
power.4m(
  cost.model = TRUE,
  expr = NULL,
  constraint = NULL,
  sig.level = 0.05,
  two.tailed = TRUE,
  d = NULL,
  power = NULL,
  m = NULL,
  n = NULL,
  J = NULL,
  K = NULL,
  L = NULL,
  p = NULL,
  icc2 = NULL,
  icc3 = NULL,
  icc4 = NULL,
  r12 = NULL,
  r22 = NULL,
  r32 = NULL,
  r42m = NULL,
  q = NULL,
  c1 = NULL,
  c2 = NULL,
  c3 = NULL,
  c4 = NULL,
)```
power.4m

Arguments

cost.model Logical; power analyses accommodating costs and budget (e.g., required budget for desired power, power/MDES under fixed budget) if TRUE, otherwise conventional power analyses (e.g., required sample size, power, or MDES calculation); default value is TRUE.

expr Returned objects from function od.4m; default is NULL; if expr is specified, parameter values of icc2, icc3, icc4, r12, r22, r32, r42m, c1, c2, c3, c4, c1t, c2t, c3t, p, n, J, and K used or solved in function od.4m will be passed to current function; only the values of p, n, J, and/or K that specified or solved in function od.4m can be overwritten if constraint is specified.

constraint The constrained values of p, n, J, and/or K in list format to overwrite those from expr; default value is NULL.

sig.level Significance level or type I error rate, default value is 0.05.

two.tailed Logical; two-tailed tests if TRUE, otherwise one-tailed tests; default value is TRUE.

d Effect size.

power Statistical power.

m Total budget.

n The level-1 sample size per level-2 unit.

J The level-2 sample size per level-3 unit.

K The level-3 sample size per level-4 unit.

L The total level-4 sample size.

p The proportion of level-3 units to be assigned to treatment.

icc2 The unconditional intraclass correlation coefficient (ICC) at level 2.

icc3 The unconditional intraclass correlation coefficient (ICC) at level 3.

icc4 The unconditional intraclass correlation coefficient (ICC) at level 4.

r12 The proportion of level-1 variance explained by covariates.

r22 The proportion of level-2 variance explained by covariates.

r32 The proportion of level-3 variance explained by covariates.

r42m The proportion of variance of site-specific treatment effect explained by covariates.
power.4m

q The number of covariates at level 4.
c1 The cost of sampling one level-1 unit in control condition.
c2 The cost of sampling one level-2 unit in control condition.
c3 The cost of sampling one level-3 unit in control condition.
c4 The cost of sampling one level-4 unit.
c1t The cost of sampling one level-1 unit in treatment condition.
c2t The cost of sampling one level-2 unit in treatment condition.
c3t The cost of sampling one level-3 unit in treatment condition.
omega The standardized variance of site-specific treatment effect.
dlim The range for searching the root of effect size (d) numerically, default value is c(0, 5).
powerlim The range for searching the root of power (power) numerically, default value is c(1e-10, 1 - 1e-10).
Llim The range for searching the root of level-4 sample size (L) numerically, default value is c(4, 1e+10).
mlim The range for searching the root of budget (m) numerically, default is the costs sampling Llim level-4 units or c(4 * Lcost, 1e+10 * Lcost) with Lcost = ((1 - p) * (c1 * n * J * K + c2 * J * K + c3 * K) + p * (c1t * n * J * K + c2t * J * K + c3t * K) + c4.
rounded Logical; round the values of p, n/J/K that are from functions od.4 to two decimal places and integer, respectively if TRUE, otherwise no rounding; default value is TRUE.

Value

Required budget (and/or required level-4 sample size), statistical power, or MDES depending on the specification of parameters. The function also returns the function name, design type, and parameters used in the calculation.

Examples

# Unconstrained optimal design #--------
myod1 <- od.4m(icc2 = 0.2, icc3 = 0.1, icc4 = 0.05, omega = 0.02,
               r12 = 0.5, r22 = 0.5, r32 = 0.5, r42m = 0.5,
               c1 = 1, c2 = 5, c3 = 25,
               c1t = 1, c2t = 50, c3t = 250, c4 = 500,
               varlim = c(0, 0.005))
myod1$out # n = 8.3, J = 3.2, K = 4.9, p = 0.36

# ------- Power analyses by default considering costs and budget -------
# Required budget and sample size
mym.1 <- power.4m(expr = myod1, d = 0.2, q = 1, power = 0.8)
mym.1$out # m = 30201, L = 20.6
# mym.1$par # parameters and their values used for the function
# Or, equivalently, specify every argument in the function
mym.1 <- power.4m(d = 0.2, power = 0.8, q = 1,
icc2 = 0.2, icc3 = 0.1, icc4 = 0.05, omega = 0.02,
r12 = 0.5, r22 = 0.5, r32 = 0.5, r42m = 0.5,
c1 = 1, c2 = 5, c3 = 25,
c1t = 1, c2t = 50, c3t = 250, c4 = 500,
n = 8, J = 3, K = 5, p = 0.36)

# Required budget and sample size with constrained p
mym.2 <- power.4m(expr = myod1, d = 0.2, q = 1, power = 0.8,
constraint = list(p = 0.5))
mym.2$out # m = 33183, L = 19.3

# Required budget and sample size with constrained p and n
mym.3 <- power.4m(expr = myod1, d = 0.2, q = 1, power = 0.8,
constraint = list(p = 0.5, n = 20))
mym.3$out # m = 34262, L = 18.0

# Power calculation
mypower <- power.4m(expr = myod1, q = 1, d = 0.2, m = 30201)
mypower$out # power = 0.80

# Power calculation under constrained p (p = 0.5)
mypower.1 <- power.4m(expr = myod1, q = 1, d = 0.2, m = 30201,
constraint = list(p = 0.5))
mypower.1$out # power = 0.76

# MDES calculation
mymdes <- power.4m(expr = myod1, q = 1, power = 0.80, m = 30201)
mymdes$out # d = 0.20

# ------ Conventional power analyses with cost.model = FALSE------
# Required sample size
myL <- power.4m(cost.model = FALSE, expr = myod1, d = 0.2, q = 1, power = 0.8)
myL$out # L = 20.6

# myL$par # parameters and their values used for the function
# Or, equivalently, specify every argument in the function
myL <- power.4m(cost.model = FALSE, d = 0.2, power = 0.8, q = 1,
icc2 = 0.2, icc3 = 0.1, icc4 = 0.05, omega = 0.02,
r12 = 0.5, r22 = 0.5, r32 = 0.5, r42m = 0.5,
c1 = 1, c2 = 5, c3 = 25,
c1t = 1, c2t = 50, c3t = 250, c4 = 500,
n = 8, J = 3, K = 5, p = 0.36)

# Power calculation
mypower1 <- power.4m(cost.model = FALSE, expr = myod1, L = 20.6, d = 0.2, q = 1)
mypower1$out # power = 0.80

# MDES calculation
mymdes1 <- power.4m(cost.model = FALSE, expr = myod1, L = 20.6, power = 0.8, q = 1)
mymdes1$out # d = 0.20

Relative efficiency (RE) calculation
Description

Calculate the relative efficiency (RE) between two designs, it returns same results as those from function `rpe`.

Usage

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

Arguments

- **od**: Returned object of first design (e.g., unconstrained optimal design) from function `od.1`, `od.2`, `od.3`, `od.4`, `od.2m`, `od.3m`, or `od.4m`.
- **subod**: Returned object of second design (e.g., constrained optimal design) from function `od.1`, `od.2`, `od.3`, `od.4`, `od.2m`, `od.3m`, or `od.4m`.
- **rounded**: Logical; round the values of `p`, `n/J/K` that are from functions to two decimal places and integer, respectively if TRUE, no rounding if FALSE; default is TRUE.
- **verbose**: Logical; print the value of relative efficiency if TRUE, otherwise not; default is TRUE.

Value

Relative efficiency value.

References


Examples

```r
# Unconstrained optimal design of 2-level CRT #--------
myod1 <- od.2(icc = 0.2, r12 = 0.5, r22 = 0.5, c1 = 1, c2 = 5, c1t = 1, c2t = 50, 
             varlim = c(0.01, 0.02))
# Constrained optimal design with n = 20
myod2 <- od.2(icc = 0.2, r12 = 0.5, r22 = 0.5, c1 = 1, c2 = 5, c1t = 1, c2t = 50, 
              n = 20, varlim = c(0.005, 0.025))
# Relative efficiency (RE)
myre <- re(od = myod1, subod= myod2)
myre$out # RE = 0.88
# Constrained optimal design with p = 0.5
myod2 <- od.2(icc = 0.2, r12 = 0.5, r22 = 0.5, c1 = 1, c2 = 5, c1t = 1, c2t = 50, 
              p = 0.5, varlim = c(0.005, 0.025))
# Relative efficiency (RE)
myre <- re(od = myod1, subod= myod2)
```
myre$out # RE = 0.90

# Unconstrained optimal design of 3-level CRT #----------
myod1 <- od.3(icc2 = 0.2, icc3 = 0.1, r12 = 0.5, r22 = 0.5, r32 = 0.5,
c1 = 1, c2 = 5, c3 = 25, c1t = 1, c2t = 50, c3t = 250,
varlim = c(0.005, 0.025))
# Constrained optimal design with J = 20
myod2 <- od.3(icc2 = 0.2, icc3 = 0.1, r12 = 0.5, r22 = 0.5, r32 = 0.5, J = 20,
c1 = 1, c2 = 5, c3 = 25, c1t = 1, c2t = 50, c3t = 250,
varlim = c(0, 0.025))
# Relative efficiency (RE)
myre <- re(od = myod1, subod= myod2)
myre$out # RE = 0.53

# Unconstrained optimal design of 4-level CRT #--------
myod1 <- od.4(icc2 = 0.2, icc3 = 0.1, icc4 = 0.05, r12 = 0.5,
r22 = 0.5, r32 = 0.5, r42 = 0.5,
c1 = 1, c2 = 5, c3 = 25, c4 = 125,
c1t = 1, c2t = 50, c3t = 250, c4t = 2500,
varlim = c(0, 0.01))
# Constrained optimal design with p = 0.5
myod2 <- od.4(icc2 = 0.2, icc3 = 0.1, icc4 = 0.05, r12 = 0.5, p = 0.5,
r22 = 0.5, r32 = 0.5, r42 = 0.5,
c1 = 1, c2 = 5, c3 = 25, c4 = 125,
c1t = 1, c2t = 50, c3t = 250, c4t = 2500,
varlim = c(0, 0.01))
# Relative efficiency (RE)
myre <- re(od = myod1, subod= myod2)
myre$out # RE = 0.78

---

**rpe**

Relative precision and efficiency (RPE) calculation

**Description**

Calculate the relative precision and efficiency (RPE) between two designs, it returns same results as those from function `re`.

**Usage**

```r
rpe(od, subod, rounded = TRUE, verbose = TRUE)
```

**Arguments**

- `od` Returned object of first design (e.g., unconstrained optimal design) from function `od.1`, `od.2`, `od.3`, `od.4`, `od.2m`, `od.3m`, or `od.4m`.
- `subod` Returned object of second design (e.g., constrained optimal design) from function `od.1`, `od.2`, `od.3`, `od.4`, `od.2m`, `od.3m`, or `od.4m`.
rounded Logical; round the values of \( p, n/J/K \) that are from functions to two decimal places and integer, respectively if TRUE, no rounding if FALSE; default is TRUE.

verbose Logical; print the value of relative precision and efficiency if TRUE, otherwise not; default is TRUE.

Value

Relative precision and efficiency value.

References


Examples

```r
# Unconstrained optimal design of 2-level CRT #----------
myod1 <- od.2(icc = 0.2, r12 = 0.5, r22 = 0.5, c1 = 1, c2 = 5, c1t = 1, c2t = 50,
              varlim = c(0.01, 0.02))
# Constrained optimal design with \( n = 20 \)
myod2 <- od.2(icc = 0.2, r12 = 0.5, r22 = 0.5, c1 = 1, c2 = 5, c1t = 1, c2t = 50,
              n = 20, varlim = c(0.005, 0.025))
# Relative precision and efficiency (RPE)
myrpe <- rpe(od = myod1, subod = myod2)
myrpe$out # RPE = 0.88
# Constrained optimal design with \( p = 0.5 \)
myod2 <- od.2(icc = 0.2, r12 = 0.5, r22 = 0.5, c1 = 1, c2 = 5, c1t = 1, c2t = 50,
              p = 0.5, varlim = c(0.005, 0.025))
# Relative precision and efficiency (RPE)
myrpe <- rpe(od = myod1, subod = myod2)
myrpe$out # RPE = 0.90
# Unconstrained optimal design of 3-level CRT #----------
myod1 <- od.3(icc2 = 0.2, icc3 = 0.1, r12 = 0.5, r22 = 0.5, r32 = 0.5,
              c1 = 1, c2 = 5, c3 = 25, c1t = 1, c2t = 50, c3t = 250,
              varlim = c(0.005, 0.025))
# Constrained optimal design with \( J = 20 \)
myod2 <- od.3(icc2 = 0.2, icc3 = 0.1, r12 = 0.5, r22 = 0.5, r32 = 0.5, J = 20,
              c1 = 1, c2 = 5, c3 = 25, c1t = 1, c2t = 50, c3t = 250,
              varlim = c(0, 0.025))
# Relative precision and efficiency (RPE)
myrpe <- rpe(od = myod1, subod = myod2)
myrpe$out # RPE = 0.53
# Unconstrained optimal design of 4-level CRT #----------
myod1 <- od.4(icc2 = 0.2, icc3 = 0.1, icc4 = 0.05, r12 = 0.5,
```
r22 = 0.5, r32 = 0.5, r42 = 0.5,
c1 = 1, c2 = 5, c3 = 25, c4 = 125,
c1t = 1, c2t = 50, c3t = 250, c4t = 2500,
varlim = c(0, 0.01))

# Constrained optimal design with p = 0.5
myod2 <- od.4(icc2 = 0.2, icc3 = 0.1, icc4 = 0.05, r12 = 0.5, p = 0.5,
               r22 = 0.5, r32 = 0.5, r42 = 0.5,
c1 = 1, c2 = 5, c3 = 25, c4 = 125,
c1t = 1, c2t = 50, c3t = 250, c4t = 2500,
               varlim = c(0, 0.01))

# Relative precision and efficiency (RPE)
myrpe <- rpe(od = myod1, subod= myod2)
myrpe$out # RPE = 0.78
Index

gen.design.pars, 3

od.1, 4, 27, 52, 53
od.2, 6, 29, 30, 52, 53
od.2.221, 8, 33
od.2m, 11, 35, 52, 53
od.3, 13, 38, 39, 52, 53
od.3m, 16, 42, 52, 53
od.4, 19, 36, 42, 45, 46, 50, 52, 53
od.4m, 23, 49, 52, 53
odr (odr-package), 2
odr-package, 2

power.1, 26
power.2, 28
power.2.221, 32
power.2m, 34
power.3, 37
power.3m, 41
power.4, 44
power.4m, 48

re, 51, 53
rpe, 52, 53