# Package ‘PowerUpR’

June 1, 2019

**Type** Package

**Title** Power Analysis Tools for Multilevel Randomized Experiments

**Version** 1.0.4

**Date** 2019-06-01

**Description**

Includes tools to calculate statistical power, minimum detectable effect size (MDES), MDES difference (MDESD), and minimum required sample size for various multilevel randomized experiments with continuous outcomes. Some of the functions can assist with planning two- and three-level cluster-randomized trials (CRTs) sensitive to multilevel moderation and mediation (2-1-1, 2-2-1, and 3-2-1). See 'PowerUp!' Excel series at <https://www.causalevaluation.org/>.

**Suggests** knitr, rmarkdown

**VignetteBuilder** knitr

**License** GPL (>= 3)

**NeedsCompilation** no

**Author** Metin Bulus [aut, cre],
Nianbo Dong [aut],
Benjamin Kelcey [aut],
Jessaca Spybrook [aut]

**Maintainer** Metin Bulus <bulusmetin@gmail.com>

**Repository** CRAN

**Date/Publication** 2019-06-01 10:40:03 UTC

## R topics documented:

---

`PowerUpR-package` *Power Analysis Tools for Multilevel Randomized Experiments*

---

## Description

*PowerUp!* series consist of three excel-based applications to design various multilevel randomized experiments to detect main treatment effects, and to design two- and three-level cluster-randomized trials (CRTs) to detect multilevel moderation and mediation. For more information please refer to http://www.causalevaluation.org/.

---

`bcra3f2` *Three-Level Blocked (Fixed) Cluster-level Random Assignment Design, Treatment at Level 2*

---

## Description

Use `mdes.bcra3f2()` to calculate the minimum detectable effect size, `power.bcra3f2()` to calculate the statistical power, and `mrss.bcra3f2()` to calculate the minimum required sample size.

## Usage

```
mdes.bcra3f2(power=.80, alpha=.05, two.tailed=TRUE,
             rho2, p=.50, g2=0, r21=0, r22=0,
             n, J, K)

power.bcra3f2(es=.25, alpha=.05, two.tailed=TRUE,
              rho2, p=.50, g2=0, r21=0, r22=0,
              n, J, K)

mrss.bcra3f2(es=.25, power=.80, alpha=.05, two.tailed=TRUE,
             n, J, K0=10, tol=.10,
             rho2, p=.50, g2=0, r21=0, r22=0)
```

## Arguments

| | |
|---|---|
| power | statistical power $(1-\beta)$. |
| es | effect size. |
| alpha | probability of type I error. |
| two.tailed | logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing. |
| rho2 | proportion of variance in the outcome between level 2 units (unconditional ICC2). |
| p | average proportion of level 2 units randomly assigned to treatment within level 3 units. |
| 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. |
| n | harmonic mean of level 1 units across level 2 units (or simple average). |
| J | harmonic mean of level 2 units across level 3 units (or simple average). |
| K | number of level 3 units. |
| K0 | starting value for K. |
| tol | tolerance to end iterative process for finding K. |

## Value

| | |
|---|---|
| fun | function name. |
| parms | list of parameters used in power calculation. |
| df | degrees of freedom. |
| ncp | noncentrality parameter. |
| power | statistical power $(1-\beta)$. |
| mdes | minimum detectable effect size. |
| K | number of level 3 units. |

## Examples

```
# cross-checks
mdes.bcra3f2(rho2=.10, n=20, J=44, K=5)
power.bcra3f2(es = .145, rho2=.10, n=20, J=44, K=5)
mrss.bcra3f2(es = .145, rho2=.10, n=20, J=44)
```

bcra3r2 — *Three-Level Blocked Cluster-level Random Assignment Design, Treatment at Level 2*

## Description

Use `mdes.bcra3r2()` to calculate the minimum detectable effect size, `power.bcra3r2()` to calculate the statistical power, and `mrss.bcra3r2()` to calculate the minimum required sample size.

## Usage

```
mdes.bcra3r2(power=.80, alpha=.05, two.tailed=TRUE,
             rho2, rho3, omega3, p=.50, g3=0, r21=0, r22=0, r2t3=0,
             n, J, K)

power.bcra3r2(es=.25, alpha=.05, two.tailed=TRUE,
              rho2, rho3,  omega3, p=.50, g3=0, r21=0, r22=0, r2t3=0,
              n, J, K)

mrss.bcra3r2(es=.25, power=.80, alpha=.05, two.tailed=TRUE,
             n, J, K0=10, tol=.10,
             rho2, rho3,  omega3, p=.50, g3=0, r21=0, r22=0, r2t3=0)
```

## Arguments

| | |
|---|---|
| `power` | statistical power $(1-\beta)$. |
| `es` | effect size. |
| `alpha` | probability of type I error. |
| `two.tailed` | logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing. |
| `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` | treatment effect heterogeneity as ratio of treatment effect variance among level 3 units to the residual variance at level 3. |
| `p` | average proportion of level 2 units randomly assigned to treatment within level 3 units. |
| `g3` | number of covariates at level 3. |
| `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. |
| `r2t3` | proportion of treatment effect variance among level 3 units explained by level 3 covariates. |
| `n` | harmonic mean of level 1 units across level 2 units (or simple average). |
| `J` | harmonic mean of level 2 units across level 3 units (or simple average). |

| | |
|---|---|
| K | number of level 3 units. |
| K0 | starting value for K. |
| tol | tolerance to end iterative process for finding K. |

## Value

| | |
|---|---|
| fun | function name. |
| parms | list of parameters used in power calculation. |
| df | degrees of freedom. |
| ncp | noncentrality parameter. |
| power | statistical power $(1-\beta)$. |
| mdes | minimum detectable effect size. |
| K | number of level 3 units. |

## See Also

cosa.bcrd3r2

## Examples

```
# cross-checks
mdes.bcra3r2(rho3=.13, rho2=.10, omega3=.4,
             n=10, J=6, K=24)
power.bcra3r2(es = .246, rho3=.13, rho2=.10, omega3=.4,
              n=10, J=6, K=24)
mrss.bcra3r2(es = .246, rho3=.13, rho2=.10, omega3=.4,
             n=10, J=6)
```

---

bcra4f3 *Four-Level Blocked (Fixed) Cluster-level Random Assignment Design, Treatment at Level 3*

---

## Description

Use mdes.bcra4f3() to calculate the minimum detectable effect size, power.bcra4f3() to calculate the statistical power, and mrss.bcra4f3() to calculate the minimum required sample size.

## Usage

```
mdes.bcra4f3(power=.80, alpha=.05, two.tailed=TRUE,
             rho2, rho3, p=.50, r21=0, r22=0, r23=0, g3=0,
             n, J, K, L)

power.bcra4f3(es=.25, alpha=.05, two.tailed=TRUE,
              rho2, rho3, p=.50, r21=0, r22=0, r23=0, g3=0,
              n, J, K, L)
```

```
mrss.bcra4f3(es=.25, power=.80, alpha=.05, two.tailed=TRUE,
             n, J, K, L0=10, tol=.10,
             rho2, rho3, p=.50, g3=0, r21=0, r22=0, r23=0)
```

## Arguments

| | |
|---|---|
| power | statistical power $(1-\beta)$. |
| es | effect size. |
| alpha | probability of type I error. |
| two.tailed | logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing. |
| 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). |
| p | average proportion of level 3 units randomly assigned to treatment within level 4 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 2 variance in the outcome explained by level 2 covariates. |
| r23 | proportion of level 3 variance in the outcome explained by level 3 covariates. |
| n | harmonic mean of level 1 units across level 2 units (or simple average). |
| J | harmonic mean of level 2 units across level 3 units (or simple average). |
| K | harmonic mean of level 3 units across level 4 units (or simple average). |
| L | number of level 4 units. |
| L0 | starting value for L. |
| tol | tolerance to end iterative process for finding L. |

## Value

| | |
|---|---|
| fun | function name. |
| parms | list of parameters used in power calculation. |
| df | degrees of freedom. |
| ncp | noncentrality parameter. |
| power | statistical power $(1-\beta)$. |
| mdes | minimum detectable effect size. |
| L | number of level 4 units. |

## Examples

```
# cross-checks
mdes.bcra4f3(rho3=.15, rho2=.15,
             n=10, J=4, K=4, L=15)
power.bcra4f3(es=0.339, rho3=.15, rho2=.15,
              n=10, J=4, K=4, L=15)
mrss.bcra4f3(es=0.339, rho3=.15, rho2=.15,
             n=10, J=4, K=4)
```

bcra4r2 *Four-Level Blocked Cluster-level Random Assignment Design, Treatment at Level 2*

## Description

Use `mdes.bcra4r2()` to calculate the minimum detectable effect size, `power.bcra4r2()` to calculate the statistical power, and `mrss.bcra4r2()` to calculate the minimum required sample size.

## Usage

```
mdes.bcra4r2(power=.80, alpha=.05, two.tailed=TRUE,
             rho2, rho3, rho4, omega3, omega4,
             p=.50, r21=0, r22=0, r2t3=0, r2t4=0, g4=0,
             n, J, K, L)

power.bcra4r2(es=.25, alpha=.05, two.tailed=TRUE,
              rho2, rho3, rho4, omega3, omega4,
              p=.50, r21=0, r22=0, r2t3=0, r2t4=0, g4=0,
              n, J, K, L)

mrss.bcra4r2(es=.25, power=.80, alpha=.05, two.tailed=TRUE,
             n, J, K, L0=10, tol=.10,
             rho2, rho3, rho4, omega3, omega4,
             p=.50, r21=0, r22=0, r2t3=0, r2t4=0, g4=0)
```

## Arguments

| | |
|---|---|
| `power` | statistical power $(1-\beta)$. |
| `es` | effect size. |
| `alpha` | probability of type I error. |
| `two.tailed` | logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing. |
| `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` | treatment effect heterogeneity as ratio of treatment effect variance among level 3 units to the residual variance at level 3. |
| `omega4` | treatment effect heterogeneity as ratio of treatment effect variance among level 4 units to the residual variance at level 4. |
| `p` | average proportion of level 2 units randomly assigned to treatment within level 3 units. |
| `g4` | number of covariates at level 4. |
| `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. |
| r2t3 | proportion of treatment effect variance among level 3 units explained by level 3 covariates. |
| r2t4 | proportion of treatment effect variance among level 4 units explained by level 4 covariates. |
| n | harmonic mean of level 1 units across level 2 units (or simple average). |
| J | harmonic mean of level 2 units across level 3 units (or simple average). |
| K | harmonic mean of level 3 units across level 4 units (or simple average). |
| L | number of level 4 units. |
| L0 | starting value for L. |
| tol | tolerance to end iterative process for finding L. |

## Value

| | |
|---|---|
| fun | function name. |
| parms | list of parameters used in power calculation. |
| df | degrees of freedom. |
| ncp | noncentrality parameter. |
| power | statistical power $(1-\beta)$. |
| mdes | minimum detectable effect size. |
| L | number of level 4 units. |

## See Also

cosa.bcrd4r2

## Examples

```
# cross-checks
mdes.bcra4r2(rho4=.05, rho3=.15, rho2=.15,
             omega4=.50, omega3=.50, n=10, J=4, K=4, L=20)
power.bcra4r2(es = .206, rho4=.05, rho3=.15, rho2=.15,
              omega4=.50, omega3=.50, n=10, J=4, K=4, L=20)
mrss.bcra4r2(es = .206, rho4=.05, rho3=.15, rho2=.15,
             omega4=.50, omega3=.50, n=10, J=4, K=4)
```

bcra4r3 *Four-Level Blocked Cluster-level Random Assignment Design, Treatment at Level 3*

## Description

Use `mdes.bcra4r3()` to calculate the minimum detectable effect size, `power.bcra4r3()` to calculate the statistical power, and `mrss.bcra4r3()` to calculate the minimum required sample size.

## Usage

```
mdes.bcra4r3(power=.80, alpha=.05, two.tailed=TRUE,
             rho2, rho3, rho4, omega4,
             p=.50, r21=0, r22=0, r23=0, r2t4=0, g4=0,
             n, J, K, L)

power.bcra4r3(es=.25, alpha=.05, two.tailed=TRUE,
              rho2, rho3, rho4, omega4,
              p=.50, r21=0, r22=0, r23=0, r2t4=0, g4=0,
              n, J, K, L)

mrss.bcra4r3(es=.25, power=.80, alpha=.05, two.tailed=TRUE,
             n, J, K, L0=10, tol=.10,
             rho2, rho3, rho4, omega4,
             p=.50, r21=0, r22=0, r23=0, r2t4=0, g4=0)
```

## Arguments

| | |
|---|---|
| `power` | statistical power $(1-\beta)$. |
| `es` | effect size. |
| `alpha` | probability of type I error. |
| `two.tailed` | logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing. |
| `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` | treatment effect heterogeneity as ratio of treatment effect variance among level 4 units to the residual variance at level 4. |
| `p` | average proportion of level 3 units randomly assigned to treatment within level 4 units. |
| `g4` | number of covariates at level 4. |
| `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 among level 4 units explained by level 4 covariates. |
| n | harmonic mean of level 1 units across level 2 units (or simple average). |
| J | harmonic mean of level 2 units across level 3 units (or simple average). |
| K | harmonic mean of level 3 units across level 4 units (or simple average). |
| L | number of level 4 units. |
| L0 | starting value for L. |
| tol | tolerance to end iterative process for finding L. |

## Value

| | |
|---|---|
| fun | function name. |
| parms | list of parameters used in power calculation. |
| df | degrees of freedom. |
| ncp | noncentrality parameter. |
| power | statistical power $(1-\beta)$. |
| mdes | minimum detectable effect size. |
| L | number of level 4 units. |

## See Also

cosa.bcrd4r3

## Examples

```
# cross-checks
mdes.bcra4r3(rho4=.05, rho3=.15, rho2=.15,
             omega4=.50, n=10, J=4, K=4, L=20)
power.bcra4r3(es = .316, rho4=.05, rho3=.15, rho2=.15,
              omega4=.50, n=10, J=4, K=4, L=20)
mrss.bcra4r3(es = .316, rho4=.05, rho3=.15, rho2=.15,
             omega4=.50, n=10, J=4, K=4)
```

---

bira2c1 — *Two-Level Blocked (Constant Treatment Effect) Individual-level Random Assignment Design, Treatment at Level 1*

---

## Description

Use mdes.bira2c1() to calculate the minimum detectable effect size, power.bira2c1() to calculate the statistical power, and mrss.bira2c1() to calculate the minimum required sample size.

## Usage

```
mdes.bira2c1(power=.80, alpha=.05, two.tailed=TRUE,
             p=.50, g1=0, r21=0,
             n, J)

power.bira2c1(es=.25, alpha=.05, two.tailed=TRUE,
             p=.50, g1=0, r21=0,
             n, J)

mrss.bira2c1(es=.25, power=.80, alpha=.05, two.tailed=TRUE,
           n, J0=10, tol=.10,
           p=.50, g1=0, r21=0)
```

## Arguments

| | |
|---|---|
| power | statistical power $(1-\beta)$. |
| es | effect size. |
| alpha | probability of type I error. |
| two.tailed | logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing. |
| p | average proportion of level 1 units randomly assigned to treatment within level 2 units. |
| g1 | number of covariates at level 1. |
| r21 | proportion of level 1 variance in the outcome explained by level 1 covariates. |
| n | harmonic mean of level 1 units across level 2 units (or simple average). |
| J | level 2 sample size. |
| J0 | starting value for J. |
| tol | tolerance to end iterative process for finding J. |

## Value

| | |
|---|---|
| fun | function name. |
| parms | list of parameters used in power calculation. |
| df | degrees of freedom. |
| ncp | noncentrality parameter. |
| power | statistical power $(1-\beta)$. |
| mdes | minimum detectable effect size. |
| J | number of level 2 units. |

## Examples

```
# cross-checks
mdes.bira2c1(n=15, J=20)
power.bira2c1(es=.325, n=15, J=20)
mrss.bira2c1(es=.325, n=15)
```

---

bira2f1 *Two-Level Blocked (Fixed) Individual-level Random Assignment Design, Treatment at Level 1*

---

## Description

Use `mdes.bira2f1()` to calculate the minimum detectable effect size, `power.bira2f1()` to calculate the statistical power, and `mrss.bira2f1()` to calculate the minimum required sample size.

## Usage

```
mdes.bira2f1(power=.80, alpha=.05, two.tailed=TRUE,
             p=.50, g1=0, r21=0, n, J)

power.bira2f1(es=.25, alpha=.05, two.tailed=TRUE,
              p=.50, g1=0, r21=0, n, J)

mrss.bira2f1(es=.25, power=.80, alpha=.05, two.tailed=TRUE,
             n, J0=10, tol=.10,
             p=.50, g1=0, r21=0)
```

## Arguments

| | |
|---|---|
| `power` | statistical power $(1 - \beta)$. |
| `es` | effect size. |
| `alpha` | probability of type I error. |
| `two.tailed` | logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing. |
| `p` | average proportion of level 1 units randomly assigned to treatment within level 2 units. |
| `g1` | number of covariates at level 1. |
| `r21` | proportion of level 1 variance in the outcome explained by level 1 covariates. |
| `n` | harmonic mean of level 1 units across level 2 units (or simple average). |
| `J` | level 2 sample size. |
| `J0` | starting value for `J`. |
| `tol` | tolerance to end iterative process for finding `J`. |

## Value

| | |
|---|---|
| `fun` | function name. |
| `parms` | list of parameters used in power calculation. |
| `df` | degrees of freedom. |
| `ncp` | noncentrality parameter. |

power
: statistical power $(1 - \beta)$.

mdes
: minimum detectable effect size.

J
: number of level 2 units.

## Examples

```
# cross-checks
mdes.bira2f1(n=15, J=20)
power.bira2f1(es=.325, n=15, J=20)
mrss.bira2f1(es=.325, n=15)
```

---

bira2r1 *Two-Level Blocked Individual-level Random Assignment Design, Treatment at Level 1*

---

## Description

Use `mdes.bira2r1()` to calculate the minimum detectable effect size, `power.bira2r1()` to calculate the statistical power, and `mrss.bira2r1()` to calculate the minimum required sample size.

## Usage

```
mdes.bira2r1(power=.80, alpha=.05, two.tailed=TRUE,
             rho2,  omega2, p=.50, g2=0, r21=0, r2t2=0,
             n, J)

power.bira2r1(es=.25, alpha=.05, two.tailed=TRUE,
              rho2,  omega2, g2=0, p=.50, r21=0, r2t2=0,
              n, J)

mrss.bira2r1(es=.25, power=.80, alpha=.05, two.tailed=TRUE,
             n, J0=10, tol=.10,
             rho2, omega2, g2=0, p=.50, r21=0, r2t2=0)
```

## Arguments

power
: statistical power $(1 - \beta)$.

es
: effect size.

alpha
: probability of type I error.

two.tailed
: logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing.

rho2
: proportion of variance in the outcome between level 2 units (unconditional ICC2).

omega2
: treatment effect heterogeneity as ratio of treatment effect variance among level 2 units to the residual variance at level 2.

p
: average proportion of level 1 units randomly assigned to treatment within level 2 units.

| | |
|---|---|
| g2 | number of covariates at level 2. |
| r21 | proportion of level 1 variance in the outcome explained by level 1 covariates. |
| r2t2 | proportion of treatment effect variance among level 2 units explained by level 2 covariates. |
| n | harmonic mean of level 1 units across level 2 units (or simple average). |
| J | level 2 sample size. |
| J0 | starting value for J. |
| tol | tolerance to end iterative process for finding J. |

## Value

| | |
|---|---|
| fun | function name. |
| parms | list of parameters used in power calculation. |
| df | degrees of freedom. |
| ncp | noncentrality parameter. |
| power | statistical power $(1 - \beta)$. |
| mdes | minimum detectable effect size. |
| J | number of level 2 units. |

## See Also

cosa.bird2r1

## Examples

```
# cross-checks
mdes.bira2r1(rho2=.17, omega2=.50, n=15, J=20)
power.bira2r1(es=.366, rho2=.17, omega2=.50, n=15, J=20)
mrss.bira2r1(es=.366, rho2=.17, omega2=.50, n=15)
```

---

# bira3r1 *Three-Level Blocked Individual-level Random Assignment Design, Treatment at Level 1*

---

## Description

Use `mdes.bira3r1()` to calculate the minimum detectable effect size, `power.bira3r1()` to calculate the statistical power, and `mrss.bira3r1()` to calculate the minimum required sample size.

### Usage

```
mdes.bira3r1(power=.80, alpha=.05, two.tailed=TRUE,
             rho2, rho3, omega2, omega3,
             p=.50, r21=0, r2t2=0, r2t3=0, g3=0,
             n, J, K)

power.bira3r1(es=.25, alpha=.05, two.tailed=TRUE,
              rho2, rho3, omega2, omega3,
              p=.50, r21=0, r2t2=0, r2t3=0, g3=0,
              n, J, K)

mrss.bira3r1(es=.25, power=.80, alpha=.05, two.tailed=TRUE,
             n, J, K0=10, tol=.10,
             rho2, rho3, omega2, omega3,
             p=.50, r21=0, r2t2=0, r2t3=0, g3=0)
```

### Arguments

| | |
|---|---|
| `power` | statistical power $(1-\beta)$. |
| `es` | effect size. |
| `alpha` | probability of type I error. |
| `two.tailed` | logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing. |
| `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` | treatment effect heterogeneity as ratio of treatment effect variance among level 2 units to the residual variance at level 2. |
| `omega3` | treatment effect heterogeneity as ratio of treatment effect variance among level 3 units to the residual variance at level 3. |
| `p` | average proportion of level 1 units randomly assigned to treatment within level 2 units. |
| `g3` | number of covariates at level 3. |
| `r21` | proportion of level 1 variance in the outcome explained by level 1 covariates. |
| `r2t2` | proportion of treatment effect variance among level 2 units explained by level 2 covariates. |
| `r2t3` | proportion of treatment effect variance among level 3 units explained by level 3 covariates. |
| `n` | harmonic mean of level 1 units across level 2 units (or simple average). |
| `J` | harmonic mean of level 2 units across level 3 units (or simple average). |
| `K` | number of level 3 units. |
| `K0` | starting value for `K`. |
| `tol` | tolerance to end iterative process for finding `K`. |

## Value

| | |
|---|---|
| fun | function name. |
| parms | list of parameters used in power calculation. |
| df | degrees of freedom. |
| ncp | noncentrality parameter. |
| power | statistical power $(1-\beta)$. |
| mdes | minimum detectable effect size. |
| K | number of level 3 units. |

## See Also

cosa.bird3r1

## Examples

```
# cross-checks
mdes.bira3r1(rho3=.20, rho2=.15,
             omega3=.10, omega2=.10,
             n=69, J=10, K=100)
power.bira3r1(es = .045, rho3=.20, rho2=.15,
              omega3=.10, omega2=.10,
              n=69, J=10, K=100)
mrss.bira3r1(es = .045, rho3=.20, rho2=.15,
             omega3=.10, omega2=.10,
             n=69, J=10)
```

---

bira4r1 *Four-Level Blocked Individual-level Random Assignment Design, Treatment at Level 1*

---

## Description

Use mdes.bira4r1() to calculate the minimum detectable effect size, power.bira4r1() to calculate the statistical power, and mrss.bira4r1() to calculate the minimum required sample size.

## Usage

```
mdes.bira4r1(power=.80, alpha=.05, two.tailed=TRUE,
             rho2, rho3, rho4, omega2, omega3, omega4,
             p=.50, r21=0, r2t2=0, r2t3=0, r2t4=0, g4=0,
             n, J, K, L)

power.bira4r1(es=.25, alpha=.05, two.tailed=TRUE,
              rho2, rho3, rho4, omega2, omega3, omega4,
              p=.50, r21=0, r2t2=0, r2t3=0, r2t4=0, g4=0,
              n, J, K, L)
```

```
mrss.bira4r1(es=.25, power=.80, alpha=.05, two.tailed=TRUE,
             n, J, K, L0=10, tol=.10,
             rho2, rho3, rho4, omega2, omega3, omega4,
             p=.50, r21=0, r2t2=0, r2t3=0, r2t4=0, g4=0)
```

## Arguments

| | |
|---|---|
| `power` | statistical power $(1 - \beta)$. |
| `es` | effect size. |
| `alpha` | probability of type I error. |
| `two.tailed` | logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing. |
| `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` | treatment effect heterogeneity as ratio of treatment effect variance among level 2 units to the residual variance at level 2. |
| `omega3` | treatment effect heterogeneity as ratio of treatment effect variance among level 3 units to the residual variance at level 3. |
| `omega4` | treatment effect heterogeneity as ratio of treatment effect variance among level 4 units to the residual variance at level 4. |
| `p` | average proportion of level 1 units randomly assigned to treatment within level 2 units. |
| `g4` | number of covariates at level 4. |
| `r21` | proportion of level 1 variance in the outcome explained by level 1 covariates. |
| `r2t2` | proportion of treatment effect variance among level 2 units explained by level 2 covariates. |
| `r2t3` | proportion of treatment effect variance among level 3 units explained by level 3 covariates. |
| `r2t4` | proportion of treatment effect variance among level 4 units explained by level 4 covariates. |
| `n` | harmonic mean of level 1 units across level 2 units (or simple average). |
| `J` | harmonic mean of level 2 units across level 3 units (or simple average). |
| `K` | harmonic mean of level 3 units across level 4 units (or simple average). |
| `L` | number of level 4 units. |
| `L0` | starting value for `L`. |
| `tol` | tolerance to end iterative process for finding `L`. |

### Value

| | |
|---|---|
| fun | function name. |
| parms | list of parameters used in power calculation. |
| df | degrees of freedom. |
| ncp | noncentrality parameter. |
| power | statistical power $(1-\beta)$. |
| mdes | minimum detectable effect size. |
| L | number of level 4 units. |

### See Also

cosa.bird4r1

### Examples

```
# cross-checks
mdes.bira4r1(rho4=.05, rho3=.15, rho2=.15,
             omega4=.50, omega3=.50, omega2=.50,
             n=10, J=4, K=4, L=27)
power.bira4r1(es = 0.142, rho4=.05, rho3=.15, rho2=.15,
              omega4=.50, omega3=.50, omega2=.50,
              n=10, J=4, K=4, L=27)
mrss.bira4r1(es = 0.142, rho4=.05, rho3=.15, rho2=.15,
             omega4=.50, omega3=.50, omega2=.50,
             n=10, J=4, K=4)
```

---

## conversion *Object Conversion*

---

### Description

Use `mrss.to.mdes()` to convert an object returned from MRSS functions into an object returned from MDES functions, `mrss.to.power()` to convert an object returned from MRSS functions into an object returned from power functions, `power.to.mdes()` to convert an object returned from power functions into an object returned from MDES functions, `mdes.to.power()` to convert an object returned from MDES functions into an object returned from power functions, and `mdes.to.pctl()` to convert effect sizes or an object returned from MDES functions into percentiles.

### Usage

```
mrss.to.mdes(object)
mrss.to.power(object)
power.to.mdes(object)
mdes.to.power(object)
mdes.to.pctl(object)
```

### Arguments

| | |
|---|---|
| `object` | an object returned from one of the functions in **PowerUpR** package. |

### Examples

```
design1 <- power.bira2r1(es=.15, rho2=.35, omega2=.10, n=83, J=10)
design2 <- power.to.mdes(design1)
mdes.to.pctl(design2)
```

---

## cra2r2 — *Two-level Cluster-randomized Trials to Detect Main, Moderation and Mediation Effects*

---

### Description

Use `mdes.<design>()` to calculate minimum detectable effect size for the main effect, `mdesd.<design>()` to calculate minimum detectable effect size difference for the moderation effect, `power.<design>()` to calculate the statistical power, and `mrss.<design>()` to calculate the minimum required sample size. Use `<output>.cra2r2()` for the main effect, `<output>.mod221()` for the moderator at level 1, `<output>.mod222()` for the moderator at level 2. Use `power.med211()` for 2-1-1 mediation, and `power.med221()` for 2-2-1 mediation.

### Usage

```
mdes.cra2r2(power=.80, alpha=.05, two.tailed=TRUE,
            rho2, p=.50, g2=0, r21=0, r22=0,
            n, J)

mdesd.mod221(power=.80, alpha=.05, two.tailed=TRUE,
            rho2, omegam2, g1=0, r21=0, r2m2=0,
            p=.50, q=NULL, n, J)

mdesd.mod222(power=.80, alpha=.05, two.tailed=TRUE,
            rho2, g2=0, r21=0, r22=0,
            p=.50, q=NULL, n, J)

power.cra2r2(es=.25, alpha=.05, two.tailed=TRUE,
             rho2, g2=0, p=.50, r21=0, r22=0,
             n, J)

power.mod221(es=.25, alpha=.05, two.tailed=TRUE,
             rho2, omegam2, g1=0, r21=0, r2m2=0,
             p=.50, q=NULL, n, J)

power.mod222(es=.25, alpha=.05, two.tailed=TRUE,
             rho2, g2=0, r21=0, r22=0,
             p=.50, q=NULL, n, J)
```

```
power.med211(esa, esb1, esB, escp, two.tailed = TRUE, alpha = .05,
             mc = FALSE, nsims = 1000, ndraws = 1000,
             rhom2, rho2, r21, r22, r2m1, r2m2,
             p, n, J)

power.med221(esa, esb, escp, two.tailed = TRUE, alpha = .05,
             mc = FALSE, nsims = 1000, ndraws = 1000,
             rho2, r22, r21, r2m2,
             p = .50, n, J)

mrss.cra2r2(es=.25, power=.80, alpha=.05, two.tailed=TRUE,
            n, J0=10, tol=.10,
            rho2, g2=0, p=.50, r21=0, r22=0)

mrss.mod221(es=.25, power=.80, alpha=.05, two.tailed=TRUE,
            n, J0=10, tol=.10, rho2, omegam2, g1=0, r21=0, r2m2=0,
            p=.50, q=NULL)

mrss.mod222(es=.25, power=.80, alpha=.05, two.tailed=TRUE,
            n, J0=10, tol=.10, rho2, g2=0, r21=0, r22=0,
            p=.50, q=NULL)

mrss.mod222(es=.25, power=.80, alpha=.05, two.tailed=TRUE,
            n, J0=10, tol=.10, rho2, g2=0, r21=0, r22=0,
            p=.50, q=NULL)
```

## Arguments

| | |
|---|---|
| power | statistical power $(1 - \beta)$ |
| es, esa, esb, esb1, esB, escp | effect size for main/moderator effects, or for path coefficients a (treatment - mediator), b (level 2 mediator - outcome), b1 (level 1 mediator - outcome), B (overall mediator - outcome) or cp (direct treatment - outcome) in the mediation model. |
| alpha | probability of type I error. |
| two.tailed | logical; FALSE for one-tailed hypothesis testing. |
| rho2 | proportion of variance in the outcome between level 2 units (unconditional ICC2). |
| rhom2 | proportion of variance in the mediator between level 2 units. |
| omegam2 | ratio of the unconditional variance in the moderator effect that is between level 2 units to the residual variance between level 2 units in the null model. |
| p | proportion of level 2 units randomly assigned to treatment. |
| q | proportion of level 1 or level 2 units in the moderator subgroup. |
| g1 | number of covariates at level 1. |
| 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. |
| r2m1 | proportion of mediator variance at level 1 explained by level 1 covariates. |
| r2m2 | proportion of variance in the moderator (or mediator) effect that is explained by level 2 predictors. For the mediation model, proportion of mediator variance at level 2 explained by level 2 predictors. |
| n | harmonic mean of level 1 units across level 2 units (or simple average). |
| J | level 2 sample size. |
| J0 | starting value for J. |
| tol | tolerance to end iterative process for finding J. |
| mc | logical; TRUE for monte carlo simulation based power. |
| nsims | number of replications, if mc = TRUE. |
| ndraws | number of draws from the distribution of the path coefficients for each replication, if mc = TRUE. |

## Value

| | |
|---|---|
| fun | function name. |
| parms | list of parameters used in power calculation. |
| df | degrees of freedom. |
| ncp | noncentrality parameter. |
| power | statistical power $(1-\beta)$. |
| mdes | minimum detectable effect size. |
| J | number of level 2 units. |

## See Also

For a more flexible sample size determination see cosa.crd2r2.

## Examples

```
# cross-checks for the main effect
mdes.cra2r2(rho2=.17, n=15, J=20)
power.cra2r2(es=.629, rho2=.17, n=15, J=20)
mrss.cra2r2(es=.629, rho2=.17, n=15)

# cross-checks for the randomly varying cont. L1 moderator effect
mdesd.mod221(rho2=.17, omegam2=.10, n=15, J=20)
power.mod221(es=.3563, rho2=.17, omegam2 =.10, n=15, J=20)
mrss.mod221(es=.3563, rho2=.17, omegam2 =.10, n=15)

# cross-checks for the non-randomly varying cont. L1 moderator effect
mdesd.mod221(rho2=.17, omegam2=0, n=15, J=20)
power.mod221(es=0.2957, rho2=.17, omegam2 =0, n=15, J=20)
mrss.mod221(es=0.2957, rho2=.17, omegam2 =0, n=15)

# cross-checks for the randomly varying bin. L1 moderator effect
```

```
mdesd.mod221(rho2=.17, omegam2=.10, q=.50, n=15, J=20)
power.mod221(es=.647, rho2=.17, omegam2 =.10, q=.50, n=15, J=20)
mrss.mod221(es=.647, rho2=.17, omegam2 =.10, q=.50, n=15)

# cross-checks for the non-randomly varying bin. L1 moderator effect
mdesd.mod221(rho2=.17, omegam2=0, q=.50, n=15, J=20)
power.mod221(es=0.5915, rho2=.17, omegam2 =0, q=.50, n=15, J=20)
mrss.mod221(es=0.5915, rho2=.17, omegam2 =0, q=.50, n=15)

# cross-checks for the cont. L2 moderator effect
mdesd.mod222(rho2=.17, n=15, J=100)
power.mod222(es=0.2742, rho2=.17, n=15, J=100)
mrss.mod222(es=0.2742, rho2=.17, n=15)

# cross-checks for the bin. L2 moderator effect
mdesd.mod222(rho2=.17, q=.50, n=15, J=100)
power.mod222(es=0.5485, rho2=.17, q=.50, n=15, J=100)
mrss.mod222(es=0.5485, rho2=.17, q=.50, n=15)

# 2-2-1 mediation
power.med221(esa=0.6596, esb=0.1891, escp=.1,
             rho2=.15, r22=.52, r21=.40, r2m2=.50,
             n=100, J=40, p=.5)

# 2-1-1 mediation
power.med211(esa=0.4135, esb1=0.0670, esB=0.3595, escp=.1,
            rhom2=.3, rho2=.3, r22=.6, r21=.6, r2m2=.6, r2m1=.6,
            n=30, J=80, p=.1)
```

---

cra3r3 — *Three-level Cluster-randomized Trials to Detect Main, Moderation, and Mediation Effects*

---

## Description

Use `mdes.<design>()` to calculate the minimum detectable effect size for the main effect, `mdesd.<design>()` to calculate the minimum detectable effect size difference for the moderation effect, `power.<design>()` to calculate the statistical power, and `mrss.<design>()` to calculate the minimum required sample size. Use `<output>.cra3r3()` for the main effect, `<output>.mod331()` for the moderator at level 1, `<output>.mod332()` for the moderator at level 2, `<output>.mod333()` for the moderator at level 3. Use`power.med321()` for 3-2-1 mediation.

## Usage

```
mdes.cra3r3(power=.80, alpha=.05, two.tailed=TRUE,
            rho2, rho3, p=.50, g3=0, r21=0, r22=0, r23=0,
            n, J, K)

mdesd.mod331(power=.80, alpha=.05, two.tailed=TRUE,
```

```
             rho2, rho3, omegam2=0, omegam3=0,
             g1=0, r21=0, r2m2=0, r2m3=0,
             p=.50, q=NULL, n, J, K)

mdesd.mod332(power=.80, alpha=.05, two.tailed=TRUE,
             rho2, rho3, omegam3, g2=0, r21=0, r22=0, r2m3=0,
             p=.50, q=NULL, n, J, K)

mdesd.mod333(power=.80, alpha=.05, two.tailed=TRUE,
             rho2, rho3,  g3=0, r21=0, r22=0, r23=0,
             p=.50, q=NULL, n, J, K)

power.cra3r3(es=.25, alpha=.05, two.tailed=TRUE,
              rho2, rho3, g3=0, r21=0, r22=0, r23=0,
              p=.50, n, J, K)

power.mod331(es=.25, alpha=.05, two.tailed=TRUE,
              rho2, rho3, omegam2, omegam3,
              g1=0, r21=0, r2m2=0, r2m3=0,
              p=.50, q=NULL, n, J, K)

power.mod332(es=.25, alpha=.05, two.tailed=TRUE,
              rho2, rho3, omegam3, g2=0, r21=0, r22=0, r2m3=0,
              p=.50, q=NULL, n, J, K)

power.mod333(es=.25, alpha=.05, two.tailed=TRUE,
              rho2, rho3, g3=0, r21=0, r22=0, r23=0,
              p=.50, q=NULL, n, J, K)

power.med321(esa, esB, two.tailed=TRUE, alpha=.05,
              mc=FALSE, nsims=1000, ndraws=1000,
              rhom3, rho2, rho3, r2m2, r2m3, r21, r22, r23,
              p=.50, n, J, K)

mrss.cra3r3(es=.25, power=.80, alpha=.05, two.tailed=TRUE,
             n, J, K0=10, tol=.10,
             rho2, rho3, p=.50, g3=0, r21=0, r22=0, r23=0)

mrss.mod331(es=.25, power=.80, alpha=.05, two.tailed=TRUE,
             rho2, rho3, omegam2, omegam3,
             g1=0, r21=0, r2m2=0, r2m3=0,
             p=.50, q=NULL, n, J, K0=10, tol=.10)

mrss.mod332(es=.25, power=.80, alpha=.05, two.tailed=TRUE,
             rho2, rho3, omegam3, g2=0, r21=0, r22=0, r2m3=0,
             p=.50, q=NULL, n, J, K0=10, tol=.10)

mrss.mod333(es=.25, power=.80, alpha=.05, two.tailed=TRUE,
```

```
rho2, rho3, g3=0, r21=0, r22=0, r23=0,
p=.50, q=NULL, n, J, K0=10, tol=.10)
```

## Arguments

| | |
|---|---|
| power | statistical power $(1-\beta)$. |
| es, esa, esB | effect size for main/moderator effects, or for path coefficients a (treatment - mediator), or B (overall mediator - outcome) in the mediation model. |
| alpha | probability of type I error. |
| two.tailed | logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing. |
| 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). |
| rhom3 | proportion of variance in the mediator between level 3 units. |
| omegam2 | ratio of the unconditional variance in the moderator effect that is between level 2 units to the residual variance between level 2 units in the null model. |
| omegam3 | ratio of the unconditional variance in the moderator effect that is between level 3 units to the residual variance between level 3 units in the null model. |
| p | proportion of level 3 units randomly assigned to treatment. |
| q | proportion of level 1, level 2, or level 3 units in the moderator subgroup. |
| g1 | number of covariates at level 1. |
| g2 | number of covariates at level 2. |
| 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. |
| r2m2 | proportion of variance in the moderator (or mediator) effect that is explained by level 2 predictors. For the mediation model, proportion of mediator variance at level 2 explained by level 2 predictors. |
| r2m3 | proportion of variance in the moderator (or mediator) effect that is explained by level 3 predictors. For the mediation model, proportion of aggregated mediator variance at level 3 explained by level 3 predictors. |
| n | harmonic mean of level 1 units across level 2 units (or simple average). |
| J | harmonic mean of level 2 units across level 3 units (or simple average). |
| K | level 3 sample size. |
| K0 | starting value for K. |
| tol | tolerance to end iterative process for finding K. |
| mc | logical; TRUE for monte carlo simulation based power. |
| nsims | number of replications, if mc = TRUE. |
| ndraws | number of draws from the distribution of the path coefficients for each replication, if mc = TRUE. |

## Value

| | |
|---|---|
| fun | function name. |
| parms | list of parameters used in power calculation. |
| df | degrees of freedom. |
| ncp | noncentrality parameter. |
| power | statistical power $(1-\beta)$. |
| mdes | minimum detectable effect size. |
| K | number of level 3 units. |

## See Also

For a more flexible sample size determination see `cosa.crd3r3`.

## Examples

```
# cross-checks for the main effect
mdes.cra3r3(rho3=.06, rho2=.17, n=15, J=3, K=60)
power.cra3r3(es=.269, rho3=.06, rho2=.17, n=15, J=3, K=60)
mrss.cra3r3(es=.269, rho3=.06, rho2=.17, n=15, J=3)

# cross-checks for the randomly varying cont. L1 moderator effect
mdes.mod331(power=.80, alpha=.05, two.tailed=TRUE,
            rho2=.17, rho3=.06, omegam2=.10, omegam3=.10,
            q=NULL, n=15, J=3, K=60)
power.mod331(es=0.1248, alpha=.05, two.tailed=TRUE,
            rho2=.17, rho3=.06, omegam2=.10, omegam3=.10,
            q=NULL, n=15, J=3, K=60)
mrss.mod331(es=0.1248, alpha=.05, two.tailed=TRUE,
            rho2=.17, rho3=.06, omegam2=.10, omegam3=.10,
            q=NULL, n=15, J=3)

# cross-checks for the non-randomly varying cont. L1 moderator effect
mdesd.mod331(power=.80, alpha=.05, two.tailed=TRUE,
            rho2=.17, rho3=.06, omegam2=0, omegam3=0,
            q=NULL, n=15, J=3, K=60)
power.mod331(es=.0946, alpha=.05, two.tailed=TRUE,
            rho2=.17, rho3=.06, omegam2=0, omegam3=0,
            q=NULL, n=15, J=3, K=60)
mrss.mod331(es=.0946, alpha=.05, two.tailed=TRUE,
            rho2=.17, rho3=.06, omegam2=0, omegam3=0,
            q=NULL, n=15, J=3)

# cross-checks for the randomly varying bin. L1 moderator effect
mdesd.mod331(power=.80, alpha=.05, two.tailed=TRUE,
            rho2=.17, rho3=.06, omegam2=.10, omegam3=.10,
            q=.50, n=15, J=3, K=60)
power.mod331(es=.2082, alpha=.05, two.tailed=TRUE,
            rho2=.17, rho3=.06, omegam2=.10, omegam3=.10,
            q=.50, n=15, J=3, K=60)
```

```
mrss.mod331(es=.2082, alpha=.05, two.tailed=TRUE,
            rho2=.17, rho3=.06, omegam2=.10, omegam3=.10,
            q=.50, n=15, J=3)

# cross-checks for the non-randomly varying bin. L1 moderator effect
mdesd.mod331(power=.80, alpha=.05, two.tailed=TRUE,
            rho2=.17, rho3=.06, omegam2=0, omegam3=0,
            q=.50, n=15, J=3, K=60)
power.mod331(es=.1893, alpha=.05, two.tailed=TRUE,
            rho2=.17, rho3=.06, omegam2=0, omegam3=0,
            q=.50, n=15, J=3, K=60)
mrss.mod331(es=.1893, alpha=.05, two.tailed=TRUE,
            rho2=.17, rho3=.06, omegam2=0, omegam3=0,
            q=.50, n=15, J=3)

# 3-2-1 mediation
power.med321(esa= .51, esB = .30, rhom3 = 0.27, rho2 = .15, rho3 = .19,
             r2m2 = .07, r2m3 = .16, r21 = .02, r22 = .41, r23 = .38,
             p = .50, n = 20, J = 4, K = 60)
```

---

cra4r4 *Four-Level Cluster-randomized Trial*

---

## Description

use `mdes.cra4r4()` calculate the minimum detectable effect size, `power.cra4r4()` to calculate the statistical power, and `mrss.cra4r4()` to calculate the minimum required sample size.

## Usage

```
mdes.cra4r4(power=.80, alpha=.05, two.tailed=TRUE,
            rho2, rho3, rho4, p=.50, r21=0, r22=0, r23=0, r24=0, g4=0,
            n, J, K, L)

power.cra4r4(es=.25, alpha=.05, two.tailed=TRUE,
             rho2, rho3, rho4, p=.50, r21=0, r22=0, r23=0, r24=0, g4=0,
             n, J, K, L)

mrss.cra4r4(es=.25, power=.80, alpha=.05, two.tailed=TRUE,
            n, J, K, L0=10, tol=.10,
            rho2, rho3, rho4, p=.50,
            r21=0, r22=0, r23=0, r24=0, g4=0)
```

## Arguments

| | |
|---|---|
| `power` | statistical power $(1 - \beta)$. |
| `es` | effect size. |
| `alpha` | probability of type I error. |

| | |
|---|---|
| two.tailed | logical; TRUE for two-tailed hypothesis testing, FALSE for one-tailed hypothesis testing. |
| 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). |
| p | proportion of level 4 units randomly assigned to treatment. |
| 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. |
| n | harmonic mean of level 1 units across level 2 units (or simple average). |
| J | harmonic mean of level 2 units across level 3 units (or simple average). |
| K | harmonic mean of level 3 units across level 4 units (or simple average). |
| L | number of level 4 units. |
| L0 | starting value for L. |
| tol | tolerance to end iterative process for finding L. |

## Value

| | |
|---|---|
| fun | function name. |
| parms | list of parameters used in power calculation. |
| df | degrees of freedom. |
| ncp | noncentrality parameter. |
| power | statistical power $(1-\beta)$. |
| mdes | minimum detectable effect size. |
| L | number of level 4 units. |

## See Also

cosa.crd4r4

## Examples

```
# cross-checks
mdes.cra4r4(rho4=.05, rho3=.05, rho2=.10,
            n=10, J=2, K=3, L=20)
power.cra4r4(es = .412, rho4=.05, rho3=.05, rho2=.10,
             n=10, J=2, K=3, L=20)
mrss.cra4r4(es = .412, rho4=.05, rho3=.05, rho2=.10,
            n=10, J=2, K=3)
```

ira1r1 *Individual-level Random Assignment Design*

## Description

Use `mdes.ira1r1()` to calculate minimum detectable effect size, `power.ira1r1()` to calculate statistical power, and `mrss.ira1r1()` to calculate minimum required sample size.

## Usage

```
mdes.ira1r1(power=.80, alpha=.05, two.tailed=TRUE,
            p=.50, g1=0, r21=0, n)

power.ira1r1(es=.25, alpha=.05, two.tailed=TRUE,
             p=.50, g1=0, r21=0, n)

mrss.ira1r1(es=.25, power=.80, alpha=.05, two.tailed=TRUE,
            n0=10, tol=.10,
            p=.50, g1=0, r21=0)
```

## Arguments

| | |
|---|---|
| `power` | statistical power $(1-\beta)$. |
| `es` | effect size. |
| `alpha` | probability of type I error. |
| `two.tailed` | logical; `TRUE` for two-tailed hypothesis testing, `FALSE` for one-tailed hypothesis testing. |
| `p` | proportion of units randomly assigned to treatment. |
| `g1` | number of covariates. |
| `r21` | proportion of variance in the outcome explained by covariates. |
| `n` | sample size. |
| `n0` | starting value for `n`. |
| `tol` | tolerance to end iterative process for finding `n`. |

## Value

| | |
|---|---|
| `fun` | function name. |
| `parms` | list of parameters used in power calculation. |
| `df` | degrees of freedom. |
| `ncp` | noncentrality parameter. |
| `power` | statistical power $(1-\beta)$. |
| `mdes` | minimum detectable effect size. |
| `n` | sample size. |

## See Also

power.ird1r1

## Examples

```
# cross-checks
mdes.ira1r1(n=250)
power.ira1r1(es=.356, n=250)
mrss.ira1r1(es=.356)
```

---

plots *Plots*

---

## Description

Plots statistical power, minimum detectable effect size (MDES), or MDES difference (MDESD) curves with (1-$\alpha$)x100 % confidence interval.

## 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, ...)

  ## 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, ...)

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

## Arguments

| | |
|---|---|
| x | an object returned from one of the PowerUpR functions. |
| 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 (ignored for objects returned from power.med211(), power.med221(), and power.med321() functions). |
| ylab | y axis label (ignored for objects returned from power.med211(), power.med221(), and power.med321() functions). |
| main | title for the plot (ignored for objects returned from power.med221() and power.med211() functions). |
| sub | subtitle for the plot (ignored for objects returned from power.med221() and power.med211() functions). |
| locate | logical; TRUE locates parameter values for design x on the plot. |
| ... | other graphical parameters to pass to plot.new(). |

## Examples

```
design1 <- mdes.cra3r3(rho3=.06, rho2=.17, n=15, J=3, K=60)
plot(design1, ypar = "mdes", xpar = "K", xlim = c(30, 100))
plot(design1, ypar = "power", xpar = "K", xlim = c(30, 100))

design2 <- power.cra3r3(es=.269, rho3=.06, rho2=.17, n=15, J=3, K=60)
plot(design2, ypar = "mdes", xpar = "K", xlim = c(30, 100))
plot(design2, ypar = "power", xpar = "K", xlim = c(30, 100))
```

---

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

---

## Description

Experimental MDES functions for 2-1-1 and 2-2-1 mediations are removed.

## Format

Deprecated or defunct functions are no longer documented.

## Details

Defunct functions:

- mdes.med211 is defunct, there is no replacement function
- mdes.med221 is defunct, there is no replacement function

t1t2.error *Plots Type I and Type II Error Rates*

## Description

t1t2.error plots Type I ($\alpha$) and Type II ($\beta$) error rates using central and noncentral t distributions for any objects returned from one of the PowerUpR functions.

## Usage

```
t1t2.error(object)
```

## Arguments

object an object returned from one of the PowerUpR functions.

## Examples

```
## Not run:

  design1 <- mdes.bira2r1(rho2=.35, omega2=.10,
                          n=83, J=480)
  t1t2.error(design1)


## End(Not run)
```

# Index