Package 'PSSTools'

January 13, 2024

Type Package

Title PSSTools: Power and Sample Size Tools

Version 0.1.0

Description This package accompanies the book Power and Sample Size in R by
Catherine M. Crespi for calculating power and sample size requirements for designing studies.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

Suggests designsize,
longpower,
npsurvSS,
powerSurvEpi,
rPowerSampleSize,
ssanv,
testthat (>= 3.0.0)

Config/testthat/edition 3

Imports magrittr,
mvtnorm,
plyr,
PowerTOST,
psych,
stats

Depends R (>= 2.10)

R topics documented:

`crossover_power` *Cross Over Power Calculation*

Description

Calculate power for either repeated cross sectional or closed cohort cross over designs.

Usage

```
crossover_power(design = "RCS", alpha, J, m, rho, rho_b, xi = 0, d)
```

Arguments

`design`	Either "RCS" for repeated cross-sectional design or "Cohort" for closed cohort design. Default is "RCS".
`alpha`	The significance level or type 1 error rate
`J`	Number of clusters
`m`	The number of individuals in each cluster at each time period
`rho`	Intraclass correlation coefficient (ICC)
`rho_b`	Between period ICC (correlation between outcomes of two individuals in the same cluster but different time periods)
`xi`	Within-cluster, within-subject correlation (correlation between two measurements within the same subject). Note this is different from subject autocorrelation. We expect xi to be larger than rho and rho_b. This is 0 for RCS cross over designs.
`d`	The standardized effect size

Value

power

Examples

```
# Repeated cross sectional cross over power calculation as in Example 12.3
crossover_power(design = "RCS", alpha = 0.05, J = 8, m = 30, rho = 0.05,
                rho_b = 0.025, xi = 0, d = 0.3)
# Closed cohort cross over power calculation as i Example 12.3
crossover_power(design = "Cohort", alpha = 0.05, J = 8, m = 30, rho = 0.05,
                rho_b = 0.025, xi = .4, d = 0.3)
```

`crossover_ss` *Cross Over Sample Size Calculation*

Description

Calculate sample size requirements for either repeated cross sectional or closed cohort cross over designs.

Usage

```
crossover_ss(design = "RCS", alpha, power, m, rho, rho_b, xi = 0, d)
```

Arguments

`design`	Either "RCS" for repeated cross-sectional design or "Cohort" for closed cohort design. Default is "RCS".
`alpha`	The significance level or type 1 error rate
`power`	Specified power level to achieve
`m`	The number of individuals in each cluster at each time period
`rho`	Intraclass correlation coefficient (ICC)

rho_b	Between period ICC (correlation between outcomes of two individuals in the same cluster but different time periods)
xi	Within-cluster, within-subject correlation (correlation between two measurements within the same subject). Note this is different from subject autocorrelation. We expect xi to be larger than rho and rho_b. This is 0 for RCS cross over designs.
d	The standardized effect size

Value

Calculated number of clusters required and adjusted number of clusters to account for normal approximation

Examples

```
# repeated cross sectional sample size as in Example 12.4
crossover_ss(design = "RCS", alpha = 0.05, power = 0.8, m = 30, rho = 0.05,
             rho_b = 0.025, xi = 0, d = 0.3)
# closed cohort sample size requirement
crossover_ss(design = "Cohort", alpha = 0.05, power = 0.8, m = 30,
             rho = 0.05, rho_b = 0.025, xi = 0.4, d = 0.3)
```

---

delta.sign *Delta Sign Table*

Description

A table documenting the sign of delta for tests comparing two parameters.

Usage

```
delta.sign
```

Format

‘delta.sign‘ A data frame with 3 rows and 3 columns:

columns Whether the test is for noninferiority of superirority by a margin

rows Whether a higher or lower parameter value is better

xij The sign of delta given the row & column `multendpoints_Ck_ss` *Algorithm and sample size for multiple co-primary endpoint clinical trials*

Description

Calculate the solution to Ck, the integral equation to calculate power for multiple endpoints introduced by Sozu et al in Sample Size Determination in Clinical Trials with Multiple Endpoints 2015.

Usage

```
multendpoints_Ck_ss(K, alpha, power, rho, gamma, a, r, delta)
```

Arguments

`K`	The number of K co-primary continuous endpoints. K >= 2.
`alpha`	The significance level.
`power`	The power level.
`rho`	The correlation matrix (K x K) that describes the relationship between each endpoint. Diagonal entries are 1.
`gamma`	The ratio of effect size(s). This can be either a value (K = 2) or a vector (K > 2).
`a`	A vector of length K. For continuous endpoints, a_1 = ... = a_K = 1. For binary endpoints using the chi-square method (without CC), this is a vector where each element is the ratio between sigma_k^star and sigma_k.
`r`	The allocation ratio between the number of people in the control and treatment condition (n_C / n_T).
`delta`	The standardized effect size (value or vector) for each endpoint.

Details

The sample size provided is the solution to equation (4.5) in the Sozu text. This is the simple formula provided for practical use to calculate sample size requirements as if there were a single endpoint. Specifically, this formula is n = (Ck + z_alpha)^2 / (kappa * delta_K^2)

Value

A dataframe with the solution to the Ck algorithm and the sample size requirements per group.

Examples

```
# Following Sozu et al Example in Appendix D.1 with 2 continuous endpoints with equal allocation
multendpoints_Ck_ss(K = 2, alpha = 0.025, power = 0.8,
                    rho = matrix(c(1, 0.5,
                                   0.5, 1), ncol = 2),
                    gamma = 8/7, a = c(1, 1), r = 1, delta = c(0.4, 0.35))
# Following Sozu et al Example in Appendix D.1 with 3 continuous endpoints and equal allocation
delta_vector <- c(0.5, 0.45, 0.4)
gamma_vector <- c(delta_vector[1]/delta_vector[3], delta_vector[2]/delta_vector[3])
rho_matrix <- matrix(c(1, 0.8, 0.8,
```

```
                        0.8, 1, 0.5,
                        0.8, 0.5, 1), ncol = 3)

multendpoints_Ck_ss(K = 3, alpha = 0.025, power = 0.8, rho = rho_matrix,
                    gamma = gamma_vector, a = c(1, 1, 1), r = 1,
                    delta = delta_vector)
```

`parallelCRT_power` *Parallel Cluster Randomized Trial Power Calculation*

Description

Parallel Cluster Randomized Trial Power Calculation

Usage

```
parallelCRT_power(alpha, J, m, d, rho, rho_c, rho_s)
```

Arguments

alpha	The significance level or type 1 error rate
J	Number of clusters
m	The number of individuals in each cluster at each time period
d	The standardized effect size
rho	Intraclass correlation coefficient (ICC)
rho_c	Cluster autocorrelation (proportion of cluster-level variance that is time invariant)
rho_s	Subject autocorrelation (This is 0 in repeated cross sectional designs because people are measured only once)

Value

power

Examples

```
# closed cohort design as in Example 12.1
parallelCRT_power(alpha = 0.05, J = 16, m = 30, d = 0.3, rho = 0.05, rho_c = 0.4, rho_s = 0.5)
# repeated cross sectional design as in Example 12.1
parallelCRT_power(alpha = 0.05, J = 16, m = 30, d = 0.3, rho = 0.05, rho_c = 0.4, rho_s = 0)
```

`parallelCRT_ss` *Parallel Cluster Randomized Trial Sample Size Calculation*

Description

Parallel Cluster Randomized Trial Sample Size Calculation

Usage

```
parallelCRT_ss(alpha, power, m, d, rho, rho_c, rho_s)
```

Arguments

alpha	The significance level or type 1 error rate
power	The specified level of power to achieve
m	The number of individuals in each cluster at each time period
d	The standardized effect size
rho	Intraclass correlation coefficient (ICC)
rho_c	Cluster autocorrelation (proportion of cluster-level variance that is time invariant)
rho_s	Subject autocorrelation (This is 0 in repeated cross sectional designs because people are measured only once)

Value

Calculated number of clusters required, adjusted number of clusters, and suggested number of clusters that includes the value of K

Examples

```
# example 12.2
parallelCRT_ss(alpha = 0.05, power = 0.8, m = 30, d = 0.3, rho = 0.05, rho_c = 0.4, rho_s = 0.5)
```

`pss.anova.f.es` *Cohen's f effect size calculation for one- or two- way analysis of variance*

Description

Cohen's f effect size calculation for one- or two- way analysis of variance

Usage

```
pss.anova.f.es(means = NULL, sd = NULL)
```

Arguments

means	A vector or matrix of group means.
sd	The estimated standard deviation within each group.

Value

A list of the arguments and various f effect sizes.

Examples

```
pss.anova.f.es(means = c(5, 10, 12), sd = 10)
mmatrix <- matrix(c(9.3, 8.9, 8.5, 8.7, 8.3, 7.9), nrow = 2, byrow = TRUE)
pss.anova.f.es(means = mmatrix, sd = 2)
```

`pss.anova1way.c.bal` *Power calculations for one-way balanced analysis of variance contrast test*

Description

Power calculations for one-way balanced analysis of variance contrast test

Usage

```
pss.anova1way.c.bal(
  n = NULL,
  mvec = NULL,
  cvec = NULL,
  sd = 1,
  Rsq = 0,
  ncov = 0,
  alpha = 0.05,
  power = NULL
)
```

Arguments

`n`	The sample size per group.
`mvec`	A vector of group means c(mu1, mu2, ...).
`cvec`	A vector of contrast coefficients c(c1, c2, ...).
`sd`	The estimated standard deviation within each group; defaults to 1.
`Rsq`	The estimated R^2 for regressing the outcome on the covariates; defaults to 0.
`ncov`	The number of covariates adjusted for in the model; defaults to 0.
`alpha`	The significance level or type 1 error rate; defaults to 0.05.
`power`	The specified level of power.

Value

A list of the arguments (including the computed one).

Examples

```
pss.anova1way.c.bal(n = 20, mvec = c(5, 10, 12), cvec = c(1, -1, 0), sd = 10, alpha = 0.025)
pss.anova1way.c.bal(n = 20, mvec = c(5, 10, 12), cvec = c(1, 0, -1), sd = 10, alpha = 0.025)
```

`pss.anova1way.c.unbal` *Power calculations for one-way unbalanced analysis of variance contrast test*

Description

Power calculations for one-way unbalanced analysis of variance contrast test

Usage

```
pss.anova1way.c.unbal(
  nvec = NULL,
  mvec = NULL,
  cvec = NULL,
  sd = NULL,
  Rsq = 0,
  ncov = 0,
  alpha = 0.05
)
```

Arguments

`nvec`	A vector of group sample sizes c(n1, n2, ...).
`mvec`	A vector of group mvec c(mu1, mu2, ...).
`cvec`	A vector of contrast coefficients c(c1, c2, ...).
`sd`	The estimated standard deviation within each group.
`Rsq`	The estimated R^2 for regressing the outcome on the covariates; defaults to 0.
`ncov`	The number of covariates adjusted for in the model; defaults to 0.
`alpha`	The significance level or type 1 error rate; defaults to 0.05.

Value

A list of the arguments (including the computed power).

Examples

```
pss.anova1way.c.unbal(nvec = c(20, 20, 20), mvec = c(5, 10, 12), cvec = c(1, -1, 0),
sd = 10, alpha = 0.025)
pss.anova1way.c.unbal(nvec = c(20, 20, 20), mvec = c(5, 10, 12), cvec = c(1, 0, -1),
sd = 10, alpha = 0.025)
```

`pss.anova1way.F.bal` *Power calculations for one-way balanced analysis of variance omnibus F test*

Description

Power calculations for one-way balanced analysis of variance omnibus F test

Usage

```
pss.anova1way.F.bal(
  n = NULL,
  mvec = NULL,
  sd = 1,
  Rsq = 0,
  ncov = 0,
  alpha = 0.05,
  power = NULL
)
```

Arguments

`n`	The sample size per group.
`mvec`	A vector of group means c(mu1, mu2, ...).
`sd`	The estimated standard deviation within each group; defaults to 1.
`Rsq`	The estimated R^2 for regressing the outcome on the covariates; defaults to 0.
`ncov`	The number of covariates adjusted for in the model; defaults to 0.
`alpha`	The significance level or type 1 error rate; defaults to 0.05.
`power`	The specified level of power.

Value

A list of the arguments (including the computed one).

Examples

```
pss.anova1way.F.bal(n = 20, mvec = c(5, 10, 12), sd = 10)
pss.anova1way.F.bal(n = NULL, mvec = c(-0.25, 0.25), sd = 1, Rsq = 0.5^2, ncov = 1, power = 0.8)
```

`pss.anova1way.F.unbal` *Power calculations for one-way unbalanced analysis of variance omnibus F test*

Description

Power calculations for one-way unbalanced analysis of variance omnibus F test

Usage

```
pss.anova1way.F.unbal(
  nvec = NULL,
  mvec = NULL,
  sd = NULL,
  Rsq = 0,
  ncov = 0,
  alpha = 0.05
)
```

Arguments

`nvec`	A vector of group sample sizes c(n1, n2, ...).
`mvec`	A vector of group mvec c(mu1, mu2, ...).
`sd`	The estimated standard deviation within each group.
`Rsq`	The estimated R^2 for regressing the outcome on the covariates; defaults to 0.
`ncov`	The number of covariates adjusted for in the model; defaults to 0.
`alpha`	The significance level or type 1 error rate; defaults to 0.05.

Value

A list of the arguments (including the computed power).

Examples

```
pss.anova1way.F.unbal(nvec = c(10, 20, 30), mvec = c(5, 10, 12), sd = 10)
```

`pss.anova2way.c.bal` *Power calculations for two-way balanced analysis of variance contrast test*

Description

Power calculations for two-way balanced analysis of variance contrast test

Usage

```
pss.anova2way.c.bal(
  n = NULL,
  mmatrix = NULL,
  cvec = NULL,
  factor = c("a", "b"),
  sd = 1,
  Rsq = 0,
  ncov = 0,
  alpha = 0.05,
  power = NULL
)
```

Arguments

n	The sample size per group.
mmatrix	A matrix of group means (see example).
cvec	A vector of contrast coefficients c(c1, c2, ...).
factor	Either "a" or "b" depending on which factor the contrast test is being made on.
sd	The estimated standard deviation within each group; defaults to 1.
Rsq	The estimated R^2 for regressing the outcome on the covariates; defaults to 0.
ncov	The number of covariates adjusted for in the model; defaults to 0.
alpha	The significance level or type 1 error rate; defaults to 0.05.
power	The specified level of power.

Value

A list of the arguments (including the computed one).

Examples

```
mmatrix <- matrix(c(9.3, 8.9, 8.5, 8.7, 8.3, 7.3), nrow = 2, byrow = TRUE)
pss.anova2way.c.bal(n = 30, mmatrix = mmatrix, cvec = c(1, 0, -1), factor = "b",
sd = 2, alpha = 0.05)
```

---

pss.anova2way.c.unbal *Power calculations for two-way unbalanced analysis of variance contrast test*

Description

Power calculations for two-way unbalanced analysis of variance contrast test

Usage

```
pss.anova2way.c.unbal(
  nmatrix = nmatrix,
  mmatrix = NULL,
  cvec = NULL,
  factor = c("a", "b"),
  sd = NULL,
  Rsq = 0,
  ncov = 0,
  alpha = 0.05
)
```

Arguments

`nmatrix`	A matrix of group sample sizes (see example).
`mmatrix`	A matrix of group means (see example).
`cvec`	A vector of contrast coefficients c(c1, c2, ...).
`factor`	Either "a" or "b" depending on which factor the contrast test is being made on.
`sd`	The estimated standard deviation within each group.
`Rsq`	The estimated R^2 for regressing the outcome on the covariates; defaults to 0.
`ncov`	The number of covariates adjusted for in the model; defaults to 0.
`alpha`	The significance level or type 1 error rate; defaults to 0.05.

Value

A list of the arguments (including the computed one).

Examples

```
nmatrix <- matrix(c(30, 30, 30, 30, 30, 30), nrow = 2, byrow = TRUE)
mmatrix <- matrix(c(9.3, 8.9, 8.5, 8.7, 8.3, 7.3), nrow = 2, byrow = TRUE)
pss.anova2way.c.unbal(nmatrix = nmatrix, mmatrix = mmatrix, cvec = c(1, 0, -1),
factor = "b", sd = 2, alpha = 0.05)
```

`pss.anova2way.F.bal` *Power calculations for two-way balanced analysis of variance omnibus F test*

Description

Power calculations for two-way balanced analysis of variance omnibus F test

Usage

```
pss.anova2way.F.bal(
  n = NULL,
  mmatrix = NULL,
  sd = 1,
  Rsq = 0,
```

```
  ncov = 0,
  alpha = 0.05,
  power = NULL
)
```

Arguments

n	The sample size per group.
mmatrix	A matrix of group means (see example).
sd	The estimated standard deviation within each group; defaults to 1.
Rsq	The estimated R^2 for regressing the outcome on the covariates; defaults to 0.
ncov	The number of covariates adjusted for in the model; defaults to 0.
alpha	The significance level or type 1 error rate; defaults to 0.05.
power	The specified level of power.

Value

A list of the arguments (including the computed one).

Examples

```
mmatrix <- matrix(c(9.3, 8.9, 8.5, 8.7, 8.3, 7.9), nrow = 2, byrow = TRUE)
pss.anova2way.F.bal(n = 30, mmatrix = mmatrix, sd = 2, alpha = 0.05)
mmatrix <- matrix(c(9.3, 8.9, 8.5, 8.7, 8.3, 7.3), nrow = 2, byrow = TRUE)
pss.anova2way.F.bal(n = 30, mmatrix = mmatrix, sd = 2, alpha = 0.05)
mmatrix <- matrix(c(9.3, 8.9, 8.5, 8.7, 8.3, 7.9), nrow = 2, byrow = TRUE)
pss.anova2way.F.bal(n = 30, mmatrix = mmatrix, sd = 2, Rsq = 0.4, ncov = 1, alpha = 0.05)
```

---

pss.anova2way.F.unbal *Power calculations for two-way unbalanced analysis of variance omnibus F test*

Description

Power calculations for two-way unbalanced analysis of variance omnibus F test

Usage

```
pss.anova2way.F.unbal(
  nmatrix = NULL,
  mmatrix = NULL,
  sd = NULL,
  Rsq = 0,
  ncov = 0,
  alpha = 0.05
)
```

Arguments

nmatrix	A matrix of group sample sizes (see example).
mmatrix	A matrix of group means (see example).
sd	The estimated standard deviation within each group.
Rsq	The estimated R^2 for regressing the outcome on the covariates; defaults to 0.
ncov	The number of covariates adjusted for in the model; defaults to 0.
alpha	The significance level or type 1 error rate; defaults to 0.05.

Value

A list of the arguments (including the computed power).

Examples

```
nmatrix <- matrix(c(30, 30, 30, 30, 30, 30), nrow = 2, byrow = TRUE)
mmatrix <- matrix(c(9.3, 8.9, 8.5, 8.7, 8.3, 7.9), nrow = 2, byrow = TRUE)
pss.anova2way.F.unbal(nmatrix = nmatrix, mmatrix = mmatrix, sd = 2, alpha = 0.05)
nmatrix <- matrix(c(30, 30, 30, 30, 30, 30), nrow = 2, byrow = TRUE)
mmatrix <- matrix(c(9.3, 8.9, 8.5, 8.7, 8.3, 7.3), nrow = 2, byrow = TRUE)
pss.anova2way.F.unbal(nmatrix = nmatrix, mmatrix = mmatrix, sd = 2, alpha = 0.05)
nmatrix <- matrix(c(30, 30, 30, 30, 30, 30), nrow = 2, byrow = TRUE)
mmatrix <- matrix(c(9.3, 8.9, 8.5, 8.7, 8.3, 7.9), nrow = 2, byrow = TRUE)
pss.anova2way.F.unbal(nmatrix = nmatrix, mmatrix = mmatrix, sd = 2, Rsq = 0.4^2,
ncov = 1, alpha = 0.05)
```

---

pss.anova2way.se.bal   *Power calculations for two-way balanced analysis of variance simple effects test*

Description

Power calculations for two-way balanced analysis of variance simple effects test

Usage

```
pss.anova2way.se.bal(
  n = NULL,
  mmatrix = NULL,
  cmatrix = NULL,
  sd = 1,
  Rsq = 0,
  ncov = 0,
  alpha = 0.05,
  power = NULL
)
```

Arguments

n	The sample size per group.
mmatrix	A matrix of group means (see example).
cmatrix	A matrix of contrast coefficients (see example).
sd	The estimated standard deviation within each group; defaults to 1.
Rsq	The estimated R^2 for regressing the outcome on the covariates; defaults to 0.
ncov	The number of covariates adjusted for in the model; defaults to 0.
alpha	The significance level or type 1 error rate; defaults to 0.05.
power	The specified level of power.

Value

A list of the arguments (including the computed one).

Examples

```
mmatrix <- matrix(c(9.3, 8.9, 8.5, 8.7, 8.3, 7.3), nrow = 2, byrow = TRUE)
cmatrix <- matrix(c(-1, 0, 0, 1, 0, 0), nrow = 2, byrow = TRUE)
pss.anova2way.se.bal(n = 30, mmatrix = mmatrix, cmatrix = cmatrix, sd = 2, alpha = 0.025)
```

---

pss.anova2way.se.unbal

Power calculations for two-way unbalanced analysis of variance simple effects test

---

Description

Power calculations for two-way unbalanced analysis of variance simple effects test

Usage

```
pss.anova2way.se.unbal(
  nmatrix = NULL,
  mmatrix = NULL,
  cmatrix = NULL,
  sd = 0,
  Rsq = 0,
  ncov = 0,
  alpha = 0.05
)
```

Arguments

nmatrix	A matrix of sample sizes (see example).
mmatrix	A matrix of group means (see example).
cmatrix	A matrix of contrast coefficients (see example).
sd	The estimated standard deviation within each group.
Rsq	The estimated R^2 for regressing the outcome on the covariates; defaults to 0.
ncov	The number of covariates adjusted for in the model; defaults to 0.
alpha	The significance level or type 1 error rate; defaults to 0.05.

Value

A list of the arguments (including the computed power).

Examples

```
nmatrix <- matrix(c(30, 30, 30, 30, 30, 30), nrow = 2, byrow = TRUE)
mmatrix <- matrix(c(9.3, 8.9, 8.5, 8.7, 8.3, 7.3), nrow = 2, byrow = TRUE)
cmatrix <- matrix(c(-1, 0, 0, 1, 0, 0), nrow = 2, byrow = TRUE)
pss.anova2way.se.unbal(nmatrix = nmatrix, mmatrix = mmatrix, cmatrix = cmatrix,
sd = 2, alpha = 0.025)
```

---

pss.chisq.gof *Power calculations for chi-square goodness-of-fit test*

Description

Power calculations for chi-square goodness-of-fit test

Usage

```
pss.chisq.gof(p0vec = NULL, p1vec = NULL, N = NULL, alpha = 0.05, power = NULL)
```

Arguments

p0vec	The first vector of probabilities (under the null).
p1vec	The second vector of probabilities (under the alternative hypothesis).
N	The number of total observations.
alpha	The significance level or type 1 error rate; defaults to 0.05.
power	The specified level of power.

Value

A list of the arguments (including the computed one).

Examples

```
pss.chisq.gof(p0vec = c(0.5, 0.3, 0.2), p1vec = c(0.7, 0.2, 0.1), N = 50)
```

pss.chisq.indep — *Power calculations for chi-square test of independence*

Description

Power calculations for chi-square test of independence

Usage

```
pss.chisq.indep(pmatrix = NULL, N = NULL, alpha = 0.05, power = NULL)
```

Arguments

pmatrix	The two-way probability table under the alternative hypothesis.
N	The number of total observations.
alpha	The significance level or type 1 error rate; defaults to 0.05.
power	The specified level of power.

Value

A list of the arguments (including the computed one).

Examples

```
pss.chisq.indep(pmatrix = matrix(c(0.050, 0.350, 0.100, 0.075, 0.250, 0.175),
nrow = 2, byrow = TRUE), N = 230)
pss.chisq.indep(pmatrix = matrix(c(0.3, 0.2, 0.4, 0.1), nrow = 2, byrow = TRUE), N = 200)
```

pss.ci.mean — *Power calculations for precision analysis for one mean*

Description

Power calculations for precision analysis for one mean

Usage

```
pss.ci.mean(
  N = NULL,
  halfwidth = NULL,
  sd = 1,
  alpha = 0.05,
  power = NULL,
  cond = FALSE
)
```

Arguments

`N`	The sample size.
`halfwidth`	The desired halfwidth.
`sd`	The estimated standard deviation; defaults to 1.
`alpha`	The significance level or type 1 error rate; defaults to 0.05.
`power`	The specified level of power.
`cond`	Specify using unconditional or conditional probability. Defaults to FALSE.

Value

A list of the arguments (including the computed one).

Examples

```
pss.ci.mean(N = NULL, halfwidth = 0.25, power = 0.8)
pss.ci.mean(N = 62, halfwidth = 0.25, power = NULL)
pss.ci.mean(N = 73, halfwidth = 0.25, cond = TRUE)
```

`pss.ci.meandiff` *Power calculations for precision analysis for a difference between means*

Description

Power calculations for precision analysis for a difference between means

Usage

```
pss.ci.meandiff(
  n1 = NULL,
  n.ratio = 1,
  halfwidth = NULL,
  sd = 1,
  alpha = 0.05,
  power = NULL,
  cond = FALSE
)
```

Arguments

`n1`	The sample size for group 1.
`n.ratio`	The ratio n2/n1 between the sample sizes of two groups; defaults to 1 (equal group sizes).
`halfwidth`	The desired halfwidth for the difference in means.
`sd`	The estimated standard deviation; defaults to 1.
`alpha`	The significance level or type 1 error rate; defaults to 0.05.
`power`	The specified level of power.
`cond`	Specify using unconditional or conditional probability. Defaults to FALSE.

Value

A list of the arguments (including the computed one).

Examples

```
pss.ci.meandiff(n1 = NULL, halfwidth = 0.25, power = 0.8)
pss.ci.meandiff(n1 = 134, halfwidth = 0.25, cond = TRUE)
```

---

pss.coprimary.t *Power calculations for multiple co-primary continuous endpoints assuming unknown covariance matrix*

Description

Power calculations for multiple co-primary continuous endpoints assuming unknown covariance matrix

Usage

```
pss.coprimary.t(
  K,
  n1 = NULL,
  n.ratio = 1,
  delta = NULL,
  Sigma,
  sd,
  rho,
  alpha = 0.05,
  power = NULL,
  M = 10000,
  min.n = NULL,
  max.n = NULL,
  tol = .Machine$double.eps^0.25,
  use.uniroot = TRUE
)
```

Arguments

`K`	The number of endpoints.
`n1`	The sample size for group 1.
`n.ratio`	The ratio n2/n1 between the sample sizes of two groups; defaults to 1 (equal group sizes).
`delta`	A vector of length K of the true mean differences mu1k - mu2k; must all be positive.
`Sigma`	The covariance matrix of the K outcomes, of dimension K x K.
`sd`	A vector of length K of the standard deviations of the K outcomes.
`rho`	A vector of length 0.5*K*(K-1) of the correlations among the K outcomes.
`alpha`	The significance level or type 1 error rate; defaults to 0.05. A one-sided test is assumed.

power	The specified level of power.
M	The number of simulation.
min.n	Minimum value of n1; used in search for n1 to achieve desired power.
max.n	Maximum value of n1; used in search for n1 to achieve desired power.

Value

A list of the arguments (including the computed one).

Examples

```
pss.coprimary.t(K = 2, n1 = 100, delta = c(0.4, 0.5), sd = c(1, 1), rho = 0.3,
alpha = 0.025, power = NULL)
```

`pss.coprimary.z` *Power calculations for multiple co-primary continuous endpoints assuming known covariance matrix*

Description

Power calculations for multiple co-primary continuous endpoints assuming known covariance matrix

Usage

```
pss.coprimary.z(
  K,
  n1 = NULL,
  n.ratio = 1,
  delta = NULL,
  Sigma,
  sd,
  rho,
  alpha = 0.05,
  power = NULL,
  tol = .Machine$double.eps^0.25
)
```

Arguments

K	The number of endpoints.
n1	The sample size for group 1.
n.ratio	The ratio n2/n1 between the sample sizes of two groups; defaults to 1 (equal group sizes).
delta	A vector of length K of the true mean differences mu1k - mu2k; must all be positive.
Sigma	The covariance matrix of the K outcomes, of dimension K x K.
sd	A vector of length K of the standard deviations of the K outcomes.
rho	A vector of length 0.5*K*(K-1) of the correlations among the K outcomes.

alpha	The significance level or type 1 error rate; defaults to 0.05. A one-sided test is assumed.
power	The specified level of power.

Value

A list of the arguments (including the computed one).

Examples

```
pss.coprimary.z(K = 2, n = 100, delta = c(0.4, 0.5), sd = c(1, 1), rho = 0.3,
alpha = 0.025, power = NULL)
```

---

pss.corr.1samp *Power calculations for one correlation coefficient*

Description

Power calculations for one correlation coefficient

Usage

```
pss.corr.1samp(
  N = NULL,
  rho0 = 0,
  rhoA = NULL,
  alpha = 0.05,
  power = NULL,
  sides = 2
)
```

Arguments

N	The sample size.
rho0	The correlation coefficient under the null hypothesis; defaults to 0.
rhoA	The correlation coefficient under the alternative hypothesis.
alpha	The significance level or type 1 error rate; defaults to 0.05.
power	The specified level of power.
sides	Either 1 or 2 (default) to specify a one- or two- sided hypothesis test.

Value

A list of the arguments (including the computed one).

Examples

```
pss.corr.1samp(N = 100, rhoA = 0.2, sides = 1)
pss.corr.1samp(N = 100, rho0 = 0.2, rhoA = 0.4, sides = 1)
```

pss.corr.2samp — *Power calculations for comparing two correlation coefficients*

Description

Power calculations for comparing two correlation coefficients

Usage

```
pss.corr.2samp(
  n1 = NULL,
  n.ratio = 1,
  rho1 = NULL,
  rho2 = NULL,
  alpha = 0.05,
  power = NULL,
  sides = 2
)
```

Arguments

`n1`	The sample size for group 1.
`n.ratio`	The ratio n2/n1 between the sample sizes of two groups; defaults to 1 (equal group sizes).
`rho1`	The correlation coefficient in the first group.
`rho2`	The correlation coefficient in the second group.
`alpha`	The significance level or type 1 error rate; defaults to 0.05.
`power`	The specified level of power.
`sides`	Either 1 or 2 (default) to specify a one- or two- sided hypothesis test.

Value

A list of the arguments (including the computed one).

Examples

```
pss.corr.2samp(n1 = 300, rho1 = 0.3, rho2 = 0.1, sides = 1)
```

pss.mcnemar.test — *Power approximation for McNemar's test for two correlated proportions*

Description

Power approximation for McNemar's test for two correlated proportions

Usage

```
pss.mcnemar.test(
  N = NULL,
  p1 = NULL,
  p2 = NULL,
  rho = NULL,
  paid = NULL,
  psi = NULL,
  alpha = 0.05,
  power = NULL,
  sides = 2
)
```

Arguments

N	The sample size; the number of pairs.
p1	The proportion in condition 1.
p2	The proportion in condition 2.
rho	The estimated correlation between the two conditions.
paid	The smaller of the two discordant probabilities. Either p1, p2, and rho, OR paid and psi must be specified.
psi	The discordant proportion ratio. Either p1, p2, and rho, OR paid and psi must be specified.
alpha	The significance level or type 1 error rate; defaults to 0.05.
power	The specified level of power.
sides	Either 1 or 2 (default) to specify a one- or two- sided hypothesis test.

Value

A list of the arguments (including the computed one).

Examples

```
pss.mcnemar.test(N = NULL, p1 = 0.8, p2 = 0.9, rho = 0, power = 0.9, sides = 2)
pss.mcnemar.test(N = NULL, paid = 0.08, psi = 0.18 / 0.08, power = 0.9, sides = 2)
```

---

pss.mlrF.overall — *Power calculations for a multiple linear regression overall F test*

Description

Power calculations for a multiple linear regression overall F test

Usage

```
pss.mlrF.overall(
  N = NULL,
  p = NULL,
  Rsq = NULL,
  fsq = NULL,
  alpha = 0.05,
  power = NULL,
  random = FALSE
)
```

Arguments

N	The sample size.
p	The number of predictors.
Rsq	The squared sample multiple correlation coefficient.
fsq	The squared f effect size. Either Rsq OR fsq must be specified.
alpha	The significance level or type 1 error rate; defaults to 0.05.
power	The specified level of power.
random	Whether the values of the predictors are random; defaults to FALSE.

Value

A list of the arguments (including the computed one).

Examples

```
pss.mlrF.overall(N = 400, p = 2, Rsq = 0.02)
pss.mlrF.overall(N = 400, p = 2, fsq = 0.02 / (1 - 0.02))
pss.mlrF.overall(N = 109, p = 1, Rsq = 0.3^2)
pss.mlrF.overall(N = 50, p = 1, Rsq = 0.2)
pss.mlrF.overall(N = 50, p = 3, Rsq = 0.2)
pss.mlrF.overall(N = 50, p = 5, Rsq = 0.2)
pss.mlrF.overall(N = 400, p = 2, Rsq = 0.02, random = TRUE)
```

---

pss.mlrF.partial — *Power calculations for a multiple linear regression partial F test*

Description

Power calculations for a multiple linear regression partial F test

Usage

```
pss.mlrF.partial(
  N = NULL,
  p = NULL,
  q = NULL,
  pc = NULL,
  Rsq.red = NULL,
```

```
  Rsq.full = NULL,
  alpha = 0.05,
  power = NULL
)
```

Arguments

N	The sample size.
p	The number of control predictors.
q	The number of test predictors.
pc	The partial correlation coefficient. Either both Rsq terms OR pc must be specified.
Rsq.red	The squared sample multiple correlation coefficient in the reduced model.
Rsq.full	The squared sample multiple correlation coefficient in the full model.
alpha	The significance level or type 1 error rate; defaults to 0.05.
power	The specified level of power.

Value

A list of the arguments (including the computed one).

Examples

```
pss.mlrF.partial(N = 80, p = 3, q = 2, Rsq.red = 0.25, Rsq.full = 0.35)
pss.mlrF.partial(N = 150, p = 4, pc = 0.2)
```

---

pss.multisite.ate *Power for test of average treatment effect*

Description

Power for test of average treatment effect

Usage

```
pss.multisite.ate(
  m = NULL,
  m.sd = 0,
  alloc.ratio = 1,
  J = NULL,
  delta = NULL,
  sd = 1,
  rho0 = NULL,
  rho1 = NULL,
  Rsq = 0,
  alpha = 0.05,
  power = NULL,
  sides = 2
)
```

Arguments

`m`	The number of subjects per site or the mean cluster size (if unequal number of participants per site).
`m.sd`	The standard deviation of cluster sizes (provide if unequal number of participants per site); defaults to 0.
`alloc.ratio`	The allocation ratio of intervention/control per site; defaults to 1.
`J`	The number of sites.
`delta`	The difference between the intervention and control means in the outcome variable.
`sd`	The total standard deviation of the outcome variable; defaults to 1.
`rho0`	The proportion of total variance of the outcome attributable to variation in site-level means.
`rho1`	The proportion of total variance of the outcome attributable to variation in the treatment effect across sites.
`Rsq`	The estimated R^2 for regressing the outcome on the covariates; defaults to 0.
`alpha`	The significance level or type 1 error rate; defaults to 0.05.
`power`	The specified level of power.
`sides`	Either 1 or 2 (default) to specify a one- or two- sided hypothesis test.

Value

A list of the arguments (including the computed one).

Examples

```
pss.multisite.ate(m = 20, J = 10, delta = 3, sd = sqrt(40), rho0 = 0.1, rho1 = 0)
pss.multisite.ate(m = 20, J = 10, delta = 3, sd = sqrt(48), rho0 = 0.095, rho1 = 0.048)
pss.multisite.ate(m = 20, alloc.ratio = 1.5, J = 10, delta = 0.43, rho0 = 0.095, rho1 = 0.048)
pss.multisite.ate(m = 10, J = NULL, delta = 0.5, sd = 1, rho0 = 0, rho1 = 0.05, power = 0.8)
pss.multisite.ate(m = 20, m.sd = 5, J = 10, delta = 3, sd = sqrt(48), rho0 = 0.095, rho1 = 0.048)
pss.multisite.ate(m = 20, J = 10, delta = 3, sd = sqrt(48), rho0 = 0.095, rho1 = 0.048, Rsq = 0.5^2)
```

`pss.multisite.bin` *Number of sites for multisite trials with binary outcomes*

Description

Number of sites for multisite trials with binary outcomes

Usage

```
pss.multisite.bin(
  m = NULL,
  alloc.ratio = 1,
  J = NULL,
  pc = NULL,
  pt = NULL,
  sigma.u = NULL,
```

```
  alpha = 0.05,
  power = NULL,
  sides = 2
)
```

Arguments

m	The number of subjects per site.
alloc.ratio	The allocation ratio of intervention/control per site; defaults to 1.
J	The number of sites.
pc	The probability of the outcome in the control condition.
pt	The probability of the outcome in the treatment condition.
sigma.u	Standard deviation of the treatment effect across sites.
alpha	The significance level or type 1 error rate; defaults to 0.05.
power	The specified level of power.
sides	Either 1 or 2 (default) to specify a one- or two- sided hypothesis test.

Value

A list of the arguments (including the computed one).

Examples

```
pss.multisite.bin(m = 30, J = 25, pc = 0.1, pt = 0.2, sigma.u = 0.4, power = NULL)
pss.multisite.bin(m = 30, J = NULL, pc = 0.1, pt = 0.2, sigma.u = 0.4, power = 0.9)
```

---

pss.multisite.hte *Power for test of heterogeneity of treatment effect*

Description

Power for test of heterogeneity of treatment effect

Usage

```
pss.multisite.hte(
  m = NULL,
  J = NULL,
  ssq.u1 = NULL,
  ssq.e = NULL,
  alpha = 0.05
)
```

Arguments

m	The number of subjects per site.
J	The number of sites.
ssq.u1	The variance of the site-level treatment effects (sigma squared u1) under the alternative.
ssq.e	The variance of the observations within sites (sigma squared epsilon) under the alternative.
alpha	The significance level or type 1 error rate; defaults to 0.05.

Value

A list of the arguments (including the computed power).

Examples

```
pss.multisite.hte(m = 10, J = 30, ssq.u1 = 8, ssq.e = 36)
```

---

pss.multisite.re *Relative efficiency due to unequal number of participants per site*

Description

Relative efficiency due to unequal number of participants per site

Usage

```
pss.multisite.re(m.mean = NULL, m.sd = NULL, rho = NULL)
```

Arguments

m.mean	The mean cluster size.
m.sd	The standard deviation of cluster sizes.
rho	The intraclass correlation among cluster sizes.

Value

A list of the arguments (including the computed relative efficiency).

Examples

```
1 / pss.multisite.re(m.mean = 30, m.sd = 23, rho = 0.05)$re
1 / pss.multisite.re(m.mean = 30, m.sd = 23, rho = 0.1)$re
```

---

pss.prop.1samp *Power calculations for one sample proportion tests*

Description

Power calculations for one sample proportion tests

Usage

```
pss.prop.1samp(
  N = NULL,
  p0 = NULL,
  pA = NULL,
  alpha = 0.05,
  power = NULL,
  sides = 2
)
```

Arguments

N	The sample size.
p0	The proportion under the null.
pA	The true proportion.
alpha	The significance level or type 1 error rate; defaults to 0.05.
power	The specified level of power.
sides	Either 1 or 2 (default) to specify a one- or two- sided hypothesis test.

Value

A list of the arguments (including the computed one).

Examples

```
pss.prop.1samp(N = NULL, p0 = 0.2, pA = 0.3, power = 0.8, sides = 1)
pss.prop.1samp(N = NULL, p0 = 0.4, pA = 0.5, power = 0.8, sides = 1)
```

---

pss.prop.2samp *Power calculations for two sample proportion tests*

Description

Power calculations for two sample proportion tests

Usage

```
pss.prop.2samp(
  n1 = NULL,
  n.ratio = 1,
  p1 = NULL,
  p2 = NULL,
  delta = 0,
  alpha = 0.05,
  power = NULL,
  sides = 2
)
```

Arguments

n1	The sample size for group 1.
n.ratio	The ratio n2/n1 between the sample sizes of two groups; defaults to 1 (equal group sizes).
p1	The proportion in group 1.
p2	The proportion in group 2.
delta	The margin of noninferiority or superiority; defaults to 0. See delta.sign for guidance on the sign of delta.
alpha	The significance level or type 1 error rate; defaults to 0.05.
power	The specified level of power.
sides	Either 1 or 2 (default) to specify a one- or two- sided hypothesis test.

Value

A list of the arguments (including the computed one).

Examples

```
pss.prop.2samp(n1 = NULL, p1 = 0.6, p2 = 0.8, alpha = 0.025, power = 0.9, sides = 1)
pss.prop.2samp(n1 = NULL, p1 = 0.25, p2 = 0.25, delta = 0.1, alpha = 0.025, power = 0.8, sides = 1)
```

---

pss.prop.test.equiv — *Power calculations for test of equivalence of two proportions*

Description

Power calculations for test of equivalence of two proportions

Usage

```
pss.prop.test.equiv(
  n1 = NULL,
  n.ratio = 1,
  p1 = NULL,
  p2 = NULL,
  delta = NULL,
  alpha = 0.05,
  power = NULL,
  sides = 2
)
```

Arguments

n1	The sample size for group 1.
n.ratio	The ratio n2/n1 between the sample sizes of two groups; defaults to 1 (equal group sizes).
p1	The outcome proportion in group 1.
p2	The outcome proportion in group 2.
delta	The equivalence margin.
alpha	The significance level or type 1 error rate; defaults to 0.05.
power	The specified level of power.
sides	Either 1 or 2 (default) to specify a one- or two- sided hypothesis test.

Value

A list of the arguments (including the computed one).

Examples

```
pss.prop.test.equiv(n1 = NULL, p1 = 0.5, p2 = 0.5, delta = 0.1, alpha = 0.05, power = 0.8,
sides = 1)
```

pss.rr *Power calculations for relative risk*

Description

Power calculations for relative risk

Usage

```
pss.rr(
  n1 = NULL,
  n.ratio = 1,
  p1 = NULL,
  p2 = NULL,
  RR0 = 1,
  alpha = 0.05,
  power = NULL,
  sides = 2
)
```

Arguments

`n1`	The sample size for group 1.
`n.ratio`	The ratio n2/n1 between the sample sizes of two groups; defaults to 1 (equal group sizes).
`p1`	The proportion in group 1.
`p2`	The proportion in group 2.
`RR0`	The relative risk under the null; defaults to 1.
`alpha`	The significance level or type 1 error rate; defaults to 0.05.
`power`	The specified level of power.
`sides`	Either 1 or 2 (default) to specify a one- or two- sided hypothesis test.

Value

A list of the arguments (including the computed one).

Examples

```
pss.rr(n1 = NULL, n.ratio = 6, p1 = 0.1, p2 = 0.1 * 2, power = 0.8)
```

pss.slr *Power calculations for a simple linear regression*

Description

Power calculations for a simple linear regression

Usage

```
pss.slr(
  N = NULL,
  beta10 = 0,
  beta1A = NULL,
  sd.x.sq = NULL,
  sigma.e = NULL,
  alpha = 0.05,
  power = NULL,
  sides = 2
)
```

Arguments

N	The sample size.
beta10	The slope regression coefficient under the null hypothesis.
beta1A	The slope regression coefficient under the alternative hypothesis.
sd.x.sq	The sample variance of the covariate X.
sigma.e	The estimated standard deviation of the error terms.
alpha	The significance level or type 1 error rate; defaults to 0.05.
power	The specified level of power.
sides	Either 1 or 2 (default) to specify a one- or two- sided hypothesis test.

Value

A list of the arguments (including the computed one).

Examples

```
# Yi = beta0 + beta1 * Xi + ei, i = 1,...,N
pss.slr(N = 100, beta10 = 1, beta1A = 1.5, sd.x.sq = 25, sigma.e = 10, sides = 1)
```

pss.t.test.1samp *Power calculations for one sample t tests*

Description

Power calculations for one sample t tests

Usage

```
pss.t.test.1samp(
  N = NULL,
  delta = NULL,
  sd = 1,
  alpha = 0.05,
  power = NULL,
  sides = 2,
  strict = TRUE
)
```

Arguments

N	The sample size.
delta	muA (the true mean) - mu0 (the mean under the null).
sd	The estimated standard deviation; defaults to 1.
alpha	The significance level or type 1 error rate; defaults to 0.05.
power	The specified level of power.
sides	Either 1 or 2 (default) to specify a one- or two- sided hypothesis test.
strict	Use strict interpretation in two-sided case; defaults to TRUE.

Value

A list of the arguments (including the computed one).

Examples

```
pss.t.test.1samp(N = 36, delta = 4.9 - 5.7, sd = 2, sides = 1)
pss.t.test.1samp(N = 36, delta = 6.3 - 5.7, sd = 2, sides = 1)
pss.t.test.1samp(N = 36, delta = 4.9 - 5.7, sd = 2, sides = 2)
pss.t.test.1samp(delta = 0.6, sd = 1, power = 0.8, sides = 1)
```

pss.t.test.2samp *Power calculations for two sample t tests allowing for unequal sample sizes and/or variances*

Description

Power calculations for two sample t tests allowing for unequal sample sizes and/or variances

Usage

```
pss.t.test.2samp(
  n1 = NULL,
  n.ratio = 1,
  delta = NULL,
  sd = 1,
  sd.ratio = 1,
  df.method = c("welch", "classical"),
  alpha = 0.05,
  power = NULL,
  sides = 2,
  strict = TRUE
)
```

Arguments

n1	The sample size for group 1.
n.ratio	The ratio n2/n1 between the sample sizes of two groups; defaults to 1 (equal group sizes).
delta	DeltaA (the true difference mu1 - mu2) - Delta0 (the difference under the null) - delta. See delta.sign for guidance on the sign of delta.
sd	The estimated standard deviation for group 1; defaults to 1 (equal standard deviations in the two groups).
sd.ratio	The ratio sd2/sd1 between the standard deviations of the two groups.
df.method	Method for calculating the degrees of freedom: "welch" (default) or "classical".
alpha	The significance level or type 1 error rate; defaults to 0.05.
power	The specified level of power.
sides	Either 1 or 2 (default) to specify a one- or two- sided hypothesis test.
strict	Use strict interpretation in two-sided case; defaults to TRUE.

Value

A list of the arguments (including the computed one).

Examples

```
pss.t.test.2samp(n1 = 50, delta = 2, sd = 5, sides = 1)
pss.t.test.2samp(n1 = NULL, n.ratio = 2, delta = 0.5, sd = 1, power = 0.8, sides = 2)
pss.t.test.2samp(n1 = 49, n.ratio = 2, delta = 0.5, sd = 1, power = NULL, sides = 2)
pss.t.test.2samp(n1 = 25, n.ratio = 3, delta = 3, sd = 4, sd.ratio = 1.5, alpha = 0.025, sides = 1)
pss.t.test.2samp(n1 = NULL, delta = 0.5, sd = 1, power = 0.8, sides = 2)
```

`pss.t.test.paired` *Power calculations for paired t tests*

Description

Power calculations for paired t tests

Usage

```
pss.t.test.paired(
  N = NULL,
  delta = NULL,
  sd1 = 1,
  sd2 = 1,
  rho = NULL,
  alpha = 0.05,
  power = NULL,
  sides = 2,
  strict = TRUE
)
```

Arguments

`N`	The sample size; the number of pairs.
`delta`	DeltaA (the true mean difference) - Delta0 (the mean difference under the null).
`sd1`	The estimated pre standard deviation; defaults to 1.
`sd2`	The estimated post standard deviation; defaults to 1.
`rho`	The estimated correlation between pre and post measurements on the same individual.
`alpha`	The significance level or type 1 error rate; defaults to 0.05.
`power`	The specified level of power.
`sides`	Either 1 or 2 (default) to specify a one- or two- sided hypothesis test.
`strict`	Use strict interpretation in two-sided case; defaults to TRUE.

Value

A list of the arguments (including the computed one).

Examples

```
pss.t.test.paired(N = NULL, delta = 4, sd1 = 10, sd2 = 10, rho = 0.4, power = 0.8, sides = 2)
```

pss.z.test.1samp — *Power calculations for one sample z tests*

Description

Power calculations for one sample z tests

Usage

```
pss.z.test.1samp(
  N = NULL,
  delta = NULL,
  sd = 1,
  alpha = 0.05,
  power = NULL,
  sides = 2,
  strict = TRUE
)
```

Arguments

N	The sample size.
delta	muA (the true mean) - mu0 (the mean under the null).
sd	The estimated standard deviation; defaults to 1.
alpha	The significance level or type 1 error rate; defaults to 0.05.
power	The specified level of power.
sides	Either 1 or 2 (default) to specify a one- or two- sided hypothesis test.
strict	Use strict interpretation in two-sided case; defaults to TRUE.

Value

A list of the arguments (including the computed one).

Examples

```
pss.z.test.1samp(N = NULL, delta = 6.5 - 5.7, sd = 2, power = 0.8, sides = 2)
pss.z.test.1samp(N = 40, delta = NULL, sd = 1, power = 0.9, sides = 2)
pss.z.test.1samp(N = NULL, delta = 0.6, sd = 1, power = 0.8, sides = 1)
```

`pss.z.test.2samp` *Power calculations for two sample z tests allowing for unequal sample sizes and/or variances*

Description

Power calculations for two sample z tests allowing for unequal sample sizes and/or variances

Usage

```
pss.z.test.2samp(
  n1 = NULL,
  n.ratio = 1,
  delta = NULL,
  sd = 1,
  sd.ratio = 1,
  alpha = 0.05,
  power = NULL,
  sides = 2,
  strict = TRUE
)
```

Arguments

`n1`	The sample size for group 1.
`n.ratio`	The ratio n2/n1 between the sample sizes of two groups; defaults to 1 (equal group sizes).
`delta`	DeltaA (the true difference mu1 - mu2) - Delta0 (the difference under the null) - delta. See delta.sign for guidance on the sign of delta.
`sd`	The estimated standard deviation for group 1; defaults to 1 (equal standard deviations in the two groups).
`sd.ratio`	The ratio sd2/sd1 between the standard deviations of the two groups.
`alpha`	The significance level or type 1 error rate; defaults to 0.05.
`power`	The specified level of power.
`sides`	Either 1 or 2 (default) to specify a one- or two- sided hypothesis test.
`strict`	Use strict interpretation in two-sided case; defaults to TRUE.

Value

A list of the arguments (including the computed one).

Examples

```
pss.z.test.2samp(n1 = NULL, n.ratio = 1, delta = 0.5, sd = 1, power = 0.8, sides = 2)
```

`pss.z.test.paired` *Power calculations for paired z tests*

Description

Power calculations for paired z tests

Usage

```
pss.z.test.paired(
  N = NULL,
  delta = NULL,
  sd1 = 1,
  sd2 = 1,
  rho = NULL,
  alpha = 0.05,
  power = NULL,
  sides = 2,
  strict = TRUE
)
```

Arguments

`N`	The sample size; the number of pairs.
`delta`	DeltaA (the true mean difference) - Delta0 (the mean difference under the null).
`sd1`	The estimated pre standard deviation; defaults to 1.
`sd2`	The estimated post standard deviation; defaults to 1.
`rho`	The estimated correlation between pre and post measurements on the same individual.
`alpha`	The significance level or type 1 error rate; defaults to 0.05.
`power`	The specified level of power.
`sides`	Either 1 or 2 (default) to specify a one- or two- sided hypothesis test.
`strict`	Use strict interpretation in two-sided case; defaults to TRUE.

Value

A list of the arguments (including the computed one).

Examples

```
pss.z.test.paired(N = NULL, delta = 4, sd1 = 10, sd2 = 10, rho = 0.4, power = 0.8, sides = 2)
```

`swd_1trt_power` *Stepped Wedge Design Power Calculation with 1 Treatment*

Description

Calculates power for stepped wedge design with one treatment and one control condition.

Usage

```
swd_1trt_power(alpha = 0.05, J, m, K, b, d, rho, rho_c, rho_s)
```

Arguments

alpha	The significance level or type 1 error rate
J	Number of clusters
m	The number of individuals in each cluster at each time period
K	The number of steps (if one baseline period, then this is periods - 1)
b	The number of baseline periods
d	The standardized effect size
rho	Intraclass correlation coefficient (ICC)
rho_c	Cluster autocorrelation (proportion of cluster-level variance that is time invariant)
rho_s	Subject autocorrelation (This is 0 in repeated cross sectional designs because people are measured only once)

Value

power

Examples

```
# closed cohort design as in Example 12.5
swd_1trt_power(alpha = 0.05, m = 30, J = 5, K = 5, b = 1, d = 0.4,
               rho = 0.05, rho_c = 0.2, rho_s = 0.4)
# repeated cross sectional design as in Example 12.5
swd_1trt_power(alpha = 0.05, m = 30, J = 5, K = 5, b = 1, d = 0.4,
               rho = 0.05, rho_c = 0.2, rho_s = 0)
```

swd_1trt_ss *Stepped Wedge Design Sample Size Calculation with 1 Treatment*

Description

Calculates sample size requirements for stepped wedge design with one treatment and one control condition.

Usage

```
swd_1trt_ss(alpha, power, m, K, b, d, rho, rho_c, rho_s)
```

Arguments

alpha	The significance level or type 1 error rate
power	The specified level of power
m	The number of individuals in each cluster at each time period
K	The number of steps (if one baseline period, then this is periods - 1)
b	The number of baseline periods
d	The standardized effect size
rho	Intraclass correlation coefficient (ICC)
rho_c	Cluster autocorrelation (proportion of cluster-level variance that is time invariant)
rho_s	Subject autocorrelation (This is 0 in repeated cross sectional designs because people are measured only once)

Value

Calculated number of clusters required, adjusted number of clusters, and suggested number of clusters that includes the value of K

Examples

```
# Repeated Cross Sectional Stepped Wedge Sample Size Calculation
# with 3 steps, 1 baseline period, 80% power, 0.05 significance, 30 ppl per
# cluster, effect size of 0.4, ICC of 0.05, and cluster autocorrelation of 0.2
# Following Example 12.6 in text
swd_1trt_ss(alpha = 0.05, power = 0.8, m = 30, K = 3, b = 1, d = 0.4,
            rho = 0.05, rho_c = 0.2, rho_s = 0)
```

swd_2trt_additive_power

Additive Treatments Stepped Wedge Design Power Calculation

Description

Calculate power for detecting treatment effects in a two-treatment stepped wedge design with additive treatment effects. We only consider complete designs. This function is based on Power Analysis for Stepped Wedge Trials with Two Treatments published in Stat Med in 2022 by Phillip Sundin and Catherine Crespi. This manuscript uses the normal distribution and the Wald Test to derive power.

Usage

```
swd_2trt_additive_power(
  RhoW,
  ModelChoice,
  n.individuals,
  n.clusters,
  n.periods,
  delta_1,
  delta_2,
  RhoA,
  IAC,
  Sequence_Tx1,
  Sequence_Tx2,
  typeIerror
)
```

Arguments

RhoW	The intraclass correlation coefficient, which refers to the correlation between outcomes of two different individuals in the same cluster at the same time.
ModelChoice	A string indicating either a repeated cross sectional design ("RCS"), nested exchangeable ("NEC"), Cohort ("Cohort"), or any one of these three designs in their standardized form. Only one of "RCS Standardized", "RCS", "Cohort Standardized", "Cohort", "NEC Standardized", or "NEC" can be specified. The standardized design refers to dividing by the standard deviation of y (the outcome) in the covariance specification.
n.individuals	The number of individuals in each condition at one time period.
n.clusters	The number of clusters
n.periods	The number of time periods.
delta_1	The standardized effect size for treatment 1
delta_2	The standardized effect size for treatment 2
RhoA	The correlation between two observations in the same individual in the same cluster but different time periods. RCS models assume RhoW = RhoA, and Cohort models assume RhoA >= RhoW.
IAC	The individual auto-correlation, which is the proportion of the individual-level variance that is time-invariant. IAC = 0 in a RCS.

Sequence_Tx1	A vector representing the time points (periods) at which each cluster transitions from control to treatment 1 (ie the first occurence).
Sequence_Tx2	A vector representing the time points (periods) at which each cluster transitions from control to treatment 2 (ie the first occurence).
typeIerror	The significance level.

Value

A list with power for detecting treatment effect for treatment 1 and 2

Examples

```
## For example, the 3 cluster, 4 period SWD,
##     0 1 1   1
##     0 0 1+2 1+2
##     0 0 2   1+2
## where 0 denotes control condition, 1 denotes the condition with only
## treatment 1, 2 denote the condition with only treatment 2, and 1+2 denotes the condition
## where a cluster receives both treatment 1 and 2
## would have Sequence1 <- c(2,3,4) and Sequence2 <- c(NA,3,3)
## notice sequence 1 cluster 3 has time period 4 even though time period 3 had treatment 2,
## so it is the first occurrence of trt 1 in this cluster.


# 12-cluster concurrent repeated cross sectional design
# Sequencing has 2 clusters transition to treatment 1 at time 2,
# 2 clusters transition to treatment 1 at time 3, and 2 clusters transition
# to treatment 1 at time 3. These clusters never receive treatment 2
Sequence_Tx1 <- c(2, 2, 3, 3, 4, 4, NA, NA, NA, NA, NA, NA)
# similar sequencing for treatment 2
Sequence_Tx2 <- c(NA, NA, NA, NA, NA, NA, 4, 4, 3, 3, 2, 2)
swd_2trt_additive_power(RhoW = 0.05, ModelChoice = "RCS", RhoA = 0.2, IAC = 0,
           n.individuals = 30, n.clusters = 12, n.periods = 4, delta_1 = 0.4, delta_2 = 0.1,
               Sequence_Tx1 = Sequence_Tx1, Sequence_Tx2 = Sequence_Tx2,
               typeIerror = 0.05)
```

---

swd_2trt_interaction_power

Stepped Wedge Design Power Calculation for Interaction Term

Description

Calculate power for detecting interaction effects in a two-treatment stepped wedge design. We only consider complete designs. This function is based on Power Analysis for Stepped Wedge Trials with Two Treatments published in Stat Med in 2022 by Phillip Sundin and Catherine Crespi. This manuscript uses the normal distribution and the Wald Test to derive power.

Usage

```
swd_2trt_interaction_power(
  RhoW,
  ModelChoice,
  n.individuals,
```

```
  n.clusters,
  n.periods,
  delta_1,
  delta_2,
  delta_3,
  RhoA,
  IAC,
  typeIerror,
  Sequence_Tx1,
  Sequence_Tx2
)
```

Arguments

RhoW	The intraclass correlation coefficient, which refers to the correlation between outcomes of two different individuals in the same cluster at the same time.
ModelChoice	A string indicating either a repeated cross sectional design ("RCS"), nested exchangeable ("NEC"), Cohort ("Cohort"), or any one of these three designs in their standardized form. Only one of "RCS Standardized", "RCS", "Cohort Standardized", "Cohort", "NEC Standardized", or "NEC" can be specified. The standardized design refers to dividing by the standard deviation of y (the outcome) in the covariance specification.
n.individuals	The number of individuals in each condition at one time period.
n.clusters	The number of clusters
n.periods	The number of time periods.
delta_1	The standardized effect size for treatment 1
delta_2	The standardized effect size for treatment 2
delta_3	The standardized effect size for the interaction effect.
RhoA	The correlation between two observations in the same individual in the same cluster but different time periods. RCS models assume RhoW = RhoA, and Cohort models assume RhoA >= RhoW.
IAC	The individual auto-correlation, which is the proportion of the individual-level variance that is time-invariant. IAC = 0 in a RCS.
typeIerror	The significance level.
Sequence_Tx1	A vector representing the time points (periods) at which each cluster transitions from control to treatment 1 (ie the first occurence).
Sequence_Tx2	A vector representing the time points (periods) at which each cluster transitions from control to treatment 2 (ie the first occurence).

Value

A matrix with the standard error, statistic, and power for detecting effects for treatment 1, treatment 2, and interaction effects.

Examples

```
# 8 Cluster repeated cross sectional SWD with main and interaction effects assumed
# to be the same with 0.6 effect size.
swd_2trt_interaction_power(RhoW = 0.2, ModelChoice = "RCS", RhoA = 0.2, IAC = 0,
                n.individuals = 15, n.clusters = 8, n.periods = 5, delta_1 = 0.6,
```

```
                 delta_2 = 0.6, delta_3 = 0.6, typeIerror = 0.05/3,
                Sequence_Tx1 = c(2,3,2,3,4,5,NA,NA), Sequence_Tx2 = c(NA,NA,2,3,4,5,3,2))
```

---

swd_2trt_linearcontrast_power   *Stepped Wedge Design Power Calculation for Linear Contrasts of Treatment Effects*

Description

Calculate the power for detecting linear contrasts of two treatment effects in a two-treatment stepped wedge design (SWD) with additive treatment effects. We only consider complete designs. This function is based on Power Analysis for Stepped Wedge Trials with Two Treatments published in Stat Med in 2022 by Phillip Sundin and Catherine Crespi. This manuscript uses the normal distribution and the Wald Test to derive power.

Usage

```
swd_2trt_linearcontrast_power(
  RhoW,
  ModelChoice,
  n.individuals,
  n.clusters,
  n.periods,
  delta_12,
  RhoA,
  IAC,
  Sequence_Tx1,
  Sequence_Tx2,
  typeIerror
)
```

Arguments

RhoW	The intraclass correlation coefficient, which refers to the correlation between outcomes of two different individuals in the same cluster at the same time.
ModelChoice	A string indicating either a repeated cross sectional design ("RCS"), nested exchangeable ("NEC"), Cohort ("Cohort"), or any one of these three designs in their standardized form. Only one of "RCS Standardized", "RCS", "Cohort Standardized", "Cohort", "NEC Standardized", or "NEC" can be specified. The standardized design refers to dividing by the standard deviation of y (the outcome) in the covariance specification.
n.individuals	The number of individuals in each condition at one time period.
n.clusters	The number of clusters
n.periods	The number of time periods.
delta_12	The standardized difference between the two treatment effects to be powered on.
RhoA	The correlation between two observations in the same individual in the same cluster but different time periods. RCS models assume RhoW = RhoA, and Cohort models assume RhoA >= RhoW.

IAC	The individual auto-correlation, which is the proportion of the individual-level variance that is time-invariant. IAC = 0 in a RCS.
Sequence_Tx1	A vector representing the time points (periods) at which each cluster transitions from control to treatment 1 (ie the first occurence).
Sequence_Tx2	A vector representing the time points (periods) at which each cluster transitions from control to treatment 2 (ie the first occurence).
typeIerror	The significance level.

Value

Power for detecting the difference of treatment effects

Examples

```
# Late factorial SWD design with 12 clusters
swd_2trt_linearcontrast_power(RhoW = 0.2, ModelChoice="RCS", IAC = 0,
                      RhoA = 0.2, n.individuals = 15, delta_12 = 0.4,
                      n.clusters = 12, n.periods = 4,
                      Sequence_Tx1 = c(2,2,3,3,4,4,4,4,4,4,4,4),
                      Sequence_Tx2 = c(4,4,4,4,4,4,2,2,3,3,4,4),
                      typeIerror = 0.05)

# Example where there are different potential designs to be considered #
#creating data frame for power as a function of rho_w
PowerTable <- data.frame(RhoW = c(rep(seq(0,0.4,by=0.01),times=3)))
PowerTable$DesignChoice <- c(rep('"Late" Factorial Design, 12 clusters',times=41),
                             rep('"Early" Factorial Design, 10 clusters',times=41),
                             rep("Concurrent Design, 12 clusters",times=41))

PowerTable$n.clusters <- c(rep(12,times=41), rep(10,times=41), rep(12,times=41))

PowerTable$Sequence1 <- c(rep(list(c(2,2,3,3,4,4,4,4,4,4,4,4)),times=41),
                          rep(list(c(2,2,2,3,4,4,3,4,4,4)),times=41),
                          rep(list(c(2,2,3,3,4,4,NA,NA,NA,NA,NA,NA)),times=41))

PowerTable$Sequence2 <- c(rep(list(c(4,4,4,4,4,4,2,2,3,3,4,4)),times=41),
                          rep(list(c(4,4,4,3,4,4,3,2,2,2)),times=41),
                          rep(list(c(NA,NA,NA,NA,NA,NA,4,4,3,3,2,2)),times=41))

PowerTable$Power <- mapply(swd_2trt_linearcontrast_power,
                        RhoW = PowerTable$RhoW,
                        ModelChoice="RCS",
                        IAC = 0, RhoA = PowerTable$RhoW,
                        n.individuals = 15, delta_12 = 0.4,
                        n.clusters = PowerTable$n.clusters,
                        n.periods = 4, typeIerror = 0.05,
                        Sequence_Tx1 = PowerTable$Sequence1,
                        Sequence_Tx2 = PowerTable$Sequence2)

##removing unnecessary columns of clusters
PowerTable <- PowerTable[ ,-which(names(PowerTable) %in%
            c("Sequence1","Sequence2","n.clusters"))]
PowerTable
```

Index