Package ‘pwranova’

October 7, 2025

Title Power Analysis of Flexible ANOVA Designs and Related Tests

Version 1.0.2

Description Provides functions for conducting power analysis in ANOVA designs, including between-, within-, and mixed-factor designs, with full support for both main effects and interactions. The package allows calculation of statistical power, required total sample size, significance level, and minimal detectable effect sizes expressed as partial eta squared or Cohen's f for ANOVA terms and planned contrasts. In addition, complementary functions are included for common related tests such as t-tests and correlation tests, making the package a convenient toolkit for power analysis in experimental psychology and related fields.

License GPL-3

Encoding UTF-8

URL https://github.com/mutopsy/pwranova,
https://mutopsy.github.io/pwranova/

BugReports https://github.com/mutopsy/pwranova/issues

RoxygenNote 7.3.2

Depends R (>= 4.1.0)

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation no

Author Hiroyuki Muto [aut, cre] (ORCID: <https://orcid.org/0000-0002-0007-6019>)

Maintainer Hiroyuki Muto <mutopsy@omu.ac.jp>

Repository CRAN

Date/Publication 2025-10-07 18:30:09 UTC

Contents

`cohensf_to_peta2` *Convert Cohen's f to Partial Eta Squared*

Description

Converts Cohen's f to partial eta squared (η_p^2) using the standard definition in Cohen (1988).

Usage

```
cohensf_to_peta2(f)
```

Arguments

`f`	A numeric vector of Cohen's f values. Each value must be greater than or equal to 0.

Details

The conversion is defined as:

$$\eta_p^2 = \frac{f^2}{1 + f^2}$$

This follows from the relationship: $f = \sqrt{\eta_p^2/(1 - \eta_p^2)}$

Value

A numeric vector of partial eta squared values.

References

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). Hillsdale, NJ: Lawrence Erlbaum Associates.

See Also

peta2_to_cohensf

Examples

```
# Convert a single Cohen's f value
cohensf_to_peta2(0.25)

# Convert multiple values
cohensf_to_peta2(c(0.1, 0.25, 0.4))
```

`peta2_to_cohensf` *Convert Partial Eta Squared to Cohen's f*

Description

Converts partial eta squared (η_p^2) to Cohen's f using the standard definition in Cohen (1988).

Usage

```
peta2_to_cohensf(peta2)
```

Arguments

`peta2`	A numeric vector of partial eta squared values. Each value must be within the range of 0 to 1.

Details

The conversion is defined as:

$$f = \sqrt{\eta_p^2/(1 - \eta_p^2)}$$

This follows from the inverse relationship:

$$\eta_p^2 = \frac{f^2}{1 + f^2}$$

Value

A numeric vector of Cohen's f values.

References

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). Hillsdale, NJ: Lawrence Erlbaum Associates.

See Also

cohensf_to_peta2

Examples

```
# Convert a single partial eta squared value
peta2_to_cohensf(0.06)

# Convert multiple values
peta2_to_cohensf(c(0.01, 0.06, 0.14))
```

pwranova — *Power Analysis for Between-, Within-, or Mixed-Factor ANOVA*

Description

Computes power, required total sample size, alpha, or minimal detectable effect size for fixed-effects terms in between-/within-/mixed-factor ANOVA designs.

Usage

```
pwranova(
  nlevels_b = NULL,
  nlevels_w = NULL,
  n_total = NULL,
  alpha = NULL,
  power = NULL,
  cohensf = NULL,
  peta2 = NULL,
  epsilon = 1,
  target = NULL,
  max_nfactor = 6,
  nlim = c(2, 10000)
)
```

Arguments

`nlevels_b`	Integer scalar or vector. Numbers of levels for between-subjects factors. Omit or set `NULL` if there is no between-subjects factor.
`nlevels_w`	Integer scalar or vector. Numbers of levels for within-subjects factors. Omit or set `NULL` if there is no within-subjects factor.
`n_total`	Integer scalar or vector. Total sample size across all groups. If `NULL`, the function solves for `n_total`.
`alpha`	Numeric in $(0, 1)$. If `NULL`, it is solved for.
`power`	Numeric in $(0, 1)$. If `NULL`, it is computed; if `n_total` is `NULL`, `n_total` is solved to achieve this power.
`cohensf`	Numeric (non-negative). Cohen's f. If `NULL`, it is derived from `peta2` when available. If both effect-size arguments (`cohensf` and `peta2`) are `NULL`, the effect size is treated as unknown and solved for given `n_total`, `alpha`, and `power`.
`peta2`	Numeric in $(0, 1)$. Partial eta squared. If `NULL`, it is derived from `cohensf` when available.
`epsilon`	Numeric in $(0, 1]$. Nonsphericity parameter applied to within-subjects terms with $\mathrm{df}_1 \geq 2$. Ignored if there is no within-subjects factor or if all within factors have two levels.
`target`	Character vector of term labels to compute (e.g., `"B1"`, `"W1"`, `"B1:W1"`, ...). If `NULL`, all terms are returned.

max_nfactor	Integer. Safety cap for the total number of factors.
nlim	Integer length-2. Search range of total n when solving sample size.

Details

- Fixed-effects, balanced designs are assumed. All groups/cells have equal cell sizes and effects are tested with standard fixed-effects ANOVA models.
- Numerator degrees of freedom for within-subjects terms with $\mathrm{df}_1 \geq 2$ are adjusted by the nonsphericity parameter epsilon.
- Denominator degrees of freedom follow standard mixed-ANOVA formulas and are multiplied by the same epsilon for within-subjects terms.
- Critical values are computed from the central *F*-distribution; power uses the noncentral *F*-distribution with noncentrality parameter $\lambda = f^2 \cdot n_{\mathrm{total}}$.
- Effect-size inputs can be given as Cohen's f or partial eta-squared η_p^2 (internally converted via $f = \sqrt{\eta_p^2/(1-\eta_p^2)}$). If both are NULL, the minimal detectable effect size is solved for given n_total, alpha, and power.
- Exactly one of n_total, an effect-size specification (cohensf/peta2), alpha, or power must be NULL; that quantity is then solved.
- **Validation against GPower:** For the subset of designs supported by GPower (between-, within-, and mixed-factor ANOVA with equal cell sizes), pwranova() was validated to produce results identical to those of GPower.

Value

A data frame with S3 class:

- "cal_power" when power is calculated given n_total, alpha, and effect size;
- "cal_n" or "cal_ns" when n_total is solved;
- "cal_alpha" or "cal_alphas" when alpha is solved;
- "cal_es" when minimal detectable effect sizes are solved.

Columns include term, df_num, df_denom, n_total, alpha, power, cohensf, peta2, F_critical, ncp, epsilon.

References

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). Hillsdale, NJ: Lawrence Erlbaum Associates.

Examples

```
# One between factor (k = 3), one within factor (m = 4), compute power
pwranova(nlevels_b = 3, nlevels_w = 4, n_total = 60,
         cohensf = 0.25, alpha = 0.05, power = NULL, epsilon = 0.8)

# Solve required total N for target power
pwranova(nlevels_b = 2, nlevels_w = NULL, n_total = NULL,
         peta2 = 0.06, alpha = 0.05, power = 0.8)
```

pwrcontrast *Power Analysis for Planned Contrast in Between- or Within-Factor ANOVA*

Description

Computes power, required total sample size, alpha, or minimal detectable effect size for a **single planned contrast** (1 df) in between-participants or paired/repeated-measures settings.

Usage

```
pwrcontrast(
  weight = NULL,
  paired = FALSE,
  n_total = NULL,
  cohensf = NULL,
  peta2 = NULL,
  alpha = NULL,
  power = NULL,
  nlim = c(2, 10000)
)
```

Arguments

`weight`	Numeric vector (length $K \geq 2$). Contrast weights whose sum must be zero.
`paired`	Logical. `FALSE` for between-subjects (default), `TRUE` for paired/repeated-measures.
`n_total`	Integer or integer vector. Total sample size(s). If `NULL`, the function solves for `n_total`.
`cohensf`	Numeric (non-negative). Cohen's f. If `NULL`, it is derived from `peta2` when available.
`peta2`	Numeric in $(0, 1)$. Partial eta squared. If `NULL`, it is derived from `cohensf` when available.
`alpha`	Numeric in $(0, 1)$. If `NULL`, it is solved for.
`power`	Numeric in $(0, 1)$. If `NULL`, it is computed; if `n_total` is `NULL`, `n_total` is solved to achieve this power.
`nlim`	Integer length-2. Search range of total `n` when solving sample size.

Details

For a contrast with weights $w_1, \ldots, w_K$ that sum to zero, the numerator df is 1. The denominator df is $n - K$ for between-subjects (unpaired) designs and $(n-1)(K-1)$ for paired/repeated-measures designs. Power uses the noncentral F-with $\lambda = f^2 \cdot n_{\text{total}}$.

- Contrast weights (`weight`) are not centered internally; only the zero-sum condition is enforced (up to numerical tolerance).

- When paired = FALSE, the total sample size n_total must be a multiple of the number of contrast groups K.
- Exactly one of n_total, an effect-size specification (cohensf/peta2), alpha, or power must be NULL; that quantity is then solved.
- Critical values are computed from the central F-distribution; power is based on the noncentral F-distribution with noncentrality parameter $\lambda = f^2 \cdot n_{\text{total}}$.
- Effect-size inputs can be given as Cohen's f or partial eta-squared η_p^2 (internally converted via $f = \sqrt{\eta_p^2/(1-\eta_p^2)}$). If both are NULL, the minimal detectable effect size is solved for given n_total, alpha, and power.

Value

A one-row data frame with class:

- "cal_power" when power is calculated given n_total, alpha, and effect size;
- "cal_n" when n_total is solved;
- "cal_alpha" when alpha is solved;
- "cal_es" when minimal detectable effect sizes are solved.

Columns: term (always "contrast"), weight (comma-separated string), df_num, df_denom, n_total, alpha, power, cohensf, peta2, F_critical, ncp.

References

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). Hillsdale, NJ: Lawrence Erlbaum Associates.

Examples

```
# Two-group contrast (1, -1), between-subjects: compute power
pwrcontrast(weight = c(1, -1), paired = FALSE,
            n_total = 40, cohensf = 0.25, alpha = 0.05)

# Four-level contrast (e.g., Helmert-like), solve required N for target power
pwrcontrast(weight = c(3, -1, -1, -1), paired = FALSE,
            n_total = NULL, peta2 = 0.06, alpha = 0.05, power = 0.80)

# Paired contrast across K=3 conditions
pwrcontrast(weight = c(1, 0, -1), paired = TRUE,
            n_total = NULL, cohensf = 0.2, alpha = 0.05, power = 0.9)
```

`pwrcortest` *Power Analysis for Pearson Correlation*

Description

Computes statistical power, required total sample size, α, or the minimal detectable correlation coefficient for a Pearson correlation test. Two computational methods are supported: exact noncentral *t* (`method = "t"`) and Fisher's z-transformation with normal approximation (`method = "z"`).

Usage

```
pwrcortest(
  alternative = c("two.sided", "one.sided"),
  n_total = NULL,
  alpha = NULL,
  power = NULL,
  rho = NULL,
  method = c("t", "z"),
  bias_correction = FALSE,
  nlim = c(2, 10000)
)
```

Arguments

`alternative`	Character. Either `"two.sided"` or `"one.sided"`.
`n_total`	Integer scalar. Total sample size (n). Must be ≥ 3 for `method = "t"` and ≥ 4 for `method = "z"`. If `NULL`, the function solves for `n_total`.
`alpha`	Numeric in $(0, 1)$. If `NULL`, it is solved for.
`power`	Numeric in $(0, 1)$. If `NULL`, it is computed; if `n_total` is `NULL`, `n_total` is solved to attain this power.
`rho`	Numeric correlation coefficient in $(-1, 1)$, nonzero. If `NULL`, `rho` is solved for given the other inputs.
`method`	Character. Either `"t"` (noncentral *t*-distribution) or `"z"` (Fisher's z transformation with normal approximation).
`bias_correction`	Logical. Applies only to `method = "z"`. If `TRUE`, uses the bias-corrected Fisher z-transformation $z_p = \text{atanh}(r) + r/(2(n-1))$.
`nlim`	Integer vector of length 2. Search range of `n_total` when solving sample size.

Details

- Exactly one of `n_total`, `rho`, `alpha`, or `power` must be `NULL`; that quantity is then solved.
- For `method = "t"`, computations are based on the noncentral *t*-distribution with noncentrality parameter $\lambda = \frac{\rho}{\sqrt{1-\rho^2}}\sqrt{n}$.

- For method = "z", computations use Fisher's z transformation of the population correlation, $z_\rho = \operatorname{atanh}(\rho)$. Let $W = \sqrt{n-3}\, z$. Under the alternative hypothesis, $W \sim \text{Normal}(\mu, 1)$ with $\mu = \sqrt{n-3}\, z_\rho$. If bias_correction = TRUE, ρ is first bias-corrected before applying Fisher's transform. Critical values are taken from the central normal distribution under $H_0 : \rho = 0$ (i.e., $W \sim \text{Normal}(0, 1)$ under the null). The returned ncp equals μ.
- **Validation against GPower:** Results have been confirmed to match those produced by GPower for equivalent correlation tests using the noncentral *t*-distribution.
- *Note:* Results from method = "z" will not exactly match pwr::pwr.r.test, because pwr uses a hybrid approach combining the Fisher-z approximation with a *t*-based critical value.

Value

A one-row data.frame with class "cal_power", "cal_n", "cal_alpha", or "cal_es", depending on the solved quantity. Columns:

- df (only for method = "t")
- n_total, alpha, power
- rho, t_critical or z_critical
- ncp (noncentrality parameter or mean under the alternative: see Details)

Examples

```
# (1) Compute power for rho = 0.3, N = 50, two-sided test
pwrcortest(alternative = "two.sided", n_total = 50, rho = 0.3, alpha = 0.05)

# (2) Solve required N for target power, using Fisher-z method
pwrcortest(method = "z", rho = 0.2, alpha = 0.05, power = 0.8)

# (3) Solve minimal detectable correlation
pwrcortest(n_total = 60, alpha = 0.05, power = 0.9, rho = NULL)
```

---

pwrttest — *Power Analysis for One-/Two-Sample and Paired t Tests*

Description

Computes statistical power, required total sample size, α, or the minimal detectable effect size for a *t*-test in one of three designs: one-sample, two-sample (independent), or paired/repeated measures.

Usage

```
pwrttest(
  paired = FALSE,
  onesample = FALSE,
  n_total = NULL,
  alpha = NULL,
```

```
  power = NULL,
  delta = NULL,
  cohensf = NULL,
  peta2 = NULL,
  alternative = c("two.sided", "one.sided"),
  nlim = c(2, 10000)
)
```

Arguments

`paired`	Logical. `FALSE` for two-sample (independent; default), `TRUE` for paired/repeated-measures. Ignored when `onesample = TRUE`.
`onesample`	Logical. `TRUE` for the one-sample *t*-test; if `TRUE`, `paired` is ignored.
`n_total`	Integer scalar. Total sample size. If `NULL`, the function solves for `n_total`.
`alpha`	Numeric in $(0, 1)$. If `NULL`, it is solved for given the other inputs.
`power`	Numeric in $(0, 1)$. If `NULL`, it is computed; if `n_total` is `NULL`, `n_total` is solved to attain this power.
`delta`	Numeric (non-negative). Cohen's d-type effect size. If `NULL`, it is derived from `cohensf` or `peta2` when available. If all three effect-size arguments (`delta`, `cohensf`, `peta2`) are `NULL`, then the effect size is treated as the unknown quantity and is solved for given `n_total`, `alpha`, and `power`. The exact definition depends on the design: • *One-sample*: Cohen's $d = (\mu - \mu_0)/\sigma$. • *Paired*: Cohen's $d_z = \bar{d}/s_d$, i.e., the mean of the difference scores divided by their standard deviation. • *Two-sample (equal allocation)*: Cohen's d is defined as the mean difference divided by the pooled standard deviation; internally related to f via $d = 2f$. If `NULL`, `delta` is derived from `cohensf` or `peta2` when available.
`cohensf`	Numeric (non-negative). Cohen's f. If `NULL`, it can be derived from `delta`; if `delta` is supplied, `cohensf` is ignored. Effect-size relations by design: • Two-sample (equal allocation): $d = 2f$ • Paired: $d_z = f$ • One-sample: f and η_p^2 are not supported
`peta2`	Numeric in $(0, 1)$. Partial eta squared. If `NULL`, it can be derived from `cohensf`; if `delta` is supplied, `peta2` is ignored. Not defined for one-sample designs.
`alternative`	Character. Either `"two.sided"` or `"one.sided"`.
`nlim`	Integer vector of length 2. Search range of total `n` when solving sample size.

Details

- If multiple effect-size arguments are supplied (`delta`, `cohensf`, `peta2`), precedence is `delta` $>$ `cohensf` $>$ `peta2`; the rest are ignored with a warning.
- For the two-sample design, equal allocation is assumed; `n_total` must be even when provided, and the solved `n_total` will be an even number.

- For the paired design, the effect size is interpreted as d_z.
- Computations use the central and noncentral *t*-distributions (`stats::qt`, `stats::pt`); root finding uses `stats::uniroot()` where needed.
- Results have been validated to match those produced by G*Power for equivalent one-sample, paired, and two-sample *t* tests.

Value

A one-row `data.frame` with class `"cal_power"`, `"cal_n"`, `"cal_alpha"`, or `"cal_es"`, depending on the solved quantity. Columns: `df`, `n_total`, `alpha`, `power`, `delta`, `cohensf`, `peta2`, `t_critical`, `ncp`.

Examples

```
# (1) Two-sample (independent), compute power given N and d
pwrttest(paired = FALSE, onesample = FALSE, alternative = "two.sided",
         n_total = 128, delta = 0.50, alpha = 0.05)

# (2) Paired t-test, solve required N for target power
pwrttest(paired = TRUE, onesample = FALSE, alternative = "one.sided",
         n_total = NULL, delta = 0.40, alpha = 0.05, power = 0.90)

# (3) One-sample t-test, solve alpha given N and power
pwrttest(onesample = TRUE, alternative = "two.sided",
         n_total = 40, delta = 0.40, alpha = NULL, power = 0.80)

# (4) Two-sample, specify effect via f or partial eta^2 (converted internally)
pwrttest(paired = FALSE, cohensf = 0.25, n_total = NULL, alpha = 0.05, power = 0.80)
```

Index