Package 'Keng'

November 24, 2024

Title Knock Errors Off Nice Guesses

Version 2024.11.25

Description Miscellaneous functions and data used in Qingyao's psychological research and teaching.
Keng currently has a built-in dataset depress, and could (1) scale a vector,
(2) test the significance and compute the cut-off values of Pearson's r without raw data,
(3) compare lm()'s fitted outputs using R-squared, f_squared, post-hoc power,
and PRE (Proportional Reduction in Error, also called partial R-squared or partial Eta-squared).
(4) Calculate PRE from partial correlation, Cohen's f, or f_squared.
(5) Compute the post-
hoc power for one or a set of predictors in regression analysis without raw data,
(6) Plan the sample size for one or a set of predictors in regression analysis.

License CC BY 4.0

Language en-US

Encoding UTF-8

RoxygenNote 7.3.2

Imports stats

Suggests knitr, rmarkdown, car, effectsize, testthat (>= 3.0.0)

Config/testthat/edition 3

URL https://github.com/qyaozh/Keng

BugReports https://github.com/qyaozh/Keng/issues

Depends R (>= 2.10)

LazyData true

VignetteBuilder knitr

NeedsCompilation no

Author Qingyao Zhang [aut, cre] (<https://orcid.org/0000-0002-6891-5982>)

Maintainer Qingyao Zhang <qingyaozhang@outlook.com>

Repository CRAN

Date/Publication 2024-11-24 19:20:02 UTC

Contents

`calc_PRE` *Calculate PRE from Cohen's f, f_squared, or partial correlation*

Description

Calculate PRE from Cohen's f, f_squared, or partial correlation

Usage

```
calc_PRE(f = NULL, f_squared = NULL, r_p = NULL)
```

Arguments

`f`	Cohen's f. Cohen (1988) suggested >=0.1, >=0.25, and >=0.40 as cut-off values of f for small, medium, and large effect sizes, respectively.
`f_squared`	Cohen's f_squared. Cohen (1988) suggested >=0.02, >=0.15, and >=0.35 as cut-off values of f for small, medium, and large effect sizes, respectively.
`r_p`	Partial correlation.

Value

A list including PRE, r_p (partial correlation), Cohen's f_squared, and f.

References

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). Routledge.

Examples

```
calc_PRE(f = 0.1)
calc_PRE(f_squared = 0.02)
calc_PRE(r_p = 0.2)
```

compare_lm   *Compare lm()'s fitted outputs using PRE and R-squared.*

Description

Compare lm()'s fitted outputs using PRE and R-squared.

Usage

```
compare_lm(
  fitC = NULL,
  fitA = NULL,
  n = NULL,
  PC = NULL,
  PA = NULL,
  SSEC = NULL,
  SSEA = NULL
)
```

Arguments

`fitC`	The result of `lm()` of the Compact model (model C).
`fitA`	The result of `lm()` of the Augmented model (model A).
`n`	Sample size of the model C or model A. model C and model A must use the same sample, and hence have the same sample size.
`PC`	The number of parameters in model C.
`PA`	The number of parameters in model A. PA must be larger than PC.
`SSEC`	The Sum of Squared Errors (SSE) of model C.
`SSEA`	The Sum of Squared Errors of model A.

Details

`compare_lm()` compare model A with model C using *PRE* (Proportional Reduction in Error) , *R-squared*, *f_squared*, and post-hoc power. *PRE* is partial R-squared (called partial Eta-squared in Anova). There are two ways of using `compare_lm()`. The first is giving `compare_lm()` fitC and fitA. The second is giving *n*, *PC*, *PA*, *SSEC*, and *SSEA*. The first way is more convenient, and it minimizes precision loss by omitting copying-and-pasting. Note that the *F*-tests for *PRE* and that for R-squared change are equivalent. Please refer to Judd et al. (2017) for more details about *PRE*, and refer to Aberson (2019) for more details about *f_squared* and post-hoc power.

Value

A matrix with 11 rows and 4 columns. The first column reports information for baseline model (intercept-only model). the second for model C, the third for model A, and the fourth for the change (model A vs. model C). *SSE* (Sum of Squared Errors) and *df* of *SSE* for baseline model, model C, model A, and change (model A vs. model C) are reported in row 1 and row 2. The information in the fourth column are all for the change; put differently, These results could quantify the effect of one or a set of new parameters model A has but model C doesn't. If fitC and fitA are not inferior to the intercept-only model, *R-squared*, *Adjusted R-squared*, *PRE*, *PRE_adjusted*, and *f_squared* for the full model (compared with the baseline model) are reported for model C and model A. If model C or model A has at least one predictor, F -test with p, and post-hoc power would be computed for the corresponding full model.

References

Aberson, C. L. (2019). *Applied power analysis for the behavioral sciences*. Routledge.

Judd, C. M., McClelland, G. H., & Ryan, C. S. (2017). *Data analysis: A model Comparison approach to regression, ANOVA, and beyond*. Routledge.

Examples

```
x1 <- rnorm(193)
x2 <- rnorm(193)
y <- 0.3 + 0.2*x1 + 0.1*x2 + rnorm(193)
dat <- data.frame(y, x1, x2)
# Fix intercept to constant 1 using I().
fit1 <- lm(I(y - 1) ~ 0, dat)
# Free intercept.
fit2 <- lm(y ~ 1, dat)
compare_lm(fit1, fit2)
# One predictor.
fit3 <- lm(y ~ x1, dat)
compare_lm(fit2, fit3)
# Fix intercept to 0.3 using offset().
intercept <- rep(0.3, 193)
fit4 <- lm(y ~ 0 + x1 + offset(intercept), dat)
compare_lm(fit4, fit3)
# Two predictors.
fit5 <- lm(y ~ x1 + x2, dat)
compare_lm(fit2, fit5)
compare_lm(fit3, fit5)
# Fix slope of x2 to 0.05 using offset().
fit6 <- lm(y ~ x1 + offset(0.05*x2), dat)
compare_lm(fit6, fit5)
```

---

cut_r *Cut-off values of r given the sample size n.*

Description

Cut-off values of r given the sample size n.

Usage

```
cut_r(n)
```

Arguments

`n`	Sample size of the *r*.

Details

Given *n* and *p*, *t* and then *r* could be determined. The formula used could be found in `test_r()`'s documentation.

Value

A data.frame including the cut-off values of *r* at the significance levels of *p* = 0.1, 0.05, 0.01, 0.001. *r* with the absolute value larger than the cut-off value is significant at the corresponding significance level.

Examples

```
cut_r(193)
```

`depress` *Depression and Coping*

Description

A subset of data from a research about depression and coping.

Usage

```
depress
```

Format

`depress`**:**
A data frame with 94 rows and 237 columns:

id Participant id
class Class
grade Grade
elite Elite classes
intervene 0 = Control group, 1 = Intervention group
gender 0 = girl, 1 = boy
age Age in year
cope1i1p Cope scale, Time1, Item1, Problem-focused coping, 1 = very seldom, 5 = very often
cope1i3a Cope scale, Time1, Item3, Avoidance coping
cope1i5e cope scale, Time1, Item5, Emotion-focused coping
cope2i1p Cope scale, Time2, Item1, Problem-focused coping
depr1i1 Depression scale, Time1, Item1, 1 = very seldom, 5 = always

ecr1avo ECR-RS scale, Item1, attachment avoidance, 1 = very disagree, 7 = very agree

ecr2anx ECR-RS scale, Item2, attachment anxiety

dm1 Depression, Mean, Time1

pm1 Problem-focused coping, Mean, Time1

em1 Emotion-focused coping, Mean, Time1

am1 Avoidance coping, Mean, Time1

avo Attachment avoidance, Mean

anx Attachment anxiety, Mean

Source

Keng package.

`power_lm` *Compute the post-hoc power and/or plan the sample size for one or a set of predictors in linear regression*

Description

Compute the post-hoc power and/or plan the sample size for one or a set of predictors in linear regression

Usage

```
power_lm(PRE = 0.02, PC = 0, PA = 1, power = 0.8, sig.level = 0.05, n = NULL)
```

Arguments

`PRE`	Proportional Reduction in Error. *PRE* = The square of partial correlation. Cohen (1988) suggested >=0.02, >=0.13, and >=0.26 as cut-off values of PRE for small, medium, and large effect sizes, respectively.
`PC`	Number of parameters of model C (compact model) without focal predictors of interest.
`PA`	Number of parameters of model A (augmented model) with focal predictors of interest.
`power`	Expected statistical power for effects of focal predictors.
`sig.level`	Expected significance level for effects of focal predictors.
`n`	The current sample size. If *n* is given, the post-hoc power would be computed.

Value

A list with 4 items: (1) `post`, the post-hoc F-test, lambda (non-centrality parameter), and power for sample size *n*; (2)`minimum`, the minimum sample size required for focal predictors to reach the expected statistical power and significance level; (3) `prior`, a data.frame including `n_i`, `PC`, `PA`,`df_A_i`, `F_i`, `p_i`, `lambda_i`, `power_i`. `_i` indicates these statistics are the intermediate iterative results. Each row of `prior` presents results for one possible sample size `n_i`. Given `n_i`, `df_A_i`, `F_i`, `p_i`, `lambda_i` and `power_i` would be computed accordingly. (4) A plot of power against sample size *n*. The cut-off value of *n* for expected statistical power `power` and expected significance level `sig.level` is annotated on the plot.

References

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). Routledge.

Examples

```
power_lm()
```

`Scale` *Scale a vector*

Description

Scale a vector

Usage

```
Scale(x, m = NULL, sd = NULL, oadvances = NULL)
```

Arguments

`x`	The original vector.
`m`	The expected Mean of the scaled vector.
`sd`	The expected Standard Deviation (unit) of the scaled vector.
`oadvances`	The distance the Origin of x advances by.

Details

To scale `x`, its origin, or unit (*sd*), or both, could be changed.

If `m` = 0 or `NULL`, and `sd` = `NULL`, `x` would be mean-centered.

If `m` is a non-zero number, and `sd` = `NULL`, the mean of `x` would be transformed to m.

If `m` = 0 or `NULL`, and `sd` = 1, `x` would be standardized to be its z-score with $m = 0$ and $m = 1$.

The standardized score is not necessarily the z-score. If neither `m` nor `sd` is `NULL`, `x` would be standardized to be a vector whose mean and standard deviation would be `m` and `sd`, respectively.

To standardize `x`, the mean and standard deviation of `x` are needed and computed, for which the missing values of `x` are removed if any.

If `oadvances` is not `NULL`, the origin of `x` will advance with the standard deviation being unchanged. In this case, `Scale()` could be used to pick points in simple slope analysis for moderation models. Note that when `oadvances` is not `NULL`, `m` and `sd` must be NULL.

Value

The scaled vector.

Examples

```
(x <- rnorm(10, 5, 2))
# Mean-center x.
Scale(x)
# Transform the mean of x to 3.
Scale(x, m = 3)
# Transform x to its z-score.
Scale(x, sd = 1)
# Standardize x with m = 100 and sd = 15.
Scale(x, m = 100, sd = 15)
# The origin of x advances by 3.
Scale(x, oadvances = 3)
```

`test_r` *Test r using the t-test and Fisher's z given r and n.*

Description

Test r using the t-test and Fisher's z given r and n.

Usage

```
test_r(r, n)
```

Arguments

`r`	Pearson's correlation.
`n`	Sample size of *r*.

Details

To test the significance of the *r* using one-sample *t*-test, the *SE* of the *r* is determined by the following formula: `SE = sqrt((1 - r^2)/(n - 2))`. Another way is transforming *r* to Fisher's z using the following formula: `fz = atanh(r)` with the *SE* of `fz` being `sqrt(n - 3)`. Note that Fisher's z is commonly used to compare two Pearson's correlations from independent samples. Fisher's transformation is presented here only for satisfying the curiosity of users interested in the difference of *t* -test and Fisher's transformation.

Value

A list including *r*, *t* -test of *r* (`SE_r`, `t`, `p_r`), 95% *CI* of *r* based on *t* -test (`LLCI_r_t`, `ULCI_r_t`), *fz* (Fisher's z) of *r*, *z* -test of Fisher's z (`SE_fz`, `z`, `p_fz`), and 95% *CI* of *r* derived from *fz*. Note that the returned *CI* of *r* may be out of *r*'s valid range [-1, 1]. This "error" is deliberately left to users, who should correct the CI manually when reporting.

Examples

```
test_r(0.2, 193)

# compare the p-values of t-test and Fisher's transformation
for (i in seq(30, 200, 10)) {
cat(c(
      "n =", i, ",",
       format(
        abs(test_r(0.2, i)[[1]][4] - test_r(0.2, i)[[2]][4]),
        nsmall = 12, scientific = FALSE)),
    fill = TRUE)
}
```

Index