

Package ‘effectsize’

October 31, 2022

Type Package

Title Indices of Effect Size

Version 0.8.2

Maintainer Mattan S. Ben-Shachar <mattansb@msbstats.info>

Description Provide utilities to work with indices of effect size for a wide variety of models and hypothesis tests (see list of supported models using the function 'insight::supported_models()'), allowing computation of and conversion between indices such as Cohen's d, r, odds, etc.

References: Ben-Shachar et al. (2020) <doi:10.21105/joss.02815>.

License GPL-3

URL <https://easystats.github.io/effectsize/>

BugReports <https://github.com/easystats/effectsize/issues/>

Depends R (>= 3.6)

Imports bayestestR (>= 0.13.0), insight (>= 0.18.4), parameters (>= 0.19.0), performance (>= 0.10.0), datawizard (>= 0.6.2), stats, utils

Suggests correlation (>= 0.8.2), see (>= 0.7.3), afex, BayesFactor, boot, brms, car, emmeans, knitr, lavaan, lme4, lmerTest, MASS, mgcv, parsnip, rmarkdown, rms, rstanarm, rstantools, testthat (>= 3.1.0)

VignetteBuilder knitr

Encoding UTF-8

Language en-US

RoxygenNote 7.2.1.9000

Config/testthat/edition 3

Config/testthat/parallel true

Config/Needs/website rstudio/bslib, r-lib/pkgdown, easystats/easystatstemplate

NeedsCompilation no

Author Mattan S. Ben-Shachar [aut, cre]
 (<<https://orcid.org/0000-0002-4287-4801>>, @mattansb),
 Dominique Makowski [aut] (<<https://orcid.org/0000-0001-5375-9967>>,
 @Dom_Makowski),
 Daniel Lüdtke [aut] (<<https://orcid.org/0000-0002-8895-3206>>,
 @strengjacke),
 Indrajeet Patil [aut] (<<https://orcid.org/0000-0003-1995-6531>>,
 @patilindrajeets),
 Brenton M. Wiernik [aut] (<<https://orcid.org/0000-0001-9560-6336>>,
 @bmwiernik),
 Ken Kelley [ctb],
 David Stanley [ctb],
 Jessica Burnett [rev] (<<https://orcid.org/0000-0002-0896-5099>>),
 Johannes Karreth [rev] (<<https://orcid.org/0000-0003-4586-7153>>)

Repository CRAN

Date/Publication 2022-10-31 10:40:09 UTC

R topics documented:

chisq_to_phi	3
cohens_d	7
cohens_g	11
diff_to_cles	13
d_to_r	15
effectsize.BFBayesFactor	16
effectsize_API	19
effectsize_CIs	20
effectsize_deprecated	23
effectsize_options	24
equivalence_test.effectsize_table	25
eta2_to_f2	27
eta_squared	28
food_class	34
format_standardize	34
F_to_eta2	35
hardlyworking	39
interpret	40
interpret_bf	42
interpret_cohens_d	43
interpret_cohens_g	45
interpret_direction	46
interpret_ess	46
interpret_gfi	47
interpret_icc	50
interpret_kendalls_w	51
interpret_oddsratio	52
interpret_omega_squared	53

interpret_p	54
interpret_pd	55
interpret_r	56
interpret_r2	58
interpret_rope	59
interpret_vif	60
is_effectsize_name	61
mahalanobis_d	61
Music_preferences	64
Music_preferences2	65
oddsratio	65
oddsratio_to_riskratio	67
odds_to_probs	68
phi	69
plot.effectsize_table	73
p_superiority	74
rank_biserial	79
rank_epsilon_squared	82
RCT_table	85
rules	86
screening_test	87
sd_pooled	87
Smoking_FASD	88
t_to_d	89

Index**93**

chisq_to_phi	<i>Convert χ^2 to ϕ and Other Correlation-like Effect Sizes</i>
--------------	---

Description

Convert between χ^2 (chi-square), ϕ (phi), Cramer's V , Tschuprow's T , Cohen's w , F_{ei} and Pearson's C for contingency tables or goodness of fit.

Usage

```
chisq_to_phi(
  chisq,
  n,
  nrow = 2,
  ncol = 2,
  adjust = TRUE,
  ci = 0.95,
  alternative = "greater",
  ...
)
```

```
chisq_to_cohens_w(  
  chisq,  
  n,  
  nrow,  
  ncol,  
  p,  
  ci = 0.95,  
  alternative = "greater",  
  ...  
)  
  
chisq_to_cramers_v(  
  chisq,  
  n,  
  nrow,  
  ncol,  
  adjust = TRUE,  
  ci = 0.95,  
  alternative = "greater",  
  ...  
)  
  
chisq_to_tschuprows_t(  
  chisq,  
  n,  
  nrow,  
  ncol,  
  ci = 0.95,  
  alternative = "greater",  
  ...  
)  
  
chisq_to_fei(chisq, n, nrow, ncol, p, ci = 0.95, alternative = "greater", ...)  
  
chisq_to_pearsons_c(  
  chisq,  
  n,  
  nrow,  
  ncol,  
  ci = 0.95,  
  alternative = "greater",  
  ...  
)  
  
phi_to_chisq(phi, n, ...)
```

Arguments

chisq The χ^2 (chi-square) statistic.

n	Total sample size.
nrow, ncol	The number of rows/columns in the contingency table.
adjust	Should the effect size be bias-corrected? Defaults to TRUE; Advisable for small samples and large tables.
ci	Confidence Interval (CI) level
alternative	a character string specifying the alternative hypothesis; Controls the type of CI returned: "greater" (default) or "less" (one-sided CI), or "two.sided" (default, two-sided CI). Partial matching is allowed (e.g., "g", "l", "two"...). See <i>One-Sided CIs</i> in effectsize_CIs .
...	Arguments passed to or from other methods.
p	Vector of expected values. See stats::chisq.test() .
phi	The ϕ (phi) statistic.

Details

These functions use the following formulas:

$$\phi = w = \sqrt{\chi^2/n}$$

$$\text{Cramer's } V = \phi / \sqrt{\min(nrow, ncol) - 1}$$

$$\text{Tschuprow's } T = \phi / \sqrt[4]{(nrow - 1) \times (ncol - 1)}$$

$$Fei = \phi / \sqrt{[1/\min(p_E)] - 1}$$

Where p_E are the expected probabilities.

$$\text{Pearson's } C = \sqrt{\chi^2/(\chi^2 + n)}$$

For bias-adjusted versions of ϕ and V , see [Bergsma, 2013](#).

Value

A data frame with the effect size(s), and confidence interval(s). See [cramers_v\(\)](#).

Confidence (Compatibility) Intervals (CIs)

Unless stated otherwise, confidence (compatibility) intervals (CIs) are estimated using the non-centrality parameter method (also called the "pivot method"). This method finds the noncentrality parameter (" ncp ") of a noncentral t , F , or χ^2 distribution that places the observed t , F , or χ^2 test statistic at the desired probability point of the distribution. For example, if the observed t statistic is 2.0, with 50 degrees of freedom, for which cumulative noncentral t distribution is $t = 2.0$ the .025 quantile (answer: the noncentral t distribution with $ncp = .04$)? After estimating these confidence bounds on the ncp , they are converted into the effect size metric to obtain a confidence interval for the effect size (Steiger, 2004).

For additional details on estimation and troubleshooting, see [effectsize_CIs](#).

CI and Significance Tests

"Confidence intervals on measures of effect size convey all the information in a hypothesis test, and more." (Steiger, 2004). Confidence (compatibility) intervals and p values are complementary summaries of parameter uncertainty given the observed data. A dichotomous hypothesis test could be performed with either a CI or a p value. The 100 (1 - α)% confidence interval contains all of the parameter values for which $p > \alpha$ for the current data and model. For example, a 95% confidence interval contains all of the values for which $p > .05$.

Note that a confidence interval including 0 *does not* indicate that the null (no effect) is true. Rather, it suggests that the observed data together with the model and its assumptions combined do not provided clear evidence against a parameter value of 0 (same as with any other value in the interval), with the level of this evidence defined by the chosen α level (Rafi & Greenland, 2020; Schweder & Hjort, 2016; Xie & Singh, 2013). To infer no effect, additional judgments about what parameter values are "close enough" to 0 to be negligible are needed ("equivalence testing"; Bauer & Kiesser, 1996).

References

- Cumming, G., & Finch, S. (2001). A primer on the understanding, use, and calculation of confidence intervals that are based on central and noncentral distributions. *Educational and Psychological Measurement*, 61(4), 532-574.
- Bergsma, W. (2013). A bias-correction for Cramer's V and Tschuprow's T. *Journal of the Korean Statistical Society*, 42(3), 323-328.
- Johnston, J. E., Berry, K. J., & Mielke Jr, P. W. (2006). Measures of effect size for chi-squared and likelihood-ratio goodness-of-fit tests. *Perceptual and motor skills*, 103(2), 412-414.
- Rosenberg, M. S. (2010). A generalized formula for converting chi-square tests to effect sizes for meta-analysis. *PloS one*, 5(4), e10059.

See Also

[phi\(\)](#) for more details.

Other effect size from test statistic: [F_to_eta2\(\)](#), [t_to_d\(\)](#)

Examples

```
data("Music_preferences")

# chisq.test(Music_preferences)
#>
#> Pearson's Chi-squared test
#>
#> data: Music_preferences
#> X-squared = 95.508, df = 6, p-value < 2.2e-16
#>

chisq_to_cohens_w(95.508,
  n = sum(Music_preferences),
  nrow = nrow(Music_preferences),
```

```
    ncol = ncol(Music_preferences)
  )

data("Smoking_FASD")

# chisq.test(Smoking_FASD, p = c(0.015, 0.010, 0.975))
#>
#> Chi-squared test for given probabilities
#>
#> data:  Smoking_FASD
#> X-squared = 7.8521, df = 2, p-value = 0.01972

chisq_to_fei(
  7.8521,
  n = sum(Smoking_FASD),
  nrow = 1,
  ncol = 3,
  p = c(0.015, 0.010, 0.975)
)
```

cohens_d

Cohen's d and Other Standardized Differences

Description

Compute effect size indices for standardized differences: Cohen's d , Hedges' g and Glass's δ (Δ). (This function returns the **population** estimate.) Pair with any reported `stats::t.test()`.

Both Cohen's d and Hedges' g are the estimated the standardized difference between the means of two populations. Hedges' g provides a bias correction (using the exact method) to Cohen's d for small sample sizes. For sample sizes > 20 , the results for both statistics are roughly equivalent. Glass's δ is appropriate when the standard deviations are significantly different between the populations, as it uses only the *second* group's standard deviation.

Usage

```
cohens_d(
  x,
  y = NULL,
  data = NULL,
  pooled_sd = TRUE,
  mu = 0,
  paired = FALSE,
  ci = 0.95,
  alternative = "two.sided",
```

```

    verbose = TRUE,
    ...
)

hedges_g(
  x,
  y = NULL,
  data = NULL,
  pooled_sd = TRUE,
  mu = 0,
  paired = FALSE,
  ci = 0.95,
  alternative = "two.sided",
  verbose = TRUE,
  ...
)

glass_delta(
  x,
  y = NULL,
  data = NULL,
  mu = 0,
  ci = 0.95,
  alternative = "two.sided",
  verbose = TRUE,
  ...
)

```

Arguments

<code>x, y</code>	A numeric vector, or a character name of one in data. Any missing values (NAs) are dropped from the resulting vector. <code>x</code> can also be a formula (see <code>stats::t.test()</code>), in which case <code>y</code> is ignored.
<code>data</code>	An optional data frame containing the variables.
<code>pooled_sd</code>	If TRUE (default), a <code>sd_pooled()</code> is used (assuming equal variance). Else the mean SD from both groups is used instead.
<code>mu</code>	a number indicating the true value of the mean (or difference in means if you are performing a two sample test).
<code>paired</code>	If TRUE, the values of <code>x</code> and <code>y</code> are considered as paired. This produces an effect size that is equivalent to the one-sample effect size on <code>x - y</code> .
<code>ci</code>	Confidence Interval (CI) level
<code>alternative</code>	a character string specifying the alternative hypothesis; Controls the type of CI returned: "two.sided" (default, two-sided CI), "greater" or "less" (one-sided CI). Partial matching is allowed (e.g., "g", "l", "two"...). See <i>One-Sided CIs</i> in effectsize_CIs .
<code>verbose</code>	Toggle warnings and messages on or off.

... Arguments passed to or from other methods. When x is a formula, these can be subset and na.action.

Details

Set `pooled_sd = FALSE` for effect sizes that are to accompany a Welch's *t*-test (Delacre et al, 2021).

Value

A data frame with the effect size (Cohens_d, Hedges_g, Glass_delta) and their CIs (CI_low and CI_high).

Confidence (Compatibility) Intervals (CIs)

Unless stated otherwise, confidence (compatibility) intervals (CIs) are estimated using the non-centrality parameter method (also called the "pivot method"). This method finds the noncentrality parameter ("*ncp*") of a noncentral *t*, *F*, or χ^2 distribution that places the observed *t*, *F*, or χ^2 test statistic at the desired probability point of the distribution. For example, if the observed *t* statistic is 2.0, with 50 degrees of freedom, for which cumulative noncentral *t* distribution is *t* = 2.0 the .025 quantile (answer: the noncentral *t* distribution with *ncp* = .04)? After estimating these confidence bounds on the *ncp*, they are converted into the effect size metric to obtain a confidence interval for the effect size (Steiger, 2004).

For additional details on estimation and troubleshooting, see [effectsize_CIs](#).

CIs and Significance Tests

"Confidence intervals on measures of effect size convey all the information in a hypothesis test, and more." (Steiger, 2004). Confidence (compatibility) intervals and p values are complementary summaries of parameter uncertainty given the observed data. A dichotomous hypothesis test could be performed with either a CI or a p value. The 100 (1 - α)% confidence interval contains all of the parameter values for which $p > \alpha$ for the current data and model. For example, a 95% confidence interval contains all of the values for which $p > .05$.

Note that a confidence interval including 0 *does not* indicate that the null (no effect) is true. Rather, it suggests that the observed data together with the model and its assumptions combined do not provided clear evidence against a parameter value of 0 (same as with any other value in the interval), with the level of this evidence defined by the chosen α level (Rafi & Greenland, 2020; Schweder & Hjort, 2016; Xie & Singh, 2013). To infer no effect, additional judgments about what parameter values are "close enough" to 0 to be negligible are needed ("equivalence testing"; Bauer & Kiesser, 1996).

Note

The indices here give the population estimated standardized difference. Some statistical packages give the sample estimate instead (without applying Bessel's correction).

References

- Algina, J., Keselman, H. J., & Penfield, R. D. (2006). Confidence intervals for an effect size when variances are not equal. *Journal of Modern Applied Statistical Methods*, 5(1), 2.
- Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd Ed.). New York: Routledge.
- Delacre, M., Lakens, D., Ley, C., Liu, L., & Leys, C. (2021, May 7). Why Hedges' g*s based on the non-pooled standard deviation should be reported with Welch's t-test. <https://doi.org/10.31234/osf.io/tu6mp>
- Hedges, L. V. & Olkin, I. (1985). *Statistical methods for meta-analysis*. Orlando, FL: Academic Press.
- Hunter, J. E., & Schmidt, F. L. (2004). *Methods of meta-analysis: Correcting error and bias in research findings*. Sage.

See Also

[sd_pooled\(\)](#), [t_to_d\(\)](#), [r_to_d\(\)](#)

Other standardized differences: [mahalanobis_d\(\)](#), [p_superiority\(\)](#), [rank_biserial\(\)](#)

Examples

```
data(mtcars)
mtcars$am <- factor(mtcars$am)

# Two Independent Samples -----

(d <- cohens_d(mpg ~ am, data = mtcars))
# Same as:
# cohens_d("mpg", "am", data = mtcars)
# cohens_d(mtcars$mpg[mtcars$am=="0"], mtcars$mpg[mtcars$am=="1"])

# More options:
cohens_d(mpg ~ am, data = mtcars, pooled_sd = FALSE)
cohens_d(mpg ~ am, data = mtcars, mu = -5)
cohens_d(mpg ~ am, data = mtcars, alternative = "less")
hedges_g(mpg ~ am, data = mtcars)
glass_delta(mpg ~ am, data = mtcars)

# One Sample -----

cohens_d(wt ~ 1, data = mtcars)

# same as:
# cohens_d("wt", data = mtcars)
# cohens_d(mtcars$wt)

# More options:
cohens_d(wt ~ 1, data = mtcars, mu = 3)
hedges_g(wt ~ 1, data = mtcars, mu = 3)
```

```

# Paired Samples -----

data(sleep)

cohens_d(Pair(extra[group == 1], extra[group == 2]) ~ 1, data = sleep)

# same as:
# cohens_d(sleep$extra[sleep$group == 1], sleep$extra[sleep$group == 2], paired = TRUE)

# More options:
cohens_d(Pair(extra[group == 1], extra[group == 2]) ~ 1, data = sleep, mu = -1)
hedges_g(Pair(extra[group == 1], extra[group == 2]) ~ 1, data = sleep)

# Interpretation -----
interpret_cohens_d(-1.48, rules = "cohen1988")
interpret_hedges_g(-1.48, rules = "sawilowsky2009")
interpret_glass_delta(-1.48, rules = "gignac2016")
# Or:
interpret(d, rules = "sawilowsky2009")

# Common Language Effect Sizes
d_to_u3(1.48)
# Or:
print(d, append_CLES = TRUE)

```

cohens_g

Effect Size for Paired Contingency Tables

Description

Cohen's g is an effect size of asymmetry (or marginal heterogeneity) for dependent (paired) contingency tables ranging between 0 (perfect symmetry) and 0.5 (perfect asymmetry) (see `stats::mcnemar.test()`). (Note this is not *not* a measure of (dis)agreement between the pairs, but of (a)symmetry.)

Usage

```
cohens_g(x, y = NULL, ci = 0.95, alternative = "two.sided", ...)
```

Arguments

<code>x</code>	a numeric vector or matrix. <code>x</code> and <code>y</code> can also both be factors.
<code>y</code>	a numeric vector; ignored if <code>x</code> is a matrix. If <code>x</code> is a factor, <code>y</code> should be a factor of the same length.
<code>ci</code>	Confidence Interval (CI) level

alternative a character string specifying the alternative hypothesis; Controls the type of CI returned: "two.sided" (two-sided CI; default), "greater" (one-sided CI) or "less" (one-sided CI). Partial matching is allowed (e.g., "g", "1", "two"...). See *One-Sided CIs* in [effectsize_CIs](#).

... Ignored

Value

A data frame with the effect size (Cohens_g, Risk_ratio (possibly with the prefix log_), Cohens_h) and its CIs (CI_low and CI_high).

Confidence (Compatibility) Intervals (CIs)

Confidence intervals are based on the proportion ($P = g + 0.5$) confidence intervals returned by [stats::prop.test\(\)](#) (minus 0.5), which give a good close approximation.

CIs and Significance Tests

"Confidence intervals on measures of effect size convey all the information in a hypothesis test, and more." (Steiger, 2004). Confidence (compatibility) intervals and p values are complementary summaries of parameter uncertainty given the observed data. A dichotomous hypothesis test could be performed with either a CI or a p value. The 100 (1 - α)% confidence interval contains all of the parameter values for which $p > \alpha$ for the current data and model. For example, a 95% confidence interval contains all of the values for which $p > .05$.

Note that a confidence interval including 0 *does not* indicate that the null (no effect) is true. Rather, it suggests that the observed data together with the model and its assumptions combined do not provided clear evidence against a parameter value of 0 (same as with any other value in the interval), with the level of this evidence defined by the chosen α level (Rafi & Greenland, 2020; Schweder & Hjort, 2016; Xie & Singh, 2013). To infer no effect, additional judgments about what parameter values are "close enough" to 0 to be negligible are needed ("equivalence testing"; Bauer & Kiesser, 1996).

References

- Cohen, J. (1988). Statistical power analysis for the behavioral sciences (2nd Ed.). New York: Routledge.

See Also

Other effect sizes for contingency table: [oddsratio\(\)](#), [phi\(\)](#)

Examples

```
data("screening_test")

phi(screening_test$Diagnosis, screening_test$Test1)

phi(screening_test$Diagnosis, screening_test$Test2)
```

```
# Both tests seem comparable - but are the tests actually different?
(tests <- table(Test1 = screening_test$Test1, Test2 = screening_test$Test2))
mcnemar.test(tests)
cohens_g(tests)
# Test 2 gives a negative result more than test 1!
```

diff_to_cles

Convert Standardized Differences to Common Language Effect Sizes

Description

Convert Standardized Differences to Common Language Effect Sizes

Usage

```
d_to_p_superiority(d)
rb_to_p_superiority(rb)
rb_to_vda(rb)
d_to_u2(d)
d_to_u1(d)
d_to_u3(d)
d_to_overlap(d)
rb_to_wmw_odds(rb)
```

Arguments

d, rb A numeric vector of Cohen's d / rank-biserial correlation *or* the output from `cohens_d()` / `rank_biserial()`.

Details

This function use the following formulae for Cohen's *d*:

$$Pr(\text{superiority}) = \Phi(d/\sqrt{2})$$

$$\text{Cohen's } U_3 = \Phi(d)$$

$$\text{Cohen's } U_2 = \Phi(|d|/2)$$

$$\text{Cohen's } U_1 = (2 \times U_2 - 1)/U_2$$

$$\text{Overlap} = 2 \times \Phi(-|d|/2)$$

And the following for the rank-biserial correlation:

$$\text{Pr}(\text{superiority}) = (r_{rb} + 1)/2$$

$$WMW_{Odds} = \text{Pr}(\text{superiority}) / (1 - \text{Pr}(\text{superiority}))$$

Value

A list of Cohen's U_3 , U_2 , U_1 , U_0 , $U_{overlap}$, $\text{Pr}(\text{superiority})$, a numeric vector of $\text{Pr}(\text{superiority})$, or a data frame, depending on the input.

Note

For d , these calculations assume that the populations have equal variance and are normally distributed.

Vargha and Delaney's A is an alias for the non-parametric *probability of superiority*.

References

- Cohen, J. (1977). *Statistical power analysis for the behavioral sciences*. New York: Routledge.
- Reiser, B., & Faraggi, D. (1999). Confidence intervals for the overlapping coefficient: the normal equal variance case. *Journal of the Royal Statistical Society*, 48(3), 413-418.
- Ruscio, J. (2008). A probability-based measure of effect size: robustness to base rates and other factors. *Psychological methods*, 13(1), 19-30.

See Also

[cohens_u3\(\)](#) for descriptions of the effect sizes (also, [cohens_d\(\)](#), [rank_biserial\(\)](#)).

Other convert between effect sizes: [d_to_r\(\)](#), [eta2_to_f2\(\)](#), [odds_to_probs\(\)](#), [oddsratio_to_riskratio\(\)](#)

d_to_r *Convert Between d, r, and Odds Ratio*

Description

Enables a conversion between different indices of effect size, such as standardized difference (Cohen's d), correlation r or (log) odds ratios.

Usage

`d_to_r(d, ...)`

`r_to_d(r, ...)`

`oddsratio_to_d(OR, log = FALSE, ...)`

`logoddsratio_to_d(OR, log = TRUE, ...)`

`d_to_oddsratio(d, log = FALSE, ...)`

`oddsratio_to_r(OR, log = FALSE, ...)`

`logoddsratio_to_r(OR, log = TRUE, ...)`

`r_to_oddsratio(r, log = FALSE, ...)`

Arguments

<code>d</code>	Standardized difference value (Cohen's d).
<code>...</code>	Arguments passed to or from other methods.
<code>r</code>	Correlation coefficient r .
<code>OR</code>	<i>Odds ratio</i> values in vector or data frame.
<code>log</code>	Take in or output the log of the ratio (such as in logistic models).

Details

Conversions between d and OR is done through these formulae:

- $d = \frac{\log(OR) \times \sqrt{3}}{\pi}$
- $\log(OR) = d * \frac{\pi}{\sqrt{(3)}}$

Converting between d and r is done through these formulae:

- $d = \frac{2*r}{\sqrt{1-r^2}}$
- $r = \frac{d}{\sqrt{d^2+4}}$

The conversion from d to r assumes equally sized groups. The resulting r is also called the binomial effect size display (BESD; Rosenthal et al., 1982).

Value

Converted index.

References

- Sánchez-Meca, J., Marín-Martínez, F., & Chacón-Moscoso, S. (2003). Effect-size indices for dichotomized outcomes in meta-analysis. *Psychological methods*, 8(4), 448.
- Borenstein, M., Hedges, L. V., Higgins, J. P. T., & Rothstein, H. R. (2009). Converting among effect sizes. *Introduction to meta-analysis*, 45-49.
- Rosenthal, R., & Rubin, D. B. (1982). A simple, general purpose display of magnitude of experimental effect. *Journal of educational psychology*, 74(2), 166.

See Also

[cohens_d\(\)](#)

Other convert between effect sizes: [diff_to_cles](#), [eta2_to_f2\(\)](#), [odds_to_probs\(\)](#), [oddsratio_to_riskratio\(\)](#)

Examples

```
r_to_d(0.5)
d_to_oddsratio(1.154701)
oddsratio_to_r(8.120534)

d_to_r(1)
r_to_oddsratio(0.4472136, log = TRUE)
oddsratio_to_d(1.813799, log = TRUE)
```

effectsize.BFBayesFactor
Effect Sizes

Description

This function tries to return the best effect-size measure for the provided input model. See details.

Usage

```
## S3 method for class 'BFBayesFactor'
effectsize(model, type = NULL, ci = 0.95, test = NULL, verbose = TRUE, ...)

effectsize(model, ...)

## S3 method for class 'aov'
effectsize(model, type = NULL, ...)

## S3 method for class 'htest'
effectsize(model, type = NULL, verbose = TRUE, ...)
```


Arguments

model	An object of class <code>htest</code> , or a statistical model. See details.
type	The effect size of interest. See details.
ci	Value or vector of probability of the CI (between 0 and 1) to be estimated. Default to .95 (95%).
test	The indices of effect existence to compute. Character (vector) or list with one or more of these options: "p_direction" (or "pd"), "rope", "p_map", "equivalence_test" (or "equitest"), "bayesfactor" (or "bf") or "all" to compute all tests. For each "test", the corresponding <code>bayestestR</code> function is called (e.g. <code>rope()</code> or <code>p_direction()</code>) and its results included in the summary output.
verbose	Toggle warnings and messages on or off.
...	Arguments passed to or from other methods. See details.

Details

- For an object of class `htest`, data is extracted via `insight::get_data()`, and passed to the relevant function according to:
 - A **t-test** depending on type: "cohens_d" (default), "hedges_g", or one of "p_superiority", "u1", "u2", "u3", "overlap".
 - A **Chi-squared tests of independence** or **Fisher's Exact Test**, depending on type: "cramers_v" (default), "tschuprows_t", "phi", "cohens_w", "pearsons_c", "cohens_h", "oddsratio", or "riskratio".
 - A **Chi-squared tests of goodness-of-fit**, depending on type: "fei" (default) "cohens_w", "pearsons_c"
 - A **One-way ANOVA test**, depending on type: "eta" (default), "omega" or "epsilon"-squared, "f", or "f2".
 - A **McNemar test** returns *Cohen's g*.
 - A **Wilcoxon test** depending on type: returns "rank_biserial" correlation (default) or one of "p_superiority", "vda", "u2", "u3", "overlap".
 - A **Kruskal-Wallis test** depending on type: "epsilon" (default) or "eta".
 - A **Friedman test** returns *Kendall's W*. (Where applicable, ci and alternative are taken from the `htest` if not otherwise provided.)
- For an object of class `BFBayesFactor`, using `bayestestR::describe_posterior()`,
 - A **t-test** depending on type: "cohens_d" (default) or one of "p_superiority", "u1", "u2", "u3", "overlap".
 - A **correlation test** returns *r*.
 - A **contingency table test**, depending on type: "cramers_v" (default), "phi", "tschuprows_t", "cohens_w", "pearsons_c", "cohens_h", "oddsratio", or "riskratio".
 - A **proportion test** returns *p*.
- Objects of class `anova`, `aov`, `aovlist` or `afex_aov`, depending on type: "eta" (default), "omega" or "epsilon"-squared, "f", or "f2".
- Other objects are passed to `parameters::standardize_parameters()`.

For statistical models it is recommended to directly use the listed functions, for the full range of options they provide.

Value

A data frame with the effect size (depending on input) and and its CIs (CI_low and CI_high).

See Also

vignette(package = "effectsize")

Examples

```
## Hypothesis Testing
## -----
data("Music_preferences")
Xsq <- chisq.test(Music_preferences)
effectsize(Xsq)
effectsize(Xsq, type = "cohens_w")

Tt <- t.test(1:10, y = c(7:20), alternative = "less")
effectsize(Tt)

Aov <- oneway.test(extra ~ group, data = sleep, var.equal = TRUE)
effectsize(Aov)
effectsize(Aov, type = "omega")

Wt <- wilcox.test(1:10, 7:20, mu = -3, alternative = "less")
effectsize(Wt)
effectsize(Wt, type = "u2")

## Models and Anova Tables
## -----
fit <- lm(mpg ~ factor(cyl) * wt + hp, data = mtcars)
effectsize(fit)

anova_table <- anova(fit)
effectsize(anova_table)
effectsize(anova_table, type = "epsilon")

## Bayesian Hypothesis Testing
## -----

bf_prop <- BayesFactor::proportionBF(3, 7, p = 0.3)
effectsize(bf_prop)

bf_corr <- BayesFactor::correlationBF(attitude$rating, attitude$complaints)
effectsize(bf_corr)

data(RCT_table)
bf_xtab <- BayesFactor::contingencyTableBF(RCT_table, sampleType = "poisson", fixedMargin = "cols")
effectsize(bf_xtab)
effectsize(bf_xtab, type = "oddsratio")
```

```
bf_ttest <- BayesFactor::ttestBF(sleep$extra[sleep$group == 1],
  sleep$extra[sleep$group == 2],
  paired = TRUE, mu = -1
)
effectsize(bf_ttest)
```

effectsize_API

effectsize *API*

Description

Read the *Support functions for model extensions* vignette.

Usage

```
.es_aov_simple(
  aov_table,
  type = c("eta", "omega", "epsilon"),
  partial = TRUE,
  generalized = FALSE,
  include_intercept = FALSE,
  ci = 0.95,
  alternative = "greater",
  verbose = TRUE
)

.es_aov_strata(
  aov_table,
  DV_names,
  type = c("eta", "omega", "epsilon"),
  partial = TRUE,
  generalized = FALSE,
  include_intercept = FALSE,
  ci = 0.95,
  alternative = "greater",
  verbose = TRUE
)

.es_aov_table(
  aov_table,
  type = c("eta", "omega", "epsilon"),
  partial = TRUE,
  generalized = FALSE,
  include_intercept = FALSE,
  ci = 0.95,
  alternative = "greater",
```

```

    verbose = TRUE
  )

```

Arguments

aov_table	Input data frame
type	Which effect size to compute?
partial, generalized, ci, alternative, verbose	See eta_squared() .
include_intercept	Should the intercept ((Intercept)) be included?
DV_names	A character vector with the names of all the predictors, including the grouping variable (e.g., "Subject").

effectsize_CIs	<i>Confidence (Compatibility) Intervals</i>
----------------	---

Description

More information regarding Confidence (Compatibility) Intervals and how they are computed in *effectsize*.

Confidence (Compatibility) Intervals (CIs)

Unless stated otherwise, confidence (compatibility) intervals (CIs) are estimated using the non-centrality parameter method (also called the "pivot method"). This method finds the noncentrality parameter ("*ncp*") of a noncentral *t*, *F*, or χ^2 distribution that places the observed *t*, *F*, or χ^2 test statistic at the desired probability point of the distribution. For example, if the observed *t* statistic is 2.0, with 50 degrees of freedom, for which cumulative noncentral *t* distribution is *t* = 2.0 the .025 quantile (answer: the noncentral *t* distribution with *ncp* = .04)? After estimating these confidence bounds on the *ncp*, they are converted into the effect size metric to obtain a confidence interval for the effect size (Steiger, 2004).

For additional details on estimation and troubleshooting, see [effectsize_CIs](#).

CIs and Significance Tests

"Confidence intervals on measures of effect size convey all the information in a hypothesis test, and more." (Steiger, 2004). Confidence (compatibility) intervals and p values are complementary summaries of parameter uncertainty given the observed data. A dichotomous hypothesis test could be performed with either a CI or a p value. The 100 (1 - α)% confidence interval contains all of the parameter values for which $p > \alpha$ for the current data and model. For example, a 95% confidence interval contains all of the values for which $p > .05$.

Note that a confidence interval including 0 *does not* indicate that the null (no effect) is true. Rather, it suggests that the observed data together with the model and its assumptions combined do not provided clear evidence against a parameter value of 0 (same as with any other value in the interval),

with the level of this evidence defined by the chosen α level (Rafi & Greenland, 2020; Schweder & Hjort, 2016; Xie & Singh, 2013). To infer no effect, additional judgments about what parameter values are "close enough" to 0 to be negligible are needed ("equivalence testing"; Bauer & Kiessler, 1996).

Bootstrapped CIs

Some effect sizes are directionless—they do have a minimum value that would be interpreted as "no effect", but they cannot cross it. For example, a null value of **Kendall's W** is 0, indicating no difference between groups, but it can never have a negative value. Same goes for **U2** and **Overlap**: the null value of U_2 is 0.5, but it can never be smaller than 0.5; an *Overlap* of 1 means "full overlap" (no difference), but it cannot be larger than 1.

When bootstrapping CIs for such effect sizes, the bounds of the CIs will never cross (and often will never cover) the null. Therefore, these CIs should not be used for statistical inference.

One-Sided CIs

Typically, CIs are constructed as two-tailed intervals, with an equal proportion of the cumulative probability distribution above and below the interval. CIs can also be constructed as *one-sided* intervals, giving only a lower bound or upper bound. This is analogous to computing a 1-tailed p value or conducting a 1-tailed hypothesis test.

Significance tests conducted using CIs (whether a value is inside the interval) and using p values (whether $p < \alpha$ for that value) are only guaranteed to agree when both are constructed using the same number of sides/tails.

Most effect sizes are not bounded by zero (e.g., r , d , g), and as such are generally tested using 2-tailed tests and 2-sided CIs.

Some effect sizes are strictly positive—they do have a minimum value, of 0. For example, R^2 , η^2 , and other variance-accounted-for effect sizes, as well as Cramer's V and multiple R , range from 0 to 1. These typically involve F - or χ^2 -statistics and are generally tested using *1-tailed* tests which test whether the estimated effect size is *larger* than the hypothesized null value (e.g., 0). In order for a CI to yield the same significance decision it must then be by a *1-sided* CI, estimating only a lower bound. This is the default CI computed by *effectsize* for these effect sizes, where `alternative = "greater"` is set.

This lower bound interval indicates the smallest effect size that is not significantly different from the observed effect size. That is, it is the minimum effect size compatible with the observed data, background model assumptions, and α level. This type of interval does not indicate a maximum effect size value; anything up to the maximum possible value of the effect size (e.g., 1) is in the interval.

One-sided CIs can also be used to test against a maximum effect size value (e.g., is R^2 significantly smaller than a perfect correlation of 1.0?) can by setting `alternative = "less"`. This estimates a CI with only an *upper* bound; anything from the minimum possible value of the effect size (e.g., 0) up to this upper bound is in the interval.

We can also obtain a 2-sided interval by setting `alternative = "two.sided"`. These intervals can be interpreted in the same way as other 2-sided intervals, such as those for r , d , or g .

An alternative approach to aligning significance tests using CIs and 1-tailed p values that can often be found in the literature is to construct a 2-sided CI at a lower confidence level (e.g., $100(1-2\alpha)\% = 100 - 2*5\% = 90\%$). This estimates the lower bound and upper bound for the above 1-sided intervals simultaneously. These intervals are commonly reported when conducting **equivalence tests**. For example, a 90% 2-sided interval gives the bounds for an equivalence test with $\alpha = .05$. However, be aware that this interval does not give 95% coverage for the underlying effect size parameter value. For that, construct a 95% 2-sided CI.

```
data("hardlyworking")
fit <- lm(salary ~ n_comps, data = hardlyworking)
eta_squared(fit) # default, ci = 0.95, alternative = "greater"
#> For one-way between subjects designs, partial eta squared is equivalent
#> to eta squared. Returning eta squared.
#> # Effect Size for ANOVA
#>
#> Parameter | Eta2 |          95% CI
#> -----
#> n_comps   | 0.19 | [0.14, 1.00]
#>
#> - One-sided CIs: upper bound fixed at [1.00].
eta_squared(fit, alternative = "less") # Test is eta is smaller than some value
#> For one-way between subjects designs, partial eta squared is equivalent
#> to eta squared. Returning eta squared.
#> # Effect Size for ANOVA
#>
#> Parameter | Eta2 |          95% CI
#> -----
#> n_comps   | 0.19 | [0.00, 0.24]
#>
#> - One-sided CIs: lower bound fixed at [0.00].
eta_squared(fit, alternative = "two.sided") # 2-sided bounds for alpha = .05
#> For one-way between subjects designs, partial eta squared is equivalent
#> to eta squared. Returning eta squared.
#> # Effect Size for ANOVA
#>
#> Parameter | Eta2 |          95% CI
#> -----
#> n_comps   | 0.19 | [0.14, 0.25]
eta_squared(fit, ci = 0.9, alternative = "two.sided") # both 1-sided bounds for alpha = .05
#> For one-way between subjects designs, partial eta squared is equivalent
#> to eta squared. Returning eta squared.
#> # Effect Size for ANOVA
#>
#> Parameter | Eta2 |          90% CI
#> -----
#> n_comps   | 0.19 | [0.14, 0.24]
```

CI Does Not Contain the Estimate

For very large sample sizes or effect sizes, the width of the CI can be smaller than the tolerance of the optimizer, resulting in CIs of width 0. This can also result in the estimated CIs excluding the point estimate.

For example:

```
t_to_d(80, df_error = 4555555)
#> d      |      95% CI
#> -----
#> 0.07 | [0.08, 0.08]
```

In these cases, consider an alternative optimizer, or an alternative method for computing CIs, such as the bootstrap.

References

- Bauer, P., & Kieser, M. (1996). A unifying approach for confidence intervals and testing of equivalence and difference. *Biometrika*, 83(4), 934–937. doi:10.1093/biomet/83.4.934
- Rafi, Z., & Greenland, S. (2020). Semantic and cognitive tools to aid statistical science: Replace confidence and significance by compatibility and surprise. *BMC Medical Research Methodology*, 20(1), Article 244. doi:10.1186/s12874020011059
- Schweder, T., & Hjort, N. L. (2016). *Confidence, likelihood, probability: Statistical inference with confidence distributions*. Cambridge University Press. doi:10.1017/CBO9781139046671
- Steiger, J. H. (2004). Beyond the *F* test: Effect size confidence intervals and tests of close fit in the analysis of variance and contrast analysis. *Psychological Methods*, 9(2), 164–182. doi:10.1037/1082989x.9.2.164
- Xie, M., & Singh, K. (2013). Confidence distribution, the frequentist distribution estimator of a parameter: A review. *International Statistical Review*, 81(1), 3–39. doi:10.1111/insr.12000

effectsize_deprecated *Deprecated / Defunct Functions*

Description

Deprecated / Defunct Functions

Usage

```
interpret_d(...)
interpret_g(...)
interpret_delta(...)
interpret_parameters(...)
```

```
normalized_chi(...)  
chisq_to_normalized(...)  
convert_d_to_common_language(...)  
d_to_common_language(...)  
convert_rb_to_common_language(...)  
rb_to_common_language(...)  
common_language(...)
```

Arguments

... Arguments to the deprecated function.

Details

- `interpret_d` is now [interpret_cohens_d](#).
- `interpret_g` is now [interpret_hedges_g](#).
- `interpret_delta` is now [interpret_glass_delta](#).
- `interpret_parameters` for *standardized parameters* was incorrect. Use [interpret_r](#) instead.
- `normalized_chi` is now [fei](#).
- `chisq_to_normalized` is now [chisq_to_fei](#).
- `d_to_cles` and `rb_to_cles` are now one of the available functions for CLES conversion, e.g. [d_to_u1](#).

effectsize_options effectsize_options

Description

Currently, the following global options are supported:

- `es.use_symbols` [logical](#): Should proper symbols be printed (TRUE) instead of transliterated effect size names (FALSE; default).

 equivalence_test.effects_size_table

Test Effect Size for Practical Equivalence to the Null

Description

Perform a **Test for Practical Equivalence** for indices of effect size.

Usage

```
## S3 method for class 'effects_size_table'
equivalence_test(
  x,
  range = "default",
  rule = c("classic", "cet", "bayes"),
  ...
)
```

Arguments

x	An effect size table, such as returned by <code>cohens_d()</code> , <code>eta_squared()</code> , <code>F_to_r()</code> , etc.
range	The range of practical equivalence of an effect. For one-sided CIs, a single value can be provided for the lower / upper bound to test against (but see more details below). For two-sided CIs, a single value is duplicated to <code>c(-range, range)</code> . If "default", will be set to <code>[-.1, .1]</code> .
rule	How should acceptance and rejection be decided? See details.
...	Arguments passed to or from other methods.

Details

The CIs used in the equivalence test are the ones in the provided effect size table. For results equivalent (ha!) to those that can be obtained using the TOST approach (e.g., Lakens, 2017), appropriate CIs should be extracted using the function used to make the effect size table (`cohens_d`, `eta_squared`, `F_to_r`, etc), with `alternative = "two.sided"`. See examples.

The Different Rules:

- "classic" - **the classic method:**
 - If the CI is completely within the ROPE - *Accept H0*
 - Else, if the CI does not contain 0 - *Reject H0*
 - Else - *Undecided*
- "cet" - **conditional equivalence testing:**
 - If the CI does not contain 0 - *Reject H0*
 - Else, If the CI is completely within the ROPE - *Accept H0*
 - Else - *Undecided*

- "bayes" - **The Bayesian approach**, as put forth by Kruschke:
 - If the CI does is completely outside the ROPE - *Reject H0*
 - Else, If the CI is completely within the ROPE - *Accept H0*
 - Else - *Undecided*

Value

A data frame with the results of the equivalence test.

References

- Campbell, H., & Gustafson, P. (2018). Conditional equivalence testing: An alternative remedy for publication bias. PLOS ONE, 13(4), e0195145. <https://doi.org/10.1371/journal.pone.0195145>
- Kruschke, J. K. (2014). Doing Bayesian data analysis: A tutorial with R, JAGS, and Stan. Academic Press
- Kruschke, J. K. (2018). Rejecting or accepting parameter values in Bayesian estimation. Advances in Methods and Practices in Psychological Science, 1(2), 270-280. doi: 10.1177/2515245918771304
- Lakens, D. (2017). Equivalence Tests: A Practical Primer for t Tests, Correlations, and Meta-Analyses. Social Psychological and Personality Science, 8(4), 355–362. <https://doi.org/10.1177/1948550617697177>

See Also

For more details, see `bayestestR::equivalence_test()`.

Examples

```
data("hardlyworking")
model <- aov(salary ~ age + factor(n_comps) * cut(seniority, 3), data = hardlyworking)
es <- eta_squared(model, ci = 0.9, alternative = "two.sided")
equivalence_test(es, range = 0.15) # TOST

data("RCT_table")
OR <- oddsratio(RCT_table, alternative = "greater")
equivalence_test(OR, range = 1)

ds <- t_to_d(
  t = c(0.45, -0.65, 7, -2.2, 2.25),
  df_error = c(675, 525, 2000, 900, 1875),
  ci = 0.9, alternative = "two.sided" # TOST
)
# Can also plot
if (require(see)) plot(equivalence_test(ds, range = 0.2))
if (require(see)) plot(equivalence_test(ds, range = 0.2, rule = "cet"))
if (require(see)) plot(equivalence_test(ds, range = 0.2, rule = "bayes"))
```

Description

Convert Between ANOVA Effect Sizes

Usage

eta2_to_f2(es)

eta2_to_f(es)

f2_to_eta2(f2)

f_to_eta2(f)

Arguments

es	Any measure of variance explained such as Eta-, Epsilon-, Omega-, or R-Squared, partial or otherwise. See details.
f, f2	Cohen's f or f -squared.

Details

Any measure of variance explained can be converted to a corresponding Cohen's f via:

$$f^2 = \frac{\eta^2}{1 - \eta^2}$$

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

If a partial Eta-Squared is used, the resulting Cohen's f is a partial-Cohen's f ; If a less biased estimate of variance explained is used (such as Epsilon- or Omega-Squared), the resulting Cohen's f is likewise a less biased estimate of Cohen's f .

References

- Cohen, J. (1988). Statistical power analysis for the behavioral sciences (2nd Ed.). New York: Routledge.
- Steiger, J. H. (2004). Beyond the F test: Effect size confidence intervals and tests of close fit in the analysis of variance and contrast analysis. *Psychological Methods*, 9, 164-182.

See Also

[eta_squared\(\)](#) for more details.

Other convert between effect sizes: [d_to_r\(\)](#), [diff_to_cles](#), [odds_to_probs\(\)](#), [oddsratio_to_riskratio\(\)](#)

 eta_squared

 η^2 and Other Effect Size for ANOVA

Description

Functions to compute effect size measures for ANOVAs, such as Eta- (η), Omega- (ω) and Epsilon- (ϵ) squared, and Cohen's f (or their partialled versions) for ANOVA tables. These indices represent an estimate of how much variance in the response variables is accounted for by the explanatory variable(s).

When passing models, effect sizes are computed using the sums of squares obtained from `anova(model)` which might not always be appropriate. See details.

Usage

```
eta_squared(
  model,
  partial = TRUE,
  generalized = FALSE,
  ci = 0.95,
  alternative = "greater",
  verbose = TRUE,
  ...
)
```

```
omega_squared(
  model,
  partial = TRUE,
  ci = 0.95,
  alternative = "greater",
  verbose = TRUE,
  ...
)
```

```
epsilon_squared(
  model,
  partial = TRUE,
  ci = 0.95,
  alternative = "greater",
  verbose = TRUE,
  ...
)
```

```

cohens_f(
  model,
  partial = TRUE,
  squared = FALSE,
  model2 = NULL,
  ci = 0.95,
  alternative = "greater",
  verbose = TRUE,
  ...
)

cohens_f_squared(
  model,
  partial = TRUE,
  squared = TRUE,
  model2 = NULL,
  ci = 0.95,
  alternative = "greater",
  verbose = TRUE,
  ...
)

eta_squared_posterior(
  model,
  partial = TRUE,
  generalized = FALSE,
  ss_function = stats::anova,
  draws = 500,
  verbose = TRUE,
  ...
)

```

Arguments

model	A model, ANOVA object, or the result of <code>parameters::model_parameters</code> .
partial	If TRUE, return partial indices.
generalized	If TRUE, returns generalized Eta Squared, assuming all variables are manipulated. Can also be a character vector of observed (non-manipulated) variables, in which case generalized Eta Squared is calculated taking these observed variables into account. For <code>afex_aov</code> model, when <code>generalized = TRUE</code> , the observed variables are extracted automatically from the fitted model, if they were provided then.
ci	Confidence Interval (CI) level
alternative	a character string specifying the alternative hypothesis; Controls the type of CI returned: "greater" (default) or "less" (one-sided CI), or "two.sided" (default, two-sided CI). Partial matching is allowed (e.g., "g", "l", "two"...). See <i>One-Sided CIs</i> in effectsize_CIs .

verbose	Toggle warnings and messages on or off.
...	Arguments passed to or from other methods. <ul style="list-style-type: none"> • Can be <code>include_intercept = TRUE</code> to include the effect size for the intercept (when it is included in the ANOVA table). • For Bayesian models, arguments passed to <code>ss_function</code>.
squared	Return Cohen's f or Cohen's f -squared?
model2	Optional second model for Cohen's f (f -squared). If specified, returns the effect size for R-squared-change between the two models.
ss_function	For Bayesian models, the function used to extract sum-of-squares. Uses <code>anova()</code> by default, but can also be <code>car::Anova()</code> for simple linear models.
draws	For Bayesian models, an integer indicating the number of draws from the posterior predictive distribution to return. Larger numbers take longer to run, but provide estimates that are more stable.

Details

For `aov`, `aovlist` and `afex_aov` models, and for `anova` objects that provide Sums-of-Squares, the effect sizes are computed directly using Sums-of-Squares (for `mlm` / `maov` models, effect sizes are computed for each response separately). For all other model, effect sizes are approximated via test statistic conversion of the omnibus F statistic provided by the appropriate `anova()` method (see `F_to_eta2()` for more details.)

Type of Sums of Squares:

The sums of squares (or F statistics) used for the computation of the effect sizes is based on those returned by `anova(model)` (whatever those may be - for `aov` and `aovlist` these are *type-1* sums of squares; for `lmerMod` (and `lmerModLmerTest`) these are *type-3* sums of squares). Make sure these are the sums of squares you are interested in; You might want to pass the result of `car::Anova(mode, type = 2)` or `type = 3` instead of the model itself, or use the `afex` package to fit ANOVA models.

For type 3 sum of squares, it is generally recommended to fit models with `contr.sum` *factor weights* and *centered covariates*, for sensible results. See examples and the `afex` package.

Un-Biased Estimate of Eta:

Both **Omega** and **Epsilon** are unbiased estimators of the population's **Eta**, which is especially important is small samples. But which to choose?

Though Omega is the more popular choice (Albers and Lakens, 2018), Epsilon is analogous to adjusted R^2 (Allen, 2017, p. 382), and has been found to be less biased (Carroll & Nordholm, 1975).

Cohen's f :

Cohen's f can take on values between zero, when the population means are all equal, and an indefinitely large number as standard deviation of means increases relative to the average standard deviation within each group.

When comparing two models in a sequential regression analysis, Cohen's f for R-square change

is the ratio between the increase in R-square and the percent of unexplained variance.

Cohen has suggested that the values of 0.10, 0.25, and 0.40 represent small, medium, and large effect sizes, respectively.

Eta Squared from Posterior Predictive Distribution:

For Bayesian models (fit with `brms` or `rstanarm`), `eta_squared_posterior()` simulates data from the posterior predictive distribution (ppd) and for each simulation the Eta Squared is computed for the model's fixed effects. This means that the returned values are the population level effect size as implied by the posterior model (and not the effect size in the sample data). See [rstantools::posterior_predict\(\)](#) for more info.

Value

A data frame with the effect size(s) between 0-1 (Eta2, Epsilon2, Omega2, Cohens_f or Cohens_f2, possibly with the `partial` or `generalized` suffix), and their CIs (`CI_low` and `CI_high`).

For `eta_squared_posterior()`, a data frame containing the ppd of the Eta squared for each fixed effect, which can then be passed to [bayestestR::describe_posterior\(\)](#) for summary stats.

A data frame containing the effect size values and their confidence intervals.

Confidence (Compatibility) Intervals (CIs)

Unless stated otherwise, confidence (compatibility) intervals (CIs) are estimated using the non-centrality parameter method (also called the "pivot method"). This method finds the noncentrality parameter ("*n*cp") of a noncentral *t*, *F*, or χ^2 distribution that places the observed *t*, *F*, or χ^2 test statistic at the desired probability point of the distribution. For example, if the observed *t* statistic is 2.0, with 50 degrees of freedom, for which cumulative noncentral *t* distribution is *t* = 2.0 the .025 quantile (answer: the noncentral *t* distribution with *n*cp = .04)? After estimating these confidence bounds on the *n*cp, they are converted into the effect size metric to obtain a confidence interval for the effect size (Steiger, 2004).

For additional details on estimation and troubleshooting, see [effectsize_CIs](#).

CIs and Significance Tests

"Confidence intervals on measures of effect size convey all the information in a hypothesis test, and more." (Steiger, 2004). Confidence (compatibility) intervals and p values are complementary summaries of parameter uncertainty given the observed data. A dichotomous hypothesis test could be performed with either a CI or a p value. The 100 (1 - α)% confidence interval contains all of the parameter values for which $p > \alpha$ for the current data and model. For example, a 95% confidence interval contains all of the values for which $p > .05$.

Note that a confidence interval including 0 *does not* indicate that the null (no effect) is true. Rather, it suggests that the observed data together with the model and its assumptions combined do not provided clear evidence against a parameter value of 0 (same as with any other value in the interval), with the level of this evidence defined by the chosen α level (Rafi & Greenland, 2020; Schweder & Hjort, 2016; Xie & Singh, 2013). To infer no effect, additional judgments about what parameter

values are "close enough" to 0 to be negligible are needed ("equivalence testing"; Bauer & Kiessler, 1996).

References

- Albers, C., and Lakens, D. (2018). When power analyses based on pilot data are biased: Inaccurate effect size estimators and follow-up bias. *Journal of experimental social psychology*, 74, 187-195.
- Allen, R. (2017). *Statistics and Experimental Design for Psychologists: A Model Comparison Approach*. World Scientific Publishing Company.
- Carroll, R. M., & Nordholm, L. A. (1975). Sampling Characteristics of Kelley's epsilon and Hays' omega. *Educational and Psychological Measurement*, 35(3), 541-554.
- Kelley, T. (1935) An unbiased correlation ratio measure. *Proceedings of the National Academy of Sciences*. 21(9). 554-559.
- Olejnik, S., & Algina, J. (2003). Generalized eta and omega squared statistics: measures of effect size for some common research designs. *Psychological methods*, 8(4), 434.
- Steiger, J. H. (2004). Beyond the F test: Effect size confidence intervals and tests of close fit in the analysis of variance and contrast analysis. *Psychological Methods*, 9, 164-182.

See Also

[F_to_eta2\(\)](#)

Other effect sizes for ANOVAs: [rank_epsilon_squared\(\)](#)

Examples

```
data(mtcars)
mtcars$am_f <- factor(mtcars$am)
mtcars$cyl_f <- factor(mtcars$cyl)

model <- aov(mpg ~ am_f * cyl_f, data = mtcars)

(eta2 <- eta_squared(model))

# More types:
eta_squared(model, partial = FALSE)
eta_squared(model, generalized = "cyl_f")
omega_squared(model)
epsilon_squared(model)
cohens_f(model)

model0 <- aov(mpg ~ am_f + cyl_f, data = mtcars) # no interaction
cohens_f_squared(model0, model2 = model)

## Interpretation of effect sizes
## -----

interpret_omega_squared(0.10, rules = "field2013")
interpret_eta_squared(0.10, rules = "cohen1992")
```



```

interpret_epsilon_squared(0.10, rules = "cohen1992")

interpret(eta2, rules = "cohen1992")

plot(eta2) # Requires the {see} package

# Recommended: Type-2 or -3 effect sizes + effects coding
# -----
contrasts(mtcars$am_f) <- contr.sum
contrasts(mtcars$cyl_f) <- contr.sum

model <- aov(mpg ~ am_f * cyl_f, data = mtcars)
model_anova <- car::Anova(model, type = 3)

epsilon_squared(model_anova)

# afex takes care of both type-3 effects and effects coding:
data(obk.long, package = "afex")
model <- afex::aov_car(value ~ treatment * gender + Error(id / (phase)),
  data = obk.long, observed = "gender"
)

omega_squared(model)
eta_squared(model, generalized = TRUE) # observed vars are pulled from the afex model.

## Approx. effect sizes for mixed models
## -----
model <- lme4::lmer(mpg ~ am_f * cyl_f + (1 | vs), data = mtcars)
omega_squared(model)

## Bayesian Models (PPD)
## -----
fit_bayes <- rstanarm::stan_glm(
  mpg ~ factor(cyl) * wt + qsec,
  data = mtcars, family = gaussian(),
  refresh = 0
)

es <- eta_squared_posterior(fit_bayes,
  verbose = FALSE,
  ss_function = car::Anova, type = 3
)
bayestestR::describe_posterior(es, test = NULL)

# compare to:
fit_freq <- lm(mpg ~ factor(cyl) * wt + qsec,
  data = mtcars
)

```

```
)  
aov_table <- car::Anova(fit_freq, type = 3)  
eta_squared(aov_table)
```

food_class	<i>Classification of Foods</i>
------------	--------------------------------

Description

Fictional data.

Format

A 2-by-3 table.

```
data("food_class")  
food_class  
#>      Soy Milk Meat  
#> Vegan    47    0    0  
#> Not-Vegan  0   12   21
```

See Also

Other effect size datasets: [Music_preferences2](#), [Music_preferences](#), [RCT_table](#), [Smoking_FASD](#), [hardlyworking](#), [screening_test](#)

format_standardize	<i>Format a Standardized Vector</i>
--------------------	-------------------------------------

Description

Transform a standardized vector into character, e.g., `c("-1 SD", "Mean", "+1 SD")`.

Usage

```
format_standardize(  
  x,  
  reference = x,  
  robust = FALSE,  
  digits = 1,  
  protect_integers = TRUE,  
  ...  
)
```

Arguments

x	A standardized numeric vector.
reference	The reference vector from which to compute the mean and SD.
robust	Logical, if TRUE, centering is done by subtracting the median from the variables and dividing it by the median absolute deviation (MAD). If FALSE, variables are standardized by subtracting the mean and dividing it by the standard deviation (SD).
digits	Number of digits for rounding or significant figures. May also be "signif" to return significant figures or "scientific" to return scientific notation. Control the number of digits by adding the value as suffix, e.g. digits = "scientific4" to have scientific notation with 4 decimal places, or digits = "signif5" for 5 significant figures (see also signif()).
protect_integers	Should integers be kept as integers (i.e., without decimals)?
...	Other arguments to pass to insight::format_value() such as digits, etc.

Examples

```
format_standardize(c(-1, 0, 1))
format_standardize(c(-1, 0, 1, 2), reference = rnorm(1000))
format_standardize(c(-1, 0, 1, 2), reference = rnorm(1000), robust = TRUE)

format_standardize(standardize(mtcars$wt), digits = 1)
format_standardize(standardize(mtcars$wt, robust = TRUE), digits = 1)
```

F_to_eta2	<i>Convert F and t Statistics to partial-η^2 and Other ANOVA Effect Sizes</i>
-----------	--

Description

These functions are convenience functions to convert F and t test statistics to **partial** Eta- (η), Omega- (ω) Epsilon- (ϵ) squared (an alias for the adjusted Eta squared) and Cohen's f. These are useful in cases where the various Sum of Squares and Mean Squares are not easily available or their computation is not straightforward (e.g., in liner mixed models, contrasts, etc.). For test statistics derived from lm and aov models, these functions give exact results. For all other cases, they return close approximations.

See [Effect Size from Test Statistics vignette](#).

Usage

```
F_to_eta2(f, df, df_error, ci = 0.95, alternative = "greater", ...)
t_to_eta2(t, df_error, ci = 0.95, alternative = "greater", ...)
F_to_epsilon2(f, df, df_error, ci = 0.95, alternative = "greater", ...)
```

```

t_to_epsilon2(t, df_error, ci = 0.95, alternative = "greater", ...)
F_to_eta2_adj(f, df, df_error, ci = 0.95, alternative = "greater", ...)
t_to_eta2_adj(t, df_error, ci = 0.95, alternative = "greater", ...)
F_to_omega2(f, df, df_error, ci = 0.95, alternative = "greater", ...)
t_to_omega2(t, df_error, ci = 0.95, alternative = "greater", ...)

F_to_f(
  f,
  df,
  df_error,
  squared = FALSE,
  ci = 0.95,
  alternative = "greater",
  ...
)

t_to_f(t, df_error, squared = FALSE, ci = 0.95, alternative = "greater", ...)

F_to_f2(
  f,
  df,
  df_error,
  squared = TRUE,
  ci = 0.95,
  alternative = "greater",
  ...
)

t_to_f2(t, df_error, squared = TRUE, ci = 0.95, alternative = "greater", ...)

```

Arguments

df, df_error	Degrees of freedom of numerator or of the error estimate (i.e., the residuals).
ci	Confidence Interval (CI) level
alternative	a character string specifying the alternative hypothesis; Controls the type of CI returned: "greater" (default) or "less" (one-sided CI), or "two.sided" (default, two-sided CI). Partial matching is allowed (e.g., "g", "l", "two"...). See <i>One-Sided CIs</i> in effectsize_CIs .
...	Arguments passed to or from other methods.
t, f	The t or the F statistics.
squared	Return Cohen's <i>f</i> or Cohen's <i>f</i> -squared?

Details

These functions use the following formulae:

$$\eta_p^2 = \frac{F \times df_{num}}{F \times df_{num} + df_{den}}$$

$$\epsilon_p^2 = \frac{(F - 1) \times df_{num}}{F \times df_{num} + df_{den}}$$

$$\omega_p^2 = \frac{(F - 1) \times df_{num}}{F \times df_{num} + df_{den} + 1}$$

$$f_p = \sqrt{\frac{\eta_p^2}{1 - \eta_p^2}}$$

For t , the conversion is based on the equality of $t^2 = F$ when $df_{num} = 1$.

Choosing an Un-Biased Estimate:

Both Omega and Epsilon are unbiased estimators of the population Eta. But which to choose? Though Omega is the more popular choice, it should be noted that:

1. The formula given above for Omega is only an approximation for complex designs.
2. Epsilon has been found to be less biased (Carroll & Nordholm, 1975).

Value

A data frame with the effect size(s) between 0-1 (Eta2_partial, Epsilon2_partial, Omega2_partial, Cohens_f_partial or Cohens_f2_partial), and their CIs (CI_low and CI_high).

Confidence (Compatibility) Intervals (CIs)

Unless stated otherwise, confidence (compatibility) intervals (CIs) are estimated using the non-centrality parameter method (also called the "pivot method"). This method finds the noncentrality parameter (" ncp ") of a noncentral t , F , or χ^2 distribution that places the observed t , F , or χ^2 test statistic at the desired probability point of the distribution. For example, if the observed t statistic is 2.0, with 50 degrees of freedom, for which cumulative noncentral t distribution is $t = 2.0$ the .025 quantile (answer: the noncentral t distribution with $ncp = .04$)? After estimating these confidence bounds on the ncp , they are converted into the effect size metric to obtain a confidence interval for the effect size (Steiger, 2004).

For additional details on estimation and troubleshooting, see [effectsize_CIs](#).

CI and Significance Tests

"Confidence intervals on measures of effect size convey all the information in a hypothesis test, and more." (Steiger, 2004). Confidence (compatibility) intervals and p values are complementary summaries of parameter uncertainty given the observed data. A dichotomous hypothesis test could be performed with either a CI or a p value. The 100 (1 - α)% confidence interval contains all of the parameter values for which $p > \alpha$ for the current data and model. For example, a 95% confidence interval contains all of the values for which $p > .05$.

Note that a confidence interval including 0 *does not* indicate that the null (no effect) is true. Rather, it suggests that the observed data together with the model and its assumptions combined do not provided clear evidence against a parameter value of 0 (same as with any other value in the interval), with the level of this evidence defined by the chosen α level (Rafi & Greenland, 2020; Schweder & Hjort, 2016; Xie & Singh, 2013). To infer no effect, additional judgments about what parameter values are "close enough" to 0 to be negligible are needed ("equivalence testing"; Bauer & Kieser, 1996).

Note

Adjusted (partial) Eta-squared is an alias for (partial) Epsilon-squared.

References

- Albers, C., & Lakens, D. (2018). When power analyses based on pilot data are biased: Inaccurate effect size estimators and follow-up bias. *Journal of experimental social psychology*, 74, 187-195. doi:10.31234/osf.io/b7z4q
- Carroll, R. M., & Nordholm, L. A. (1975). Sampling Characteristics of Kelley's epsilon and Hays' omega. *Educational and Psychological Measurement*, 35(3), 541-554.
- Cumming, G., & Finch, S. (2001). A primer on the understanding, use, and calculation of confidence intervals that are based on central and noncentral distributions. *Educational and Psychological Measurement*, 61(4), 532-574.
- Friedman, H. (1982). Simplified determinations of statistical power, magnitude of effect and research sample sizes. *Educational and Psychological Measurement*, 42(2), 521-526. doi:10.1177/001316448204200214
- Mordkoff, J. T. (2019). A Simple Method for Removing Bias From a Popular Measure of Standardized Effect Size: Adjusted Partial Eta Squared. *Advances in Methods and Practices in Psychological Science*, 2(3), 228-232. doi:10.1177/2515245919855053
- Morey, R. D., Hoekstra, R., Rouder, J. N., Lee, M. D., & Wagenmakers, E. J. (2016). The fallacy of placing confidence in confidence intervals. *Psychonomic bulletin & review*, 23(1), 103-123.
- Steiger, J. H. (2004). Beyond the F test: Effect size confidence intervals and tests of close fit in the analysis of variance and contrast analysis. *Psychological Methods*, 9, 164-182.

See Also

[eta_squared\(\)](#) for more details.

Other effect size from test statistic: [chisq_to_phi\(\)](#), [t_to_d\(\)](#)

Examples

```

mod <- aov(mpg ~ factor(cyl) * factor(am), mtcars)
anova(mod)
(etas <- F_to_eta2(
  f = c(44.85, 3.99, 1.38),
  df = c(2, 1, 2),
  df_error = 26
))

if (require(see)) plot(etas)

# Compare to:
eta_squared(mod)

fit <- lmerTest::lmer(extra ~ group + (1 | ID), sleep)
# anova(fit)
# #> Type III Analysis of Variance Table with Satterthwaite's method
# #>      Sum Sq Mean Sq NumDF DenDF F value  Pr(>F)
# #> group 12.482   12.482     1     9  16.501 0.002833 **
# #> ---
# #> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

F_to_eta2(16.501, 1, 9)
F_to_omega2(16.501, 1, 9)
F_to_epsilon2(16.501, 1, 9)
F_to_f(16.501, 1, 9)

#' @examplesIf require(emmeans)
if (require(emmeans)) {
  ## Use with emmeans based contrasts
  ## -----
  warp.lm <- lm(breaks ~ wool * tension, data = warpbreaks)

  jt <- emmeans::joint_tests(warp.lm, by = "wool")
  F_to_eta2(jt$F.ratio, jt$df1, jt$df2)
}

```

Description

A sample (simulated) dataset, used in tests and some examples.

Format

A data frame with 500 rows and 5 variables:

salary Salary, in Shmekels

xtra_hours Number of overtime hours (on average, per week)

n_comps Number of compliments given to the boss (observed over the last week)

age Age in years

seniority How many years with the company

is_senior Has this person been working here for more than 4 years?

```
data("hardlyworking")
head(hardlyworking, n = 5)
#>   salary xtra_hours n_comps age seniority is_senior
#> 1 19744.65     4.16     1 32      3     FALSE
#> 2 11301.95     1.62     0 34      3     FALSE
#> 3 20635.62     1.19     3 33      5      TRUE
#> 4 23047.16     7.19     1 35      3     FALSE
#> 5 27342.15    11.26     0 33      4     FALSE
```

See Also

Other effect size datasets: [Music_preferences2](#), [Music_preferences](#), [RCT_table](#), [Smoking_FASD](#), [food_class](#), [screening_test](#)

interpret

Generic Function for Interpretation

Description

Interpret a value based on a set of rules. See [rules\(\)](#).

Usage

```
interpret(x, ...)

## S3 method for class 'numeric'
interpret(x, rules, name = attr(rules, "rule_name"), ...)

## S3 method for class 'effectsize_table'
interpret(x, rules, ...)
```


Arguments

x	Vector of value break points (edges defining categories), or a data frame of class <code>effectsize_table</code> .
...	Currently not used.
rules	Set of <code>rules()</code> . When x is a data frame, can be a name of an established set of rules.
name	Name of the set of rules (will be printed).

Value

- For numeric input: A character vector of interpretations.
- For data frames: the x input with an additional `Interpretation` column.

See Also

[rules\(\)](#)

Examples

```
rules_grid <- rules(c(0.01, 0.05), c("very significant", "significant", "not significant"))
interpret(0.001, rules_grid)
interpret(0.021, rules_grid)
interpret(0.08, rules_grid)
interpret(c(0.01, 0.005, 0.08), rules_grid)

interpret(c(0.35, 0.15), c("small" = 0.2, "large" = 0.4), name = "Cohen's Rules")
interpret(c(0.35, 0.15), rules(c(0.2, 0.4), c("small", "medium", "large")))

# -----
d <- cohens_d(mpg ~ am, data = mtcars)
interpret(d, rules = "cohen1988")

d <- glass_delta(mpg ~ am, data = mtcars)
interpret(d, rules = "gignac2016")

interpret(d, rules = rules(1, c("tiny", "yeah okay")))

m <- lm(formula = wt ~ am * cyl, data = mtcars)
eta2 <- eta_squared(m)
interpret(eta2, rules = "field2013")

X <- chisq.test(mtcars$am, mtcars$cyl == 8)
interpret(oddsratio(X), rules = "chen2010")
interpret(cramers_v(X), "lovakov2021")
```

interpret_bf	<i>Interpret Bayes Factor (BF)</i>
--------------	------------------------------------

Description

Interpret Bayes Factor (BF)

Usage

```
interpret_bf(
  bf,
  rules = "jeffreys1961",
  log = FALSE,
  include_value = FALSE,
  protect_ratio = TRUE,
  exact = TRUE
)
```

Arguments

bf	Value or vector of Bayes factor (BF) values.
rules	Can be "jeffreys1961" (default), "raftery1995" or custom set of <code>rules()</code> (for the <i>absolute magnitude</i> of evidence).
log	Is the bf value $\log(\text{bf})$?
include_value	Include the value in the output.
protect_ratio	Should values smaller than 1 be represented as ratios?
exact	Should very large or very small values be reported with a scientific format (e.g., 4.24e5), or as truncated values (as "> 1000" and "< 1/1000").

Details

Argument names can be partially matched.

Rules

Rules apply to BF as ratios, so BF of 10 is as extreme as a BF of 0.1 (1/10).

- Jeffreys (1961) ("jeffreys1961"; default)
 - **BF = 1** - No evidence
 - **1 < BF <= 3** - Anecdotal
 - **3 < BF <= 10** - Moderate
 - **10 < BF <= 30** - Strong
 - **30 < BF <= 100** - Very strong
 - **BF > 100** - Extreme.
- Raftery (1995) ("raftery1995")

- **BF = 1** - No evidence
- **1 < BF <= 3** - Weak
- **3 < BF <= 20** - Positive
- **20 < BF <= 150** - Strong
- **BF > 150** - Very strong

References

- Jeffreys, H. (1961), Theory of Probability, 3rd ed., Oxford University Press, Oxford.
- Raftery, A. E. (1995). Bayesian model selection in social research. Sociological methodology, 25, 111-164.
- Jarosz, A. F., & Wiley, J. (2014). What are the odds? A practical guide to computing and reporting Bayes factors. The Journal of Problem Solving, 7(1), 2.

Examples

```
interpret_bf(1)
interpret_bf(c(5, 2))
```

interpret_cohens_d *Interpret Standardized Differences*

Description

Interpretation of standardized differences using different sets of rules of thumb.

Usage

```
interpret_cohens_d(d, rules = "cohen1988", ...)
interpret_hedges_g(g, rules = "cohen1988")
interpret_glass_delta(delta, rules = "cohen1988")
```

Arguments

d, g, delta	Value or vector of effect size values.
rules	Can be "cohen1988" (default), "gignac2016", "sawilowsky2009", "lovakov2021" or a custom set of <code>rules()</code> .
...	Not directly used.

Rules

Rules apply to equally to positive and negative d (i.e., they are given as absolute values).

- Cohen (1988) ("cohen1988"; default)
 - $d < 0.2$ - Very small
 - $0.2 \leq d < 0.5$ - Small
 - $0.5 \leq d < 0.8$ - Medium
 - $d \geq 0.8$ - Large
- Sawilowsky (2009) ("sawilowsky2009")
 - $d < 0.1$ - Tiny
 - $0.1 \leq d < 0.2$ - Very small
 - $0.2 \leq d < 0.5$ - Small
 - $0.5 \leq d < 0.8$ - Medium
 - $0.8 \leq d < 1.2$ - Large
 - $1.2 \leq d < 2$ - Very large
 - $d \geq 2$ - Huge
- Lovakov & Agadullina (2021) ("lovakov2021")
 - $d < 0.15$ - Very small
 - $0.15 \leq d < 0.36$ - Small
 - $0.36 \leq d < 0.65$ - Medium
 - $d \geq 0.65$ - Large
- Gignac & Szodorai (2016) ("gignac2016", based on the `d_to_r()` conversion, see [interpret_r\(\)](#))
 - $d < 0.2$ - Very small
 - $0.2 \leq d < 0.41$ - Small
 - $0.41 \leq d < 0.63$ - Moderate
 - $d \geq 0.63$ - Large

References

- Lovakov, A., & Agadullina, E. R. (2021). Empirically Derived Guidelines for Effect Size Interpretation in Social Psychology. *European Journal of Social Psychology*.
- Gignac, G. E., & Szodorai, E. T. (2016). Effect size guidelines for individual differences researchers. *Personality and individual differences*, 102, 74-78.
- Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd Ed.). New York: Routledge.
- Sawilowsky, S. S. (2009). New effect size rules of thumb.

Examples

```
interpret_cohens_d(.02)
interpret_cohens_d(c(.5, .02))
interpret_cohens_d(.3, rules = "lovakov2021")
```

interpret_cohens_g *Interpret Cohen's g*

Description

Interpret Cohen's g

Usage

```
interpret_cohens_g(g, rules = "cohen1988", ...)
```

Arguments

g	Value or vector of effect size values.
rules	Can be "cohen1988" (default) or a custom set of <code>rules()</code> .
...	Not directly used.

Rules

Rules apply to equally to positive and negative g (i.e., they are given as absolute values).

- Cohen (1988) ("cohen1988"; default)
 - $d < 0.05$ - Very small
 - $0.05 \leq d < 0.15$ - Small
 - $0.15 \leq d < 0.25$ - Medium
 - $d \geq 0.25$ - Large

Note

"Since g is so transparently clear a unit, it is expected that workers in any given substantive area of the behavioral sciences will very frequently be able to set relevant [effect size] values without the proposed conventions, or set up conventions of their own which are suited to their area of inquiry."
- Cohen, 1988, page 147.

References

- Cohen, J. (1988). Statistical power analysis for the behavioral sciences (2nd Ed.). New York: Routledge.

Examples

```
interpret_cohens_g(.02)  
interpret_cohens_g(c(.3, .15))
```

interpret_direction *Interpret Direction*

Description

Interpret Direction

Usage

```
interpret_direction(x)
```

Arguments

x Numeric value.

Examples

```
interpret_direction(.02)
interpret_direction(c(.5, -.02))
```

interpret_ess *Interpret Bayesian Diagnostic Indices*

Description

Interpretation of Bayesian diagnostic indices, such as Effective Sample Size (ESS) and Rhat.

Usage

```
interpret_ess(ess, rules = "burkner2017")
interpret_rhat(rhat, rules = "vehtari2019")
```

Arguments

ess Value or vector of Effective Sample Size (ESS) values.
rules A character string (see *Rules*) or a custom set of `rules()`.
rhat Value or vector of Rhat values.

Rules

ESS:

- Bürkner, P. C. (2017) ("burkner2017"; default)
 - **ESS < 1000** - Insufficient
 - **ESS >= 1000** - Sufficient

Rhat:

- Vehtari et al. (2019) ("vehtari2019"; default)
 - **Rhat < 1.01** - Converged
 - **Rhat >= 1.01** - Failed
- Gelman & Rubin (1992) ("gelman1992")
 - **Rhat < 1.1** - Converged
 - **Rhat >= 1.1** - Failed

References

- Bürkner, P. C. (2017). brms: An R package for Bayesian multilevel models using Stan. *Journal of Statistical Software*, 80(1), 1-28.
- Gelman, A., & Rubin, D. B. (1992). Inference from iterative simulation using multiple sequences. *Statistical science*, 7(4), 457-472.
- Vehtari, A., Gelman, A., Simpson, D., Carpenter, B., & Bürkner, P. C. (2019). Rank-normalization, folding, and localization: An improved Rhat for assessing convergence of MCMC. arXiv preprint arXiv:1903.08008.

Examples

```
interpret_ess(1001)
interpret_ess(c(852, 1200))

interpret_rhat(1.00)
interpret_rhat(c(1.5, 0.9))
```

 interpret_gfi

Interpret of CFA / SEM Indices of Goodness of Fit

Description

Interpretation of indices of fit found in confirmatory analysis or structural equation modelling, such as RMSEA, CFI, NFI, IFI, etc.

Usage

```

interpret_gfi(x, rules = "default")

interpret_agfi(x, rules = "default")

interpret_nfi(x, rules = "byrne1994")

interpret_nnfi(x, rules = "byrne1994")

interpret_cfi(x, rules = "default")

interpret_rmsea(x, rules = "default")

interpret_srmr(x, rules = "default")

interpret_rfi(x, rules = "default")

interpret_ifi(x, rules = "default")

interpret_pnfi(x, rules = "default")

## S3 method for class 'lavaan'
interpret(x, ...)

## S3 method for class 'performance_lavaan'
interpret(x, ...)

```

Arguments

x	vector of values, or an object of class lavaan.
rules	Can be "default" or custom set of rules() .
...	Currently not used.

Details**Indices of fit:**

- **Chisq:** The model Chi-squared assesses overall fit and the discrepancy between the sample and fitted covariance matrices. Its p-value should be $> .05$ (i.e., the hypothesis of a perfect fit cannot be rejected). However, it is quite sensitive to sample size.
- **GFI/AGFI:** The (Adjusted) Goodness of Fit is the proportion of variance accounted for by the estimated population covariance. Analogous to R^2 . The GFI and the AGFI should be $> .95$ and $> .90$, respectively.
- **NFI/NNFI/TLI:** The (Non) Normed Fit Index. An NFI of 0.95, indicates the model of interest improves the fit by 95\ NNFI (also called the Tucker Lewis index; TLI) is preferable for smaller samples. They should be $> .90$ (Byrne, 1994) or $> .95$ (Schumacker & Lomax, 2004).

- **CFI**: The Comparative Fit Index is a revised form of NFI. Not very sensitive to sample size (Fan, Thompson, & Wang, 1999). Compares the fit of a target model to the fit of an independent, or null, model. It should be $> .90$.
- **RMSEA**: The Root Mean Square Error of Approximation is a parsimony-adjusted index. Values closer to 0 represent a good fit. It should be $< .08$ or $< .05$. The p-value printed with it tests the hypothesis that RMSEA is less than or equal to .05 (a cutoff sometimes used for good fit), and thus should be not significant.
- **RMR/SRMR**: the (Standardized) Root Mean Square Residual represents the square-root of the difference between the residuals of the sample covariance matrix and the hypothesized model. As the RMR can be sometimes hard to interpret, better to use SRMR. Should be $< .08$.
- **RFI**: the Relative Fit Index, also known as RHO1, is not guaranteed to vary from 0 to 1. However, RFI close to 1 indicates a good fit.
- **IFI**: the Incremental Fit Index (IFI) adjusts the Normed Fit Index (NFI) for sample size and degrees of freedom (Bollen's, 1989). Over 0.90 is a good fit, but the index can exceed 1.
- **PNFI**: the Parsimony-Adjusted Measures Index. There is no commonly agreed-upon cutoff value for an acceptable model for this index. Should be > 0.50 .

See the documentation for `fitmeasures()`.

What to report:

For structural equation models (SEM), Kline (2015) suggests that at a minimum the following indices should be reported: The model **chi-square**, the **RMSEA**, the **CFI** and the **SRMR**.

Note

When possible, it is recommended to report dynamic cutoffs of fit indices. See <https://dynamicfit.app/cfa/>.

References

- Awang, Z. (2012). A handbook on SEM. Structural equation modeling.
- Byrne, B. M. (1994). Structural equation modeling with EQS and EQS/Windows. Thousand Oaks, CA: Sage Publications.
- Tucker, L. R., and Lewis, C. (1973). The reliability coefficient for maximum likelihood factor analysis. *Psychometrika*, 38, 1-10.
- Schumacker, R. E., and Lomax, R. G. (2004). A beginner's guide to structural equation modeling, Second edition. Mahwah, NJ: Lawrence Erlbaum Associates.
- Fan, X., B. Thompson, and L. Wang (1999). Effects of sample size, estimation method, and model specification on structural equation modeling fit indexes. *Structural Equation Modeling*, 6, 56-83.
- Kline, R. B. (2015). Principles and practice of structural equation modeling. Guilford publications.

Examples

```
interpret_gfi(c(.5, .99))
interpret_agfi(c(.5, .99))
interpret_nfi(c(.5, .99))
```

```

interpret_nnfi(c(.5, .99))
interpret_cfi(c(.5, .99))
interpret_rmsea(c(.07, .04))
interpret_srmr(c(.5, .99))
interpret_rfi(c(.5, .99))
interpret_ifi(c(.5, .99))
interpret_pnfi(c(.5, .99))

# Structural Equation Models (SEM)
structure <- " ind60 =~ x1 + x2 + x3
              dem60 =~ y1 + y2 + y3
              dem60 ~ ind60 "
model <- lavaan::sem(structure, data = lavaan::PoliticalDemocracy)
interpret(model)

```

interpret_icc	<i>Interpret Intraclass Correlation Coefficient (ICC)</i>
---------------	---

Description

The value of an ICC lies between 0 to 1, with 0 indicating no reliability among raters and 1 indicating perfect reliability.

Usage

```
interpret_icc(icc, rules = "koo2016", ...)
```

Arguments

icc	Value or vector of Intraclass Correlation Coefficient (ICC) values.
rules	Can be "koo2016" (default) or custom set of rules() .
...	Not used for now.

Rules

- Koo (2016) ("koo2016"; default)
 - **ICC < 0.50** - Poor reliability
 - **0.5 <= ICC < 0.75** - Moderate reliability
 - **0.75 <= ICC < 0.9** - Good reliability
 - ****ICC >= 0.9**** - Excellent reliability

References

- Koo, T. K., and Li, M. Y. (2016). A guideline of selecting and reporting intraclass correlation coefficients for reliability research. *Journal of chiropractic medicine*, 15(2), 155-163.

Examples

```
interpret_icc(0.6)
interpret_icc(c(0.4, 0.8))
```

interpret_kendalls_w *Interpret Kendall's Coefficient of Concordance W*

Description

Interpret Kendall's Coefficient of Concordance W

Usage

```
interpret_kendalls_w(w, rules = "landis1977")
```

Arguments

w	Value or vector of Kendall's coefficient of concordance.
rules	Can be "landis1977" (default) or a custom set of <code>rules()</code> .

Rules

- Landis & Koch (1977) ("landis1977"; default)
 - **0.00** <= w < **0.20** - Slight agreement
 - **0.20** <= w < **0.40** - Fair agreement
 - **0.40** <= w < **0.60** - Moderate agreement
 - **0.60** <= w < **0.80** - Substantial agreement
 - w >= **0.80** - Almost perfect agreement

References

- Landis, J. R., & Koch G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, 33:159-74.

interpret_oddsratio *Interpret Odds Ratio*

Description

Interpret Odds Ratio

Usage

```
interpret_oddsratio(OR, rules = "chen2010", log = FALSE, ...)
```

Arguments

OR	Value or vector of (log) odds ratio values.
rules	Can be "chen2010" (default), "cohen1988" (through transformation to standardized difference, see oddsratio_to_d()) or custom set of rules() .
log	Are the provided values log odds ratio.
...	Currently not used.

Rules

Rules apply to OR as ratios, so OR of 10 is as extreme as a OR of 0.1 (1/10).

- Chen et al. (2010) ("chen2010"; default)
 - **OR < 1.68** - Very small
 - **1.68 <= OR < 3.47** - Small
 - **3.47 <= OR < 6.71** - Medium
 - ****OR >= 6.71 **** - Large
- Cohen (1988) ("cohen1988", based on the [oddsratio_to_d\(\)](#) conversion, see [interpret_cohens_d\(\)](#))
 - **OR < 1.44** - Very small
 - **1.44 <= OR < 2.48** - Small
 - **2.48 <= OR < 4.27** - Medium
 - ****OR >= 4.27 **** - Large

References

- Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd Ed.). New York: Routledge.
- Chen, H., Cohen, P., & Chen, S. (2010). How big is a big odds ratio? Interpreting the magnitudes of odds ratios in epidemiological studies. *Communications in Statistics-Simulation and Computation*, 39(4), 860-864.
- Sánchez-Meca, J., Marín-Martínez, F., & Chacón-Moscoso, S. (2003). Effect-size indices for dichotomized outcomes in meta-analysis. *Psychological methods*, 8(4), 448.

Examples

```
interpret_oddsratio(1)
interpret_oddsratio(c(5, 2))
```

```
interpret_omega_squared
      Interpret ANOVA Effect Sizes
```

Description

Interpret ANOVA Effect Sizes

Usage

```
interpret_omega_squared(es, rules = "field2013", ...)
interpret_eta_squared(es, rules = "field2013", ...)
interpret_epsilon_squared(es, rules = "field2013", ...)
```

Arguments

es	Value or vector of eta / omega / epsilon squared values.
rules	Can be "field2013" (default), "cohen1992" or custom set of <code>rules()</code> .
...	Not used for now.

Rules

- Field (2013) ("field2013"; default)
 - **ES < 0.01** - Very small
 - **0.01 <= ES < 0.06** - Small
 - **0.16 <= ES < 0.14** - Medium
 - ****ES >= 0.14 **** - Large
- Cohen (1992) ("cohen1992") applicable to one-way anova, or to *partial* eta / omega / epsilon squared in multi-way anova.
 - **ES < 0.02** - Very small
 - **0.02 <= ES < 0.13** - Small
 - **0.13 <= ES < 0.26** - Medium
 - **ES >= 0.26** - Large

References

- Field, A (2013) Discovering statistics using IBM SPSS Statistics. Fourth Edition. Sage:London.
- Cohen, J. (1992). A power primer. Psychological bulletin, 112(1), 155.

See Also

<https://imaging.mrc-cbu.cam.ac.uk/statswiki/FAQ/effectSize/>

Examples

```
interpret_eta_squared(.02)
interpret_eta_squared(c(.5, .02), rules = "cohen1992")
```

interpret_p	<i>Interpret p-Values</i>
-------------	---------------------------

Description

Interpret p -Values

Usage

```
interpret_p(p, rules = "default")
```

Arguments

p	Value or vector of p-values.
rules	Can be "default", "rss" (for <i>Redefine statistical significance</i> rules) or custom set of <code>rules()</code> .

Rules

- Default
 - $p \geq 0.05$ - Not significant
 - $p < 0.05$ - Significant
- Benjamin et al. (2018) ("rss")
 - $p \geq 0.05$ - Not significant
 - $0.005 \leq p < 0.05$ - Suggestive
 - $p < 0.005$ - Significant

References

- Benjamin, D. J., Berger, J. O., Johannesson, M., Nosek, B. A., Wagenmakers, E. J., Berk, R., ... & Cesarini, D. (2018). Redefine statistical significance. *Nature Human Behaviour*, 2(1), 6-10.

Examples

```
interpret_p(c(.5, .02, 0.001))
interpret_p(c(.5, .02, 0.001), rules = "rss")
```

interpret_pd	<i>Interpret Probability of Direction (pd)</i>
--------------	--

Description

Interpret Probability of Direction (pd)

Usage

```
interpret_pd(pd, rules = "default", ...)
```

Arguments

pd	Value or vector of probabilities of direction.
rules	Can be "default", "makowski2019" or a custom set of <code>rules()</code> .
...	Not directly used.

Rules

- Default (i.e., equivalent to p-values)
 - **pd** \leq **0.975** - not significant
 - **pd** $>$ **0.975** - significant
- Makowski et al. (2019) ("makowski2019")
 - **pd** \leq **0.95** - uncertain
 - **pd** $>$ **0.95** - possibly existing
 - **pd** $>$ **0.97** - likely existing
 - **pd** $>$ **0.99** - probably existing
 - **pd** $>$ **0.999** - certainly existing

References

- Makowski, D., Ben-Shachar, M. S., Chen, S. H., and Lüdtke, D. (2019). Indices of effect existence and significance in the Bayesian framework. *Frontiers in psychology*, 10, 2767.

Examples

```
interpret_pd(.98)
interpret_pd(c(.96, .99), rules = "makowski2019")
```

interpret_r *Interpret Correlation Coefficient*

Description

Interpret Correlation Coefficient

Usage

```
interpret_r(r, rules = "funder2019", ...)
interpret_phi(r, rules = "funder2019", ...)
interpret_cramers_v(r, rules = "funder2019", ...)
interpret_rank_biserial(r, rules = "funder2019", ...)
```

Arguments

r	Value or vector of correlation coefficient.
rules	Can be "funder2019" (default), "gignac2016", "cohen1988", "evans1996", "lovakov2021" or a custom set of <code>rules()</code> .
...	Not directly used.

Rules

Rules apply positive and negative r alike.

- Funder & Ozer (2019) ("funder2019"; default)
 - $r < 0.05$ - Tiny
 - $0.05 \leq r < 0.1$ - Very small
 - $0.1 \leq r < 0.2$ - Small
 - $0.2 \leq r < 0.3$ - Medium
 - $0.3 \leq r < 0.4$ - Large
 - $r \geq 0.4$ - Very large
- Gignac & Szodorai (2016) ("gignac2016")
 - $r < 0.1$ - Very small
 - $0.1 \leq r < 0.2$ - Small
 - $0.2 \leq r < 0.3$ - Moderate
 - $r \geq 0.3$ - Large
- Cohen (1988) ("cohen1988")
 - $r < 0.1$ - Very small
 - $0.1 \leq r < 0.3$ - Small
 - $0.3 \leq r < 0.5$ - Moderate

- $r \geq 0.5$ - Large
- Lovakov & Agadullina (2021) ("lovakov2021")
 - $r < 0.12$ - Very small
 - $0.12 \leq r < 0.24$ - Small
 - $0.24 \leq r < 0.41$ - Moderate
 - $r \geq 0.41$ - Large
- Evans (1996) ("evans1996")
 - $r < 0.2$ - Very weak
 - $0.2 \leq r < 0.4$ - Weak
 - $0.4 \leq r < 0.6$ - Moderate
 - $0.6 \leq r < 0.8$ - Strong
 - $r \geq 0.8$ - Very strong

Note

As ϕ can be larger than 1 - it is recommended to compute and interpret Cramer's V instead.

References

- Lovakov, A., & Agadullina, E. R. (2021). Empirically Derived Guidelines for Effect Size Interpretation in Social Psychology. *European Journal of Social Psychology*.
- Funder, D. C., & Ozer, D. J. (2019). Evaluating effect size in psychological research: sense and nonsense. *Advances in Methods and Practices in Psychological Science*.
- Gignac, G. E., & Szodorai, E. T. (2016). Effect size guidelines for individual differences researchers. *Personality and individual differences*, 102, 74-78.
- Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd Ed.). New York: Routledge.
- Evans, J. D. (1996). *Straightforward statistics for the behavioral sciences*. Thomson Brooks/Cole Publishing Co.

See Also

Page 88 of APA's 6th Edition.

Examples

```
interpret_r(.015)
interpret_r(c(.5, -.02))
interpret_r(.3, rules = "lovakov2021")
```

interpret_r2

*Interpret Coefficient of Determination (R^2)***Description**Interpret Coefficient of Determination (R^2)**Usage**

interpret_r2(r2, rules = "cohen1988")

Arguments

r2	Value or vector of R^2 values.
rules	Can be "cohen1988" (default), "falk1992", "chin1998", "hair2011", or custom set of <code>rules()</code>].

Rules**For Linear Regression:**

- Cohen (1988) ("cohen1988"; default)
 - $R^2 < 0.02$ - Very weak
 - $0.02 \leq R^2 < 0.13$ - Weak
 - $0.13 \leq R^2 < 0.26$ - Moderate
 - $R^2 \geq 0.26$ - Substantial
- Falk & Miller (1992) ("falk1992")
 - $R^2 < 0.1$ - Negligible
 - $R^2 \geq 0.1$ - Adequate

For PLS / SEM R-Squared of *latent* variables:

- Chin, W. W. (1998) ("chin1998")
 - $R^2 < 0.19$ - Very weak
 - $0.19 \leq R^2 < 0.33$ - Weak
 - $0.33 \leq R^2 < 0.67$ - Moderate
 - $R^2 \geq 0.67$ - Substantial
- Hair et al. (2011) ("hair2011")
 - $R^2 < 0.25$ - Very weak
 - $0.25 \leq R^2 < 0.50$ - Weak
 - $0.50 \leq R^2 < 0.75$ - Moderate
 - $R^2 \geq 0.75$ - Substantial

References

- Cohen, J. (1988). Statistical power analysis for the behavioral sciences (2nd Ed.). New York: Routledge.
- Falk, R. F., & Miller, N. B. (1992). A primer for soft modeling. University of Akron Press.
- Chin, W. W. (1998). The partial least squares approach to structural equation modeling. Modern methods for business research, 295(2), 295-336.
- Hair, J. F., Ringle, C. M., & Sarstedt, M. (2011). PLS-SEM: Indeed a silver bullet. Journal of Marketing theory and Practice, 19(2), 139-152.

Examples

```
interpret_r2(.02)
interpret_r2(c(.5, .02))
```

interpret_rope	<i>Interpret Bayesian Posterior Percentage in ROPE.</i>
----------------	---

Description

Interpretation of

Usage

```
interpret_rope(rope, ci = 0.9, rules = "default")
```

Arguments

rope	Value or vector of percentages in ROPE.
ci	The Credible Interval (CI) probability, corresponding to the proportion of HDI, that was used. Can be 1 in the case of "full ROPE".
rules	A character string (see details) or a custom set of <code>rules()</code> .

Rules

- Default
 - For $CI < 1$
 - * **Rope = 0** - Significant
 - * **0 < Rope < 1** - Undecided
 - * **Rope = 1** - Negligible
 - For $CI = 1$
 - * **Rope < 0.01** - Significant
 - * **0.01 < Rope < 0.025** - Probably significant
 - * **0.025 < Rope < 0.975** - Undecided
 - * **0.975 < Rope < 0.99** - Probably negligible
 - * **Rope > 0.99** - Negligible

References

BayestestR's reporting guidelines

Examples

```
interpret_rope(0, ci = 0.9)
interpret_rope(c(0.005, 0.99), ci = 1)
```

interpret_vif	<i>Interpret the Variance Inflation Factor (VIF)</i>
---------------	--

Description

Interpret VIF index of multicollinearity.

Usage

```
interpret_vif(vif, rules = "default")
```

Arguments

vif	Value or vector of VIFs.
rules	Can be "default" or a custom set of rules() .

Rules

- Default
 - **VIF < 5** - Low
 - **5 <= VIF < 10** - Moderate
 - **VIF >= 10** - High

Examples

```
interpret_vif(c(1.4, 30.4))
```

is_effectsize_name *Checks for a Valid Effect Size Name*

Description

For use by other functions and packages.

Usage

```
is_effectsize_name(x, ignore_case = TRUE)

get_effectsize_name(x, ignore_case = TRUE)

get_effectsize_label(
  x,
  ignore_case = TRUE,
  use_symbols = getOption("es.use_symbols", FALSE)
)
```

Arguments

x	A character, or a vector.
ignore_case	Should case of input be ignored?
use_symbols	Should proper symbols be printed (TRUE) instead of transliterated effect size names (FALSE). See effectsize_options .

mahalanobis_d *Mahalanobis' D (a multivariate Cohen's d)*

Description

Compute effect size indices for standardized difference between two normal multivariate distributions or between one multivariate distribution and a defined point. This is the standardized effect size for Hotelling's T^2 test (e.g., DescTools::HotellingsT2Test()). D is computed as:

$$D = \sqrt{(\bar{X}_1 - \bar{X}_2 - \mu)^T \Sigma_p^{-1} (\bar{X}_1 - \bar{X}_2 - \mu)}$$

Where \bar{X}_i are the column means, Σ_p is the *pooled* covariance matrix, and μ is a vector of the null differences for each variable. When there is only one variate, this formula reduces to Cohen's d .

Usage

```

mahalanobis_d(
  x,
  y = NULL,
  data = NULL,
  pooled_cov = TRUE,
  mu = 0,
  ci = 0.95,
  alternative = "two.sided",
  verbose = TRUE,
  ...
)

```

Arguments

<code>x, y</code>	A data frame or matrix. Any incomplete observations (with NA values) are dropped. <code>x</code> can also be a formula (see details) in which case <code>y</code> is ignored.
<code>data</code>	An optional data frame containing the variables.
<code>pooled_cov</code>	Should equal covariance be assumed? Currently only <code>pooled_cov = TRUE</code> is supported.
<code>mu</code>	A named list/vector of the true difference in means for each variable. Can also be a vector of length 1, which will be recycled.
<code>ci</code>	Confidence Interval (CI) level
<code>alternative</code>	a character string specifying the alternative hypothesis; Controls the type of CI returned: "two.sided" (default, two-sided CI), "greater" or "less" (one-sided CI). Partial matching is allowed (e.g., "g", "1", "two"...). See <i>One-Sided CIs</i> in effectsize_CIs .
<code>verbose</code>	Toggle warnings and messages on or off.
<code>...</code>	Not used.

Details

To specify a `x` as a formula:

- Two sample case: `DV1 + DV2 ~ group` or `cbind(DV1, DV2) ~ group`
- One sample case: `DV1 + DV2 ~ 1` or `cbind(DV1, DV2) ~ 1`

Value

A data frame with the Mahalanobis_D and potentially its CI (`CI_low` and `CI_high`).

Confidence (Compatibility) Intervals (CIs)

Unless stated otherwise, confidence (compatibility) intervals (CIs) are estimated using the non-centrality parameter method (also called the "pivot method"). This method finds the noncentrality parameter ("*nep*") of a noncentral *t*, *F*, or χ^2 distribution that places the observed *t*, *F*, or χ^2 test statistic at the desired probability point of the distribution. For example, if the observed *t* statistic is

2.0, with 50 degrees of freedom, for which cumulative noncentral t distribution is $t = 2.0$ the .025 quantile (answer: the noncentral t distribution with $ncp = .04$)? After estimating these confidence bounds on the ncp , they are converted into the effect size metric to obtain a confidence interval for the effect size (Steiger, 2004).

For additional details on estimation and troubleshooting, see [effectsize_CIs](#).

CI and Significance Tests

"Confidence intervals on measures of effect size convey all the information in a hypothesis test, and more." (Steiger, 2004). Confidence (compatibility) intervals and p values are complementary summaries of parameter uncertainty given the observed data. A dichotomous hypothesis test could be performed with either a CI or a p value. The 100 (1 - α)% confidence interval contains all of the parameter values for which $p > \alpha$ for the current data and model. For example, a 95% confidence interval contains all of the values for which $p > .05$.

Note that a confidence interval including 0 *does not* indicate that the null (no effect) is true. Rather, it suggests that the observed data together with the model and its assumptions combined do not provided clear evidence against a parameter value of 0 (same as with any other value in the interval), with the level of this evidence defined by the chosen α level (Rafi & Greenland, 2020; Schweder & Hjort, 2016; Xie & Singh, 2013). To infer no effect, additional judgments about what parameter values are "close enough" to 0 to be negligible are needed ("equivalence testing"; Bauer & Kiesser, 1996).

References

- Del Giudice, M. (2017). Heterogeneity coefficients for Mahalanobis' D as a multivariate effect size. *Multivariate Behavioral Research*, 52(2), 216-221.
- Mahalanobis, P. C. (1936). On the generalized distance in statistics. National Institute of Science of India.
- Reiser, B. (2001). Confidence intervals for the Mahalanobis distance. *Communications in Statistics-Simulation and Computation*, 30(1), 37-45.

See Also

[stats::mahalanobis\(\)](#), [cov_pooled\(\)](#)

Other standardized differences: [cohens_d\(\)](#), [p_superiority\(\)](#), [rank_biserial\(\)](#)

Examples

```
## Two samples -----
mtcars_am0 <- subset(mtcars, am == 0,
  select = c(mpg, hp, cyl)
)
mtcars_am1 <- subset(mtcars, am == 1,
  select = c(mpg, hp, cyl)
)

mahalanobis_d(mtcars_am0, mtcars_am1)
```

```

# Or
mahalanobis_d(mpg + hp + cyl ~ am, data = mtcars)

mahalanobis_d(mpg + hp + cyl ~ am, data = mtcars, alternative = "greater")

# Different mu:
mahalanobis_d(mpg + hp + cyl ~ am,
  data = mtcars,
  mu = c(mpg = -4, hp = 15, cyl = 0)
)

# D is a multivariate d, so when only 1 variate is provided:
mahalanobis_d(hp ~ am, data = mtcars)

cohens_d(hp ~ am, data = mtcars)

# One sample -----
mahalanobis_d(mtcars[, c("mpg", "hp", "cyl")])

# Or
mahalanobis_d(mpg + hp + cyl ~ 1,
  data = mtcars,
  mu = c(mpg = 15, hp = 5, cyl = 3)
)

```

Music_preferences

Music Preference by College Major

Description

Fictional data.

Format

A 4-by-3 table, with a *column* for each major and a *row* for each type of music.

```

data("Music_preferences")
Music_preferences
#>      Pop Rock Jazz Classic
#> Psych 150  100  165   130
#> Econ   50   65   35    10
#> Law    2   55   40    25

```


See Also

Other effect size datasets: [Music_preferences2](#), [RCT_table](#), [Smoking_FASD](#), [food_class](#), [hardlyworking](#), [screening_test](#)

Music_preferences2 *Music Preference by College Major*

Description

Fictional data, with more extreme preferences than [Music_preferences](#)

Format

A 4-by-3 table, with a *column* for each major and a *row* for each type of music.

```
data("Music_preferences2")
Music_preferences2
#>      Pop Rock Jazz Classic
#> Psych 151 130 12      7
#> Econ  77   6 111     4
#> Law   0   4  2     165
```

See Also

Other effect size datasets: [Music_preferences](#), [RCT_table](#), [Smoking_FASD](#), [food_class](#), [hardlyworking](#), [screening_test](#)

oddsratio *Odds Ratios, Risk Ratios and Cohen's h for 2-by-2 Contingency Tables*

Description

Report with any `stats::chisq.test()` or `stats::fisher.test()`.

Note that these are computed with each **column** representing the different groups, and the *first* column representing the treatment group and the *second* column baseline (or control). Effects are given as treatment / control. If you wish you use rows as groups you must pass a transposed table, or switch the x and y arguments.

Usage

```
oddsratio(x, y = NULL, ci = 0.95, alternative = "two.sided", log = FALSE, ...)
```

```
riskratio(x, y = NULL, ci = 0.95, alternative = "two.sided", log = FALSE, ...)
```

```
cohens_h(x, y = NULL, ci = 0.95, alternative = "two.sided", ...)
```

Arguments

x	a numeric vector or matrix. x and y can also both be factors.
y	a numeric vector; ignored if x is a matrix. If x is a factor, y should be a factor of the same length.
ci	Confidence Interval (CI) level
alternative	a character string specifying the alternative hypothesis; Controls the type of CI returned: "two.sided" (two-sided CI; default), "greater" (one-sided CI) or "less" (one-sided CI). Partial matching is allowed (e.g., "g", "l", "two"...). See <i>One-Sided CIs</i> in effectsize_CIs .
log	Take in or output the log of the ratio (such as in logistic models).
...	Ignored

Value

A data frame with the effect size (Odds_ratio, Risk_ratio (possibly with the prefix log_), Cohens_h) and its CIs (CI_low and CI_high).

Confidence (Compatibility) Intervals (CIs)

For Odds ratios, Risk ratios and Cohen's *h*, confidence intervals are estimated using the standard normal parametric method (see Katz et al., 1978; Szumilas, 2010).

CIs and Significance Tests

"Confidence intervals on measures of effect size convey all the information in a hypothesis test, and more." (Steiger, 2004). Confidence (compatibility) intervals and p values are complementary summaries of parameter uncertainty given the observed data. A dichotomous hypothesis test could be performed with either a CI or a p value. The 100 (1 - α)% confidence interval contains all of the parameter values for which $p > \alpha$ for the current data and model. For example, a 95% confidence interval contains all of the values for which $p > .05$.

Note that a confidence interval including 0 *does not* indicate that the null (no effect) is true. Rather, it suggests that the observed data together with the model and its assumptions combined do not provided clear evidence against a parameter value of 0 (same as with any other value in the interval), with the level of this evidence defined by the chosen α level (Rafi & Greenland, 2020; Schweder & Hjort, 2016; Xie & Singh, 2013). To infer no effect, additional judgments about what parameter values are "close enough" to 0 to be negligible are needed ("equivalence testing"; Bauer & Kiessner, 1996).

References

- Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd Ed.). New York: Routledge.
- Katz, D. J. S. M., Baptista, J., Azen, S. P., & Pike, M. C. (1978). Obtaining confidence intervals for the risk ratio in cohort studies. *Biometrics*, 469-474.
- Szumilas, M. (2010). Explaining odds ratios. *Journal of the Canadian academy of child and adolescent psychiatry*, 19(3), 227.

See Also

Other effect sizes for contingency table: [cohens_g\(\)](#), [phi\(\)](#)

Examples

```
data("RCT_table")
RCT_table # note groups are COLUMNS

oddsratio(RCT_table)
oddsratio(RCT_table, alternative = "greater")

riskratio(RCT_table)

cohens_h(RCT_table)
```

oddsratio_to_riskratio

Convert Between Odds Ratios and Risk Ratios

Description

Convert Between Odds Ratios and Risk Ratios

Usage

```
oddsratio_to_riskratio(OR, p0, log = FALSE, ...)
riskratio_to_oddsratio(RR, p0, log = FALSE)
```

Arguments

OR, RR	Risk ratio of $p1/p0$ or Odds ratio of $odds(p1)/odds(p0)$, possibly log-ed. OR can also be a logistic regression model.
$p0$	Baseline risk
log	Take in or output the log of the ratio (such as in logistic models).
...	Arguments passed to and from other methods.

Value

Converted index, or if OR is a logistic regression model, a parameter table with the converted indices.

References

Grant, R. L. (2014). Converting an odds ratio to a range of plausible relative risks for better communication of research findings. *Bmj*, 348, f7450.

See Also

[oddsratio\(\)](#) and [riskratio\(\)](#)

Other convert between effect sizes: [d_to_r\(\)](#), [diff_to_cles](#), [eta2_to_f2\(\)](#), [odds_to_probs\(\)](#)

Examples

```
p0 <- 0.4
p1 <- 0.7

(OR <- probs_to_odds(p1) / probs_to_odds(p0))
(RR <- p1 / p0)

riskratio_to_oddsratio(RR, p0 = p0)
oddsratio_to_riskratio(OR, p0 = p0)

m <- glm(am ~ factor(cyl),
  data = mtcars,
  family = binomial()
)
oddsratio_to_riskratio(m)
```

odds_to_probs

Convert Between Odds and Probabilities

Description

Convert Between Odds and Probabilities

Usage

```
odds_to_probs(odds, log = FALSE, ...)

## S3 method for class 'data.frame'
odds_to_probs(odds, log = FALSE, select = NULL, exclude = NULL, ...)

probs_to_odds(probs, log = FALSE, ...)

## S3 method for class 'data.frame'
probs_to_odds(probs, log = FALSE, select = NULL, exclude = NULL, ...)
```

Arguments

odds	The <i>Odds</i> (or $\log(\text{odds})$ when $\text{log} = \text{TRUE}$) to convert.
log	Take in or output log odds (such as in logistic models).
...	Arguments passed to or from other methods.
select	When a data frame is passed, character or list of of column names to be transformed.

exclude	When a data frame is passed, character or list of column names to be excluded from transformation.
probs	Probability values to convert.

Value

Converted index.

See Also

[stats::plogis\(\)](#)

Other convert between effect sizes: [d_to_r\(\)](#), [diff_to_cles](#), [eta2_to_f2\(\)](#), [oddsratio_to_riskratio\(\)](#)

Examples

```
odds_to_probs(3)
odds_to_probs(1.09, log = TRUE)

probs_to_odds(0.95)
probs_to_odds(0.95, log = TRUE)
```

 phi

φ and Other Contingency Tables Correlations

Description

Compute phi (ϕ), Cramer's V , Tschuprow's T , Cohen's w , F_{ei} , Pearson's contingency coefficient for contingency tables or goodness-of-fit. Pair with any reported [stats::chisq.test\(\)](#).

Usage

```
phi(x, y = NULL, adjust = TRUE, ci = 0.95, alternative = "greater", ...)
cramers_v(x, y = NULL, adjust = TRUE, ci = 0.95, alternative = "greater", ...)
tschuprows_t(x, y = NULL, ci = 0.95, alternative = "greater", ...)

cohens_w(
  x,
  y = NULL,
  p = rep(1, length(x)),
  ci = 0.95,
  alternative = "greater",
  ...
)

fei(x, p = rep(1, length(x)), ci = 0.95, alternative = "greater", ...)
```

```

pearsons_c(
  x,
  y = NULL,
  p = rep(1, length(x)),
  ci = 0.95,
  alternative = "greater",
  ...
)

```

Arguments

x	a numeric vector or matrix. x and y can also both be factors.
y	a numeric vector; ignored if x is a matrix. If x is a factor, y should be a factor of the same length.
adjust	Should the effect size be bias-corrected? Defaults to TRUE; Advisable for small samples and large tables.
ci	Confidence Interval (CI) level
alternative	a character string specifying the alternative hypothesis; Controls the type of CI returned: "greater" (default) or "less" (one-sided CI), or "two.sided" (default, two-sided CI). Partial matching is allowed (e.g., "g", "l", "two"...). See <i>One-Sided CIs</i> in effectsize_CIs .
...	Ignored.
p	a vector of probabilities of the same length as x. An error is given if any entry of p is negative.

Details

phi (ϕ), Cramer's V , Tschuprow's T , Cohen's w , and Pearson's C are effect sizes for tests of independence in 2D contingency tables. For 2-by-2 tables, phi, Cramer's V , Tschuprow's T , and Cohen's w are identical, and are equal to the simple correlation between two dichotomous variables, ranging between 0 (no dependence) and 1 (perfect dependence).

For larger tables, Cramer's V , Tschuprow's T or Pearson's C should be used, as they are bounded between 0-1. (Cohen's w can also be used, but since it is not bounded at 1 (can be larger) its interpretation is more difficult.) For square table, Cramer's V and Tschuprow's T give the same results, but for non-square tables Tschuprow's T is more conservative: while V will be 1 if either columns are fully dependent on rows (for each column, there is only one non-0 cell) *or* rows are fully dependent on columns, T will only be 1 if both are true.

For goodness-of-fit in 1D tables Cohen's W , Fei or Pearson's C can be used. Cohen's w has no upper bound (can be arbitrarily large, depending on the expected distribution). Fei is an adjusted Cohen's w , accounting for the expected distribution, making it bounded between 0-1. Pearson's C is also bounded between 0-1.

To summarize, for correlation-like effect sizes, we recommend:

- For a 2x2 table, use phi ()

- For larger tables, use `cramers_v()`
- For goodness-of-fit, use `fei()`

Value

A data frame with the effect size (Cramers_v, phi (possibly with the suffix `_adjusted`), Cohens_w, Fei) and its CIs (CI_low and CI_high).

Confidence (Compatibility) Intervals (CIs)

Unless stated otherwise, confidence (compatibility) intervals (CIs) are estimated using the non-centrality parameter method (also called the "pivot method"). This method finds the noncentrality parameter ("*ncp*") of a noncentral *t*, *F*, or χ^2 distribution that places the observed *t*, *F*, or χ^2 test statistic at the desired probability point of the distribution. For example, if the observed *t* statistic is 2.0, with 50 degrees of freedom, for which cumulative noncentral *t* distribution is *t* = 2.0 the .025 quantile (answer: the noncentral *t* distribution with *ncp* = .04)? After estimating these confidence bounds on the *ncp*, they are converted into the effect size metric to obtain a confidence interval for the effect size (Steiger, 2004).

For additional details on estimation and troubleshooting, see [effectsize_CIs](#).

CIs and Significance Tests

"Confidence intervals on measures of effect size convey all the information in a hypothesis test, and more." (Steiger, 2004). Confidence (compatibility) intervals and p values are complementary summaries of parameter uncertainty given the observed data. A dichotomous hypothesis test could be performed with either a CI or a p value. The 100 (1 - α)% confidence interval contains all of the parameter values for which $p > \alpha$ for the current data and model. For example, a 95% confidence interval contains all of the values for which $p > .05$.

Note that a confidence interval including 0 *does not* indicate that the null (no effect) is true. Rather, it suggests that the observed data together with the model and its assumptions combined do not provided clear evidence against a parameter value of 0 (same as with any other value in the interval), with the level of this evidence defined by the chosen α level (Rafi & Greenland, 2020; Schweder & Hjort, 2016; Xie & Singh, 2013). To infer no effect, additional judgments about what parameter values are "close enough" to 0 to be negligible are needed ("equivalence testing"; Bauer & Kiesser, 1996).

References

- Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd Ed.). New York: Routledge.
- Johnston, J. E., Berry, K. J., & Mielke Jr, P. W. (2006). Measures of effect size for chi-squared and likelihood-ratio goodness-of-fit tests. *Perceptual and motor skills*, 103(2), 412-414.
- Rosenberg, M. S. (2010). A generalized formula for converting chi-square tests to effect sizes for meta-analysis. *PloS one*, 5(4), e10059.

See Also

[chisq_to_phi\(\)](#) for details regarding estimation and CIs.

Other effect sizes for contingency table: [cohens_g\(\)](#), [oddsratio\(\)](#)

Examples

```
## 2-by-2 tables
## -----
data("RCT_table")
RCT_table # note groups are COLUMNS

phi(RCT_table)
pearsons_c(RCT_table)

## Larger tables
## -----
data("Music_preferences")
Music_preferences

cramers_v(Music_preferences)

cohens_w(Music_preferences)

pearsons_c(Music_preferences)

## Goodness of fit
## -----
data("Smoking_FASD")
Smoking_FASD

fei(Smoking_FASD)

cohens_w(Smoking_FASD)

pearsons_c(Smoking_FASD)

# Use custom expected values:
fei(Smoking_FASD, p = c(0.015, 0.010, 0.975))

cohens_w(Smoking_FASD, p = c(0.015, 0.010, 0.975))

pearsons_c(Smoking_FASD, p = c(0.015, 0.010, 0.975))
```

plot.effectsize_table *Methods for {effectsize} Tables*

Description

Printing, formatting and plotting methods for effectsize tables.

Usage

```
## S3 method for class 'effectsize_table'
plot(x, ...)

## S3 method for class 'effectsize_table'
print(x, digits = 2, use_symbols = getOption("es.use_symbols", FALSE), ...)

## S3 method for class 'effectsize_table'
print_md(x, digits = 2, use_symbols = getOption("es.use_symbols", FALSE), ...)

## S3 method for class 'effectsize_table'
print_html(
  x,
  digits = 2,
  use_symbols = getOption("es.use_symbols", FALSE),
  ...
)

## S3 method for class 'effectsize_table'
format(
  x,
  digits = 2,
  output = c("text", "markdown", "html"),
  use_symbols = getOption("es.use_symbols", FALSE),
  ...
)

## S3 method for class 'effectsize_difference'
print(x, digits = 2, append_CLES = NULL, ...)
```

Arguments

x	Object to print.
...	Arguments passed to or from other functions.
digits	Number of digits for rounding or significant figures. May also be "signif" to return significant figures or "scientific" to return scientific notation. Control the number of digits by adding the value as suffix, e.g. digits = "scientific4" to have scientific notation with 4 decimal places, or digits = "signif5" for 5 significant figures (see also signif()).

use_symbols	Should proper symbols be printed (TRUE) instead of transliterated effect size names (FALSE). See effectsize_options .
output	Which output is the formatting intended for? Affects how title and footers are formatted.
append_CLES	Which Common Language Effect Sizes should be printed as well? Only applicable to Cohen's d , Hedges' g for independent samples of equal variance (pooled sd) or for the rank-biserial correlation for independent samples (See d_to_cles).

See Also

[insight::display\(\)](#)

p_superiority

Cohen's Us and Other Common Language Effect Sizes (CLES)

Description

Cohen's U_1 , U_2 , and U_3 , probability of superiority, proportion of overlap, Wilcoxon-Mann-Whitney odds, and Vargha and Delaney's A are CLESs. These are effect sizes that represent differences between two (independent) distributions in probabilistic terms (See details). Pair with any reported [stats::t.test\(\)](#) or [stats::wilcox.test\(\)](#).

Usage

```
p_superiority(
  x,
  y = NULL,
  data = NULL,
  mu = 0,
  paired = FALSE,
  parametric = TRUE,
  ci = 0.95,
  alternative = "two.sided",
  verbose = TRUE,
  ...
)

cohens_u1(
  x,
  y = NULL,
  data = NULL,
  mu = 0,
  parametric = TRUE,
  ci = 0.95,
  alternative = "two.sided",
  iterations = 200,
```

```
    verbose = TRUE,  
    ...  
  )  
  
cohens_u2(  
  x,  
  y = NULL,  
  data = NULL,  
  mu = 0,  
  parametric = TRUE,  
  ci = 0.95,  
  alternative = "two.sided",  
  iterations = 200,  
  verbose = TRUE,  
  ...  
)  
  
cohens_u3(  
  x,  
  y = NULL,  
  data = NULL,  
  mu = 0,  
  parametric = TRUE,  
  ci = 0.95,  
  alternative = "two.sided",  
  iterations = 200,  
  verbose = TRUE,  
  ...  
)  
  
p_overlap(  
  x,  
  y = NULL,  
  data = NULL,  
  mu = 0,  
  parametric = TRUE,  
  ci = 0.95,  
  alternative = "two.sided",  
  iterations = 200,  
  verbose = TRUE,  
  ...  
)  
  
vd_a(  
  x,  
  y = NULL,  
  data = NULL,  
  mu = 0,
```

```

    ci = 0.95,
    alternative = "two.sided",
    verbose = TRUE,
    ...
)

wmm_odds(
  x,
  y = NULL,
  data = NULL,
  mu = 0,
  paired = FALSE,
  ci = 0.95,
  alternative = "two.sided",
  verbose = TRUE,
  ...
)

```

Arguments

x, y	A numeric vector, or a character name of one in data. Any missing values (NAs) are dropped from the resulting vector. x can also be a formula (see <code>stats::t.test()</code>), in which case y is ignored.
data	An optional data frame containing the variables.
mu	a number indicating the true value of the mean (or difference in means if you are performing a two sample test).
paired	If TRUE, the values of x and y are considered as paired. This produces an effect size that is equivalent to the one-sample effect size on x - y.
parametric	Use parametric estimation (see <code>cohens_d()</code>) or non-parametric estimation (see <code>rank_biserial()</code>). See details.
ci	Confidence Interval (CI) level
alternative	a character string specifying the alternative hypothesis; Controls the type of CI returned: "two.sided" (default, two-sided CI), "greater" or "less" (one-sided CI). Partial matching is allowed (e.g., "g", "1", "two"...). See <i>One-Sided CIs</i> in <code>effectsize_CIs</code> .
verbose	Toggle warnings and messages on or off.
...	Arguments passed to or from other methods. When x is a formula, these can be subset and na.action.
iterations	The number of bootstrap replicates for computing confidence intervals. Only applies when ci is not NULL and parametric = FALSE.

Details

These measures of effect size present group differences in probabilistic terms:

- **Probability of superiority** is the probability that, when sampling an observation from each of the groups at random, that the observation from the second group will be larger than the sample

from the first group. For the one-sample (or paired) case, it is the probability that the sample (or difference) is larger than μ . (Vargha and Delaney's A is an alias for the non-parametric probability of superiority.)

- **Cohen's U_1** is the proportion of the total of both distributions that does not overlap.
- **Cohen's U_2** is the proportion of one of the groups that exceeds *the same proportion* in the other group.
- **Cohen's U_3** is the proportion of the second group that is smaller than the median of the first group.
- **Overlap (OVL)** is the proportional overlap between the distributions. (When `parametric = FALSE`, `bayestestR::overlap()` is used.)

Wilcoxon-Mann-Whitney odds are the *odds* of non-parametric superiority (via `probs_to_odds()`), that is the odds that, when sampling an observation from each of the groups at random, that the observation from the second group will be larger than the sample from the first group.

Where U_1 , U_2 , and *Overlap* are agnostic to the direction of the difference between the groups, U_3 and probability of superiority are not.

The parametric version of these effects assumes normality of both populations and homoscedasticity. If those are not met, the non parametric versions should be used.

Value

A data frame containing the common language effect sizes (and optionally their CIs).

Confidence (Compatibility) Intervals (CIs)

For parametric CLES, the CIs are transformed CIs for Cohen's d (see `d_to_u3()`). For non-parametric (`parametric = FALSE`) CLES, the CI of $Pr(\text{superiority})$ is a transformed CI of the rank-biserial correlation (`rb_to_p_superiority()`), while for all others, confidence intervals are estimated using the bootstrap method (using the `{boot}` package).

CIs and Significance Tests

"Confidence intervals on measures of effect size convey all the information in a hypothesis test, and more." (Steiger, 2004). Confidence (compatibility) intervals and p values are complementary summaries of parameter uncertainty given the observed data. A dichotomous hypothesis test could be performed with either a CI or a p value. The 100 (1 - α)% confidence interval contains all of the parameter values for which $p > \alpha$ for the current data and model. For example, a 95% confidence interval contains all of the values for which $p > .05$.

Note that a confidence interval including 0 *does not* indicate that the null (no effect) is true. Rather, it suggests that the observed data together with the model and its assumptions combined do not provided clear evidence against a parameter value of 0 (same as with any other value in the interval), with the level of this evidence defined by the chosen α level (Rafi & Greenland, 2020; Schweder & Hjort, 2016; Xie & Singh, 2013). To infer no effect, additional judgments about what parameter values are "close enough" to 0 to be negligible are needed ("equivalence testing"; Bauer & Kiesser, 1996).

Bootstrapped CIs

Some effect sizes are directionless—they do have a minimum value that would be interpreted as "no effect", but they cannot cross it. For example, a null value of [Kendall's W](#) is 0, indicating no difference between groups, but it can never have a negative value. Same goes for [U2](#) and [Overlap](#): the null value of U_2 is 0.5, but it can never be smaller than 0.5; an *Overlap* of 1 means "full overlap" (no difference), but it cannot be larger than 1.

When bootstrapping CIs for such effect sizes, the bounds of the CIs will never cross (and often will never cover) the null. Therefore, these CIs should not be used for statistical inference.

Note

If μ is not 0, the effect size represents the difference between the first *shifted sample* (by μ) and the second sample.

References

- Cohen, J. (1977). *Statistical power analysis for the behavioral sciences*. New York: Routledge.
- Reiser, B., & Faraggi, D. (1999). Confidence intervals for the overlapping coefficient: the normal equal variance case. *Journal of the Royal Statistical Society*, 48(3), 413-418.
- Ruscio, J. (2008). A probability-based measure of effect size: robustness to base rates and other factors. *Psychological methods*, 13(1), 19–30.
- Vargha, A., & Delaney, H. D. (2000). A critique and improvement of the CL common language effect size statistics of McGraw and Wong. *Journal of Educational and Behavioral Statistics*, 25(2), 101-132.
- O'Brien, R. G., & Casteloe, J. (2006, March). Exploiting the link between the Wilcoxon-Mann-Whitney test and a simple odds statistic. In *Proceedings of the Thirty-first Annual SAS Users Group International Conference* (pp. 209-31). Cary, NC: SAS Institute.
- Agresti, A. (1980). Generalized odds ratios for ordinal data. *Biometrics*, 59-67.

See Also

[sd_pooled\(\)](#)

Other standardized differences: [cohens_d\(\)](#), [mahalanobis_d\(\)](#), [rank_biserial\(\)](#)

Other rank-based effect sizes: [rank_biserial\(\)](#), [rank_epsilon_squared\(\)](#)

Examples

```
cohens_u2(mpg ~ am, data = mtcars)
```

```
p_superiority(mpg ~ am, data = mtcars, parametric = FALSE)
```

```
wmw_odds(mpg ~ am, data = mtcars)
```

```
x <- c(1.83, 0.5, 1.62, 2.48, 1.68, 1.88, 1.55, 3.06, 1.3)
```

```
y <- c(0.878, 0.647, 0.598, 2.05, 1.06, 1.29, 1.06, 3.14, 1.29)
```

```

p_overlap(x, y)
p_overlap(y, x) # direction of effect does not matter

cohens_u3(x, y)
cohens_u3(y, x) # direction of effect does matter

```

rank_biserial	<i>Dominance Effect Sizes for Rank Based Differences</i>
---------------	--

Description

Compute the rank-biserial correlation (r_{rb}) and Cliff's *delta* (δ) effect sizes for non-parametric (rank sum) differences. These effect sizes of dominance are closely related to the [Common Language Effect Sizes](#). Pair with any reported `stats::wilcox.test()`.

Usage

```

rank_biserial(
  x,
  y = NULL,
  data = NULL,
  mu = 0,
  paired = FALSE,
  ci = 0.95,
  alternative = "two.sided",
  verbose = TRUE,
  ...
)

cliffs_delta(
  x,
  y = NULL,
  data = NULL,
  mu = 0,
  ci = 0.95,
  alternative = "two.sided",
  verbose = TRUE,
  ...
)

```

Arguments

x, y	A numeric or ordered vector, or a character name of one in data. Any missing values (NAs) are dropped from the resulting vector. x can also be a formula (see stats::wilcox.test()), in which case y is ignored.
data	An optional data frame containing the variables.

mu	a number indicating the value around which (a-)symmetry (for one-sample or paired samples) or shift (for independent samples) is to be estimated. See stats::wilcox.test .
paired	If TRUE, the values of x and y are considered as paired. This produces an effect size that is equivalent to the one-sample effect size on $x - y$.
ci	Confidence Interval (CI) level
alternative	a character string specifying the alternative hypothesis; Controls the type of CI returned: "two.sided" (default, two-sided CI), "greater" or "less" (one-sided CI). Partial matching is allowed (e.g., "g", "1", "two"...). See <i>One-Sided CIs</i> in effectsize_CIs .
verbose	Toggle warnings and messages on or off.
...	Arguments passed to or from other methods. When x is a formula, these can be subset and na.action.

Details

The rank-biserial correlation is appropriate for non-parametric tests of differences - both for the one sample or paired samples case, that would normally be tested with Wilcoxon's Signed Rank Test (giving the **matched-pairs** rank-biserial correlation) and for two independent samples case, that would normally be tested with Mann-Whitney's *U* Test (giving **Glass'** rank-biserial correlation). See [stats::wilcox.test](#). In both cases, the correlation represents the difference between the proportion of favorable and unfavorable pairs / signed ranks (Kerby, 2014). Values range from -1 complete dominance of the second sample (*all* values of the second sample are larger than *all* the values of the first sample) to +1 complete dominance of the first sample (*all* values of the second sample are smaller than *all* the values of the first sample).

Cliff's *delta* is an alias to the rank-biserial correlation in the two sample case.

Value

A data frame with the effect size `r_rank_biserial` and its CI (`CI_low` and `CI_high`).

Ties

When tied values occur, they are each given the average of the ranks that would have been given had no ties occurred. This results in an effect size of reduced magnitude. A correction has been applied for Kendall's *W*.

Confidence (Compatibility) Intervals (CIs)

Confidence intervals for the rank-biserial correlation (and Cliff's *delta*) are estimated using the normal approximation (via Fisher's transformation).

CIs and Significance Tests

"Confidence intervals on measures of effect size convey all the information in a hypothesis test, and more." (Steiger, 2004). Confidence (compatibility) intervals and p values are complementary summaries of parameter uncertainty given the observed data. A dichotomous hypothesis test could be performed with either a CI or a p value. The 100 (1 - α)% confidence interval contains all of the

parameter values for which $p > \alpha$ for the current data and model. For example, a 95% confidence interval contains all of the values for which $p > .05$.

Note that a confidence interval including 0 *does not* indicate that the null (no effect) is true. Rather, it suggests that the observed data together with the model and its assumptions combined do not provide clear evidence against a parameter value of 0 (same as with any other value in the interval), with the level of this evidence defined by the chosen α level (Rafi & Greenland, 2020; Schweder & Hjort, 2016; Xie & Singh, 2013). To infer no effect, additional judgments about what parameter values are "close enough" to 0 to be negligible are needed ("equivalence testing"; Bauer & Kiessler, 1996).

References

- Cureton, E. E. (1956). Rank-biserial correlation. *Psychometrika*, 21(3), 287-290.
- Glass, G. V. (1965). A ranking variable analogue of biserial correlation: Implications for short-cut item analysis. *Journal of Educational Measurement*, 2(1), 91-95.
- Kerby, D. S. (2014). The simple difference formula: An approach to teaching nonparametric correlation. *Comprehensive Psychology*, 3, 11-IT.
- King, B. M., & Minium, E. W. (2008). *Statistical reasoning in the behavioral sciences*. John Wiley & Sons Inc.
- Cliff, N. (1993). Dominance statistics: Ordinal analyses to answer ordinal questions. *Psychological bulletin*, 114(3), 494.
- Tomczak, M., & Tomczak, E. (2014). The need to report effect size estimates revisited. An overview of some recommended measures of effect size.

See Also

Other standardized differences: [cohens_d\(\)](#), [mahalanobis_d\(\)](#), [p_superiority\(\)](#)

Other rank-based effect sizes: [p_superiority\(\)](#), [rank_epsilon_squared\(\)](#)

Examples

```
data(mtcars)
mtcars$am <- factor(mtcars$am)
mtcars$cyl <- factor(mtcars$cyl)

# Two Independent Samples -----
(rb <- rank_biserial(mpg ~ am, data = mtcars))
# Same as:
# rank_biserial("mpg", "am", data = mtcars)
# rank_biserial(mtcars$mpg[mtcars$am=="0"], mtcars$mpg[mtcars$am=="1"])

# More options:
rank_biserial(mpg ~ am, data = mtcars, mu = -5)
print(rb, append_CLES = TRUE)

# One Sample -----
```

```

rank_biserial(wt ~ 1, data = mtcars, mu = 3)
# same as:
# rank_biserial("wt", data = mtcars, mu = 3)
# rank_biserial(mtcars$wt, mu = 3)

# Paired Samples -----
dat <- data.frame(
  Cond1 = c(1.83, 0.5, 1.62, 2.48, 1.68, 1.88, 1.55, 3.06, 1.3),
  Cond2 = c(0.878, 0.647, 0.598, 2.05, 1.06, 1.29, 1.06, 3.14, 1.29)
)
(rb <- rank_biserial(Pair(Cond1, Cond2) ~ 1, data = dat, paired = TRUE))

# same as:
# rank_biserial(dat$Cond1, dat$Cond2, paired = TRUE)

interpret_rank_biserial(0.78)
interpret(rb, rules = "funder2019")

```

rank_epsilon_squared *Effect Size for Rank Based ANOVA*

Description

Compute rank epsilon squared (E_R^2) or rank eta squared (η_H^2) (to accompany `stats::kruskal.test()`), and Kendall's W (to accompany `stats::friedman.test()`) effect sizes for non-parametric (rank sum) one-way ANOVAs.

Usage

```

rank_epsilon_squared(
  x,
  groups,
  data = NULL,
  ci = 0.95,
  alternative = "greater",
  iterations = 200,
  verbose = TRUE,
  ...
)

rank_eta_squared(
  x,
  groups,
  data = NULL,
  ci = 0.95,
  alternative = "greater",

```

```

    iterations = 200,
    verbose = TRUE,
    ...
)

kendalls_w(
  x,
  groups,
  blocks,
  data = NULL,
  blocks_on_rows = TRUE,
  ci = 0.95,
  alternative = "greater",
  iterations = 200,
  verbose = TRUE,
  ...
)

```

Arguments

x	Can be one of: <ul style="list-style-type: none"> • A numeric or ordered vector, or a character name of one in data. • A list of vectors (for rank_eta/epsilon_squared()). • A matrix of blocks x groups (for kendalls_w()) (or groups x blocks if blocks_on_rows = FALSE). See details for the blocks and groups terminology used here. • A formula in the form of: <ul style="list-style-type: none"> – DV ~ groups for rank_eta/epsilon_squared(). – DV ~ groups blocks for kendalls_w() (See details for the blocks and groups terminology used here).
groups, blocks	A factor vector giving the group / block for the corresponding elements of x, or a character name of one in data. Ignored if x is not a vector.
data	An optional data frame containing the variables.
ci	Confidence Interval (CI) level
alternative	a character string specifying the alternative hypothesis; Controls the type of CI returned: "two.sided" (default, two-sided CI), "greater" or "less" (one-sided CI). Partial matching is allowed (e.g., "g", "1", "two"...). See <i>One-Sided CIs</i> in effectsize_CIs .
iterations	The number of bootstrap replicates for computing confidence intervals. Only applies when ci is not NULL.
verbose	Toggle warnings and messages on or off.
...	Arguments passed to or from other methods. When x is a formula, these can be subset and na.action.
blocks_on_rows	Are blocks on rows (TRUE) or columns (FALSE).

Details

The rank epsilon squared and rank eta squared are appropriate for non-parametric tests of differences between 2 or more samples (a rank based ANOVA). See [stats::kruskal.test](#). Values range from 0 to 1, with larger values indicating larger differences between groups.

Kendall's W is appropriate for non-parametric tests of differences between 2 or more dependent samples (a rank based rmANOVA), where each group (e.g., experimental condition) was measured for each block (e.g., subject). This measure is also common as a measure of reliability of the rankings of the groups between raters (blocks). See [stats::friedman.test](#). Values range from 0 to 1, with larger values indicating larger differences between groups / higher agreement between raters.

Value

A data frame with the effect size and its CI.

Confidence (Compatibility) Intervals (CIs)

Confidence intervals for E_R^2 , η_H^2 , and Kendall's W are estimated using the bootstrap method (using the `{boot}` package).

CIs and Significance Tests

"Confidence intervals on measures of effect size convey all the information in a hypothesis test, and more." (Steiger, 2004). Confidence (compatibility) intervals and p values are complementary summaries of parameter uncertainty given the observed data. A dichotomous hypothesis test could be performed with either a CI or a p value. The $100(1 - \alpha)\%$ confidence interval contains all of the parameter values for which $p > \alpha$ for the current data and model. For example, a 95% confidence interval contains all of the values for which $p > .05$.

Note that a confidence interval including 0 *does not* indicate that the null (no effect) is true. Rather, it suggests that the observed data together with the model and its assumptions combined do not provided clear evidence against a parameter value of 0 (same as with any other value in the interval), with the level of this evidence defined by the chosen α level (Rafi & Greenland, 2020; Schweder & Hjort, 2016; Xie & Singh, 2013). To infer no effect, additional judgments about what parameter values are "close enough" to 0 to be negligible are needed ("equivalence testing"; Bauer & Kiessner, 1996).

Bootstrapped CIs

Some effect sizes are directionless—they do have a minimum value that would be interpreted as "no effect", but they cannot cross it. For example, a null value of [Kendall's W](#) is 0, indicating no difference between groups, but it can never have a negative value. Same goes for [U2](#) and [Overlap](#): the null value of U_2 is 0.5, but it can never be smaller than 0.5; an *Overlap* of 1 means "full overlap" (no difference), but it cannot be larger than 1.

When bootstrapping CIs for such effect sizes, the bounds of the CIs will never cross (and often will never cover) the null. Therefore, these CIs should not be used for statistical inference.

Ties

When tied values occur, they are each given the average of the ranks that would have been given had no ties occurred. This results in an effect size of reduced magnitude. A correction has been applied for Kendall's W .

References

- Kendall, M.G. (1948) Rank correlation methods. London: Griffin.
- Tomczak, M., & Tomczak, E. (2014). The need to report effect size estimates revisited. An overview of some recommended measures of effect size. Trends in sport sciences, 1(21), 19-25.

See Also

Other rank-based effect sizes: [p_superiority\(\)](#), [rank_biserial\(\)](#)

Other effect sizes for ANOVAs: [eta_squared\(\)](#)

Examples

```
# Rank Eta/Epsilon Squared
# =====

rank_eta_squared(mpg ~ cyl, data = mtcars)

rank_epsilon_squared(mpg ~ cyl, data = mtcars)

# Kendall's W
# =====
dat <- data.frame(
  cond = c("A", "B", "A", "B", "A", "B"),
  ID = c("L", "L", "M", "M", "H", "H"),
  y = c(44.56, 28.22, 24, 28.78, 24.56, 18.78)
)
(W <- kendalls_w(y ~ cond | ID, data = dat, verbose = FALSE))

interpret_kendalls_w(0.11)
interpret(W, rules = "landis1977")
```

RCT_table

Fictional Results from a Workers' Randomized Control Trial

Description

Fictional Results from a Workers' Randomized Control Trial

Format

A 2-by-2 table, with a *column* for each group and a *row* for the diagnosis.

```
data("RCT_table")
RCT_table
#>           Group
#> Diagnosis Treatment Control
#> Sick           71         30
#> Recovered      50        100
```

See Also

Other effect size datasets: [Music_preferences2](#), [Music_preferences](#), [Smoking_FASD](#), [food_class](#), [hardlyworking](#), [screening_test](#)

rules

Create an Interpretation Grid

Description

Create a container for interpretation rules of thumb. Usually used in conjunction with [interpret](#).

Usage

```
rules(values, labels = NULL, name = NULL, right = TRUE)
```

```
is.rules(x)
```

Arguments

values	Vector of reference values (edges defining categories or critical values).
labels	Labels associated with each category. If NULL, will try to infer it from values (if it is a named vector or a list), otherwise, will return the breakpoints.
name	Name of the set of rules (will be printed).
right	logical, for threshold-type rules, indicating if the thresholds themselves should be included in the interval to the right (lower values) or in the interval to the left (higher values).
x	An arbitrary R object.

See Also

[interpret\(\)](#)

Examples

```
rules(c(0.05), c("significant", "not significant"), right = FALSE)
rules(c(0.2, 0.5, 0.8), c("small", "medium", "large"))
rules(c("small" = 0.2, "medium" = 0.5), name = "Cohen's Rules")
```

screening_test	<i>Results from 2 Screening Tests</i>
----------------	---------------------------------------

Description

A sample (simulated) dataset, used in tests and some examples.

Format

A data frame with 1600 rows and 3 variables:

Diagnosis Ground truth

Test1 Results given by the 1st test

Test2 Results given by the 2nd test

```
data("screening_test")
head(screening_test, n = 5)
#>   Diagnosis Test1 Test2
#> 1      Neg "Neg" "Neg"
#> 2      Neg "Neg" "Neg"
#> 3      Neg "Neg" "Neg"
#> 4      Neg "Neg" "Neg"
#> 5      Neg "Neg" "Neg"
```

See Also

Other effect size datasets: [Music_preferences2](#), [Music_preferences](#), [RCT_table](#), [Smoking_FASD](#), [food_class](#), [hardlyworking](#)

sd_pooled	<i>Pooled Indices of (Co)Deviation</i>
-----------	--

Description

The Pooled Standard Deviation is a weighted average of standard deviations for two or more groups, *assumed to have equal variance*. It represents the common deviation among the groups, around each of their respective means.

Usage

```
sd_pooled(x, y = NULL, data = NULL, verbose = TRUE, ...)
```

```
mad_pooled(x, y = NULL, data = NULL, constant = 1.4826, verbose = TRUE, ...)
```

```
cov_pooled(x, y = NULL, data = NULL, verbose = TRUE, ...)
```

Arguments

<code>x, y</code>	A numeric vector, or a character name of one in <code>data</code> . Any missing values (NAs) are dropped from the resulting vector. <code>x</code> can also be a formula (see <code>stats::t.test()</code>), in which case <code>y</code> is ignored.
<code>data</code>	An optional data frame containing the variables.
<code>verbose</code>	Toggle warnings and messages on or off.
<code>...</code>	Arguments passed to or from other methods. When <code>x</code> is a formula, these can be <code>subset</code> and <code>na.action</code> .
<code>constant</code>	scale factor.

Details

The standard version is calculated as:

$$\sqrt{\frac{\sum (x_i - \bar{x})^2}{n_1 + n_2 - 2}}$$

The robust version is calculated as:

$$1.4826 \times \text{Median}(|\{x - \text{Median}_x, y - \text{Median}_y\}|)$$

Value

Numeric, the pooled standard deviation. For `cov_pooled()` a matrix.

See Also

[cohens_d\(\)](#), [mahalanobis_d\(\)](#)

Examples

```
sd_pooled(mpg ~ am, data = mtcars)
mad_pooled(mtcars$mpg, factor(mtcars$am))

cov_pooled(mpg + hp + cyl ~ am, data = mtcars)
```

Smoking_FASD

Frequency of FASD for Smoking Mothers

Description

Fictional data.

Format

A 1-by-3 table, with a *column* for each diagnosis.

```
data("Smoking_FASD")
Smoking_FASD
#>   FAS PFAS  TD
#>   17   11 640
```

See Also

Other effect size datasets: [Music_preferences2](#), [Music_preferences](#), [RCT_table](#), [food_class](#), [hardlyworking](#), [screening_test](#)

t_to_d

*Convert t, z, and F to Cohen's d or **partial-r***

Description

These functions are convenience functions to convert t, z and F test statistics to Cohen's d and **partial r**. These are useful in cases where the data required to compute these are not easily available or their computation is not straightforward (e.g., in liner mixed models, contrasts, etc.).

See [Effect Size from Test Statistics vignette](#).

Usage

```
t_to_d(t, df_error, paired = FALSE, ci = 0.95, alternative = "two.sided", ...)
```

```
z_to_d(z, n, paired = FALSE, ci = 0.95, alternative = "two.sided", ...)
```

```
F_to_d(
  f,
  df,
  df_error,
  paired = FALSE,
  ci = 0.95,
  alternative = "two.sided",
  ...
)
```

```
t_to_r(t, df_error, ci = 0.95, alternative = "two.sided", ...)
```

```
z_to_r(z, n, ci = 0.95, alternative = "two.sided", ...)
```

```
F_to_r(f, df, df_error, ci = 0.95, alternative = "two.sided", ...)
```

Arguments

t, f, z	The t, the F or the z statistics.
paired	Should the estimate account for the t-value being testing the difference between dependent means?
ci	Confidence Interval (CI) level
alternative	a character string specifying the alternative hypothesis; Controls the type of CI returned: "two.sided" (default, two-sided CI), "greater" or "less" (one-sided CI). Partial matching is allowed (e.g., "g", "l", "two"...). See <i>One-Sided CIs</i> in effectsize_CIs .
...	Arguments passed to or from other methods.
n	The number of observations (the sample size).
df, df_error	Degrees of freedom of numerator or of the error estimate (i.e., the residuals).

Details

These functions use the following formulae to approximate r and d :

$$r_{\text{partial}} = t / \sqrt{t^2 + df_{\text{error}}}$$

$$r_{\text{partial}} = z / \sqrt{z^2 + N}$$

$$d = 2 * t / \sqrt{df_{\text{error}}}$$

$$d_z = t / \sqrt{df_{\text{error}}}$$

$$d = 2 * z / \sqrt{N}$$

The resulting d effect size is an *approximation* to Cohen's d , and assumes two equal group sizes. When possible, it is advised to directly estimate Cohen's d , with [cohens_d\(\)](#), `emmeans::eff_size()`, or similar functions.

Value

A data frame with the effect size(s) (r or d), and their CIs (`CI_low` and `CI_high`).

Confidence (Compatibility) Intervals (CIs)

Unless stated otherwise, confidence (compatibility) intervals (CIs) are estimated using the non-centrality parameter method (also called the "pivot method"). This method finds the noncentrality parameter ("*npc*") of a noncentral *t*, *F*, or χ^2 distribution that places the observed *t*, *F*, or χ^2 test statistic at the desired probability point of the distribution. For example, if the observed *t* statistic is 2.0, with 50 degrees of freedom, for which cumulative noncentral *t* distribution is *t* = 2.0 the .025 quantile (answer: the noncentral *t* distribution with *npc* = .04)? After estimating these confidence bounds on the *npc*, they are converted into the effect size metric to obtain a confidence interval for the effect size (Steiger, 2004).

For additional details on estimation and troubleshooting, see [effectsize_CIs](#).

CIs and Significance Tests

"Confidence intervals on measures of effect size convey all the information in a hypothesis test, and more." (Steiger, 2004). Confidence (compatibility) intervals and p values are complementary summaries of parameter uncertainty given the observed data. A dichotomous hypothesis test could be performed with either a CI or a p value. The 100 (1 - α)% confidence interval contains all of the parameter values for which $p > \alpha$ for the current data and model. For example, a 95% confidence interval contains all of the values for which $p > .05$.

Note that a confidence interval including 0 *does not* indicate that the null (no effect) is true. Rather, it suggests that the observed data together with the model and its assumptions combined do not provide clear evidence against a parameter value of 0 (same as with any other value in the interval), with the level of this evidence defined by the chosen α level (Rafi & Greenland, 2020; Schweder & Hjort, 2016; Xie & Singh, 2013). To infer no effect, additional judgments about what parameter values are "close enough" to 0 to be negligible are needed ("equivalence testing"; Bauer & Kiesser, 1996).

References

- Friedman, H. (1982). Simplified determinations of statistical power, magnitude of effect and research sample sizes. *Educational and Psychological Measurement*, 42(2), 521-526. [doi:10.1177/001316448204200214](https://doi.org/10.1177/001316448204200214)
- Wolf, F. M. (1986). *Meta-analysis: Quantitative methods for research synthesis* (Vol. 59). Sage.
- Rosenthal, R. (1994) Parametric measures of effect size. In H. Cooper and L.V. Hedges (Eds.). *The handbook of research synthesis*. New York: Russell Sage Foundation.
- Steiger, J. H. (2004). Beyond the F test: Effect size confidence intervals and tests of close fit in the analysis of variance and contrast analysis. *Psychological Methods*, 9, 164-182.
- Cumming, G., & Finch, S. (2001). A primer on the understanding, use, and calculation of confidence intervals that are based on central and noncentral distributions. *Educational and Psychological Measurement*, 61(4), 532-574.

See Also

[cohens_d\(\)](#)

Other effect size from test statistic: [F_to_eta2\(\)](#), [chisq_to_phi\(\)](#)

Examples

```
## t Tests
res <- t.test(1:10, y = c(7:20), var.equal = TRUE)
t_to_d(t = res$statistic, res$parameter)
t_to_r(t = res$statistic, res$parameter)
t_to_r(t = res$statistic, res$parameter, alternative = "less")

res <- with(sleep, t.test(extra[group == 1], extra[group == 2], paired = TRUE))
t_to_d(t = res$statistic, res$parameter, paired = TRUE)
t_to_r(t = res$statistic, res$parameter)
t_to_r(t = res$statistic, res$parameter, alternative = "greater")

## Linear Regression
model <- lm(rating ~ complaints + critical, data = attitude)
(param_tab <- parameters::model_parameters(model))

(rs <- t_to_r(param_tab$t[2:3], param_tab$df_error[2:3]))

if (require(see)) plot(rs)

# How does this compare to actual partial correlations?
if (require("correlation")) {
  correlation::correlation(attitude[, c(1, 2, 6)], partial = TRUE)[1:2, c(2, 3, 7, 8)]
}
```

Index

- * **convert between effect sizes**
 - d_to_r, 15
 - diff_to_cles, 13
 - eta2_to_f2, 27
 - odds_to_probs, 68
 - oddsratio_to_riskratio, 67
- * **data**
 - food_class, 34
 - hardlyworking, 39
 - Music_preferences, 64
 - Music_preferences2, 65
 - RCT_table, 85
 - screening_test, 87
 - Smoking_FASD, 88
- * **effect size datasets**
 - food_class, 34
 - hardlyworking, 39
 - Music_preferences, 64
 - Music_preferences2, 65
 - RCT_table, 85
 - screening_test, 87
 - Smoking_FASD, 88
- * **effect size from test statistic**
 - chisq_to_phi, 3
 - F_to_eta2, 35
 - t_to_d, 89
- * **effect size indices**
 - rank_biserial, 79
- * **effect sizes for ANOVAs**
 - eta_squared, 28
 - rank_epsilon_squared, 82
- * **effect sizes for contingency table**
 - cohens_g, 11
 - oddsratio, 65
 - phi, 69
- * **interpreters**
 - interpret_bf, 42
 - interpret_cohens_d, 43
 - interpret_cohens_g, 45
 - interpret_direction, 46
 - interpret_ess, 46
 - interpret_gfi, 47
 - interpret_icc, 50
 - interpret_kendalls_w, 51
 - interpret_oddsratio, 52
 - interpret_omega_squared, 53
 - interpret_p, 54
 - interpret_pd, 55
 - interpret_r, 56
 - interpret_r2, 58
 - interpret_rope, 59
 - interpret_vif, 60
- * **rank-based effect sizes**
 - p_superiority, 74
 - rank_biserial, 79
 - rank_epsilon_squared, 82
- * **standardized differences**
 - cohens_d, 7
 - mahalanobis_d, 61
 - p_superiority, 74
 - rank_biserial, 79
 - .es_aov_simple (effectsize_API), 19
 - .es_aov_strata (effectsize_API), 19
 - .es_aov_table (effectsize_API), 19
- anova(), 30
- bayestestR::describe_posterior(), 17, 31
- bayestestR::equivalence_test(), 26
- bayestestR::overlap(), 77
- chisq_to_cohens_w (chisq_to_phi), 3
- chisq_to_cramers_v (chisq_to_phi), 3
- chisq_to_fei, 24
- chisq_to_fei (chisq_to_phi), 3
- chisq_to_normalized (effectsize_deprecated), 23
- chisq_to_pearsons_c (chisq_to_phi), 3

- chisq_to_phi, [3](#), [38](#), [91](#)
- chisq_to_phi(), [72](#)
- chisq_to_tschuprows_t (chisq_to_phi), [3](#)
- cles (p_superiority), [74](#)
- cliffs_delta (rank_biserial), [79](#)
- cohens_d, [7](#), [63](#), [78](#), [81](#)
- cohens_d(), [13](#), [14](#), [16](#), [25](#), [76](#), [88](#), [90](#), [91](#)
- cohens_f (eta_squared), [28](#)
- cohens_f_squared (eta_squared), [28](#)
- cohens_g, [11](#), [67](#), [72](#)
- cohens_h (oddsratio), [65](#)
- cohens_u1 (p_superiority), [74](#)
- cohens_u2 (p_superiority), [74](#)
- cohens_u3 (p_superiority), [74](#)
- cohens_u3(), [14](#)
- cohens_w (phi), [69](#)
- Common Language Effect Sizes, [79](#)
- common_language
 - (effectsize_deprecated), [23](#)
- convert_d_to_common_language
 - (effectsize_deprecated), [23](#)
- convert_d_to_oddsratio (d_to_r), [15](#)
- convert_d_to_r (d_to_r), [15](#)
- convert_logoddsratio_to_d (d_to_r), [15](#)
- convert_logoddsratio_to_r (d_to_r), [15](#)
- convert_odds_to_probs (odds_to_probs), [68](#)
- convert_oddsratio_to_d (d_to_r), [15](#)
- convert_oddsratio_to_r (d_to_r), [15](#)
- convert_probs_to_odds (odds_to_probs), [68](#)
- convert_r_to_d (d_to_r), [15](#)
- convert_r_to_oddsratio (d_to_r), [15](#)
- convert_rb_to_common_language
 - (effectsize_deprecated), [23](#)
- cov_pooled (sd_pooled), [87](#)
- cov_pooled(), [63](#)
- cramers_v (phi), [69](#)
- cramers_v(), [5](#)

- d_to_cles, [74](#)
- d_to_cles (diff_to_cles), [13](#)
- d_to_common_language
 - (effectsize_deprecated), [23](#)
- d_to_oddsratio (d_to_r), [15](#)
- d_to_overlap (diff_to_cles), [13](#)
- d_to_p_superiority (diff_to_cles), [13](#)
- d_to_r, [14](#), [15](#), [28](#), [68](#), [69](#)
- d_to_r(), [44](#)
- d_to_u1, [24](#)
- d_to_u1 (diff_to_cles), [13](#)
- d_to_u2 (diff_to_cles), [13](#)
- d_to_u3 (diff_to_cles), [13](#)
- d_to_u3(), [77](#)
- diff_to_cles, [13](#), [16](#), [28](#), [68](#), [69](#)

- effectsize (effectsize.BFBayesFactor), [16](#)
- effectsize.BFBayesFactor, [16](#)
- effectsize_API, [19](#)
- effectsize_CIs, [5](#), [8](#), [9](#), [12](#), [20](#), [20](#), [29](#), [31](#), [36](#), [37](#), [62](#), [63](#), [66](#), [70](#), [71](#), [76](#), [80](#), [83](#), [90](#), [91](#)
- effectsize_deprecated, [23](#)
- effectsize_options, [24](#), [61](#), [74](#)
- epsilon_squared (eta_squared), [28](#)
- equivalence_test.effectsize_table, [25](#)
- eta2_to_f (eta2_to_f2), [27](#)
- eta2_to_f2, [14](#), [16](#), [27](#), [68](#), [69](#)
- eta_squared, [28](#), [85](#)
- eta_squared(), [20](#), [25](#), [28](#), [38](#)
- eta_squared_posterior (eta_squared), [28](#)

- f2_to_eta2 (eta2_to_f2), [27](#)
- F_to_d (t_to_d), [89](#)
- F_to_epsilon2 (F_to_eta2), [35](#)
- F_to_eta2, [6](#), [35](#), [91](#)
- f_to_eta2 (eta2_to_f2), [27](#)
- F_to_eta2(), [30](#), [32](#)
- F_to_eta2_adj (F_to_eta2), [35](#)
- F_to_f (F_to_eta2), [35](#)
- F_to_f2 (F_to_eta2), [35](#)
- F_to_omega2 (F_to_eta2), [35](#)
- F_to_r (t_to_d), [89](#)
- F_to_r(), [25](#)
- fei, [24](#)
- fei (phi), [69](#)
- fitmeasures(), [49](#)
- food_class, [34](#), [40](#), [65](#), [86](#), [87](#), [89](#)
- format.effectsize_table
 - (plot.effectsize_table), [73](#)
- format_standardize, [34](#)

- get_effectsize_label
 - (is_effectsize_name), [61](#)
- get_effectsize_name
 - (is_effectsize_name), [61](#)
- glass_delta (cohens_d), [7](#)

- hardlyworking, [34](#), [39](#), [65](#), [86](#), [87](#), [89](#)
- hedges_g (cohens_d), [7](#)
- insight::display(), [74](#)
- insight::format_value(), [35](#)
- insight::get_data(), [17](#)
- interpret, [40](#), [86](#)
- interpret(), [86](#)
- interpret.lavaan (interpret_gfi), [47](#)
- interpret.performance.lavaan (interpret_gfi), [47](#)
- interpret_agfi (interpret_gfi), [47](#)
- interpret_bf, [42](#)
- interpret_cfi (interpret_gfi), [47](#)
- interpret_cohens_d, [24](#), [43](#)
- interpret_cohens_d(), [52](#)
- interpret_cohens_g, [45](#)
- interpret_cramers_v (interpret_r), [56](#)
- interpret_d (effectsize_deprecated), [23](#)
- interpret_delta (effectsize_deprecated), [23](#)
- interpret_direction, [46](#)
- interpret_epsilon_squared (interpret_omega_squared), [53](#)
- interpret_ess, [46](#)
- interpret_eta_squared (interpret_omega_squared), [53](#)
- interpret_g (effectsize_deprecated), [23](#)
- interpret_gfi, [47](#)
- interpret_glass_delta, [24](#)
- interpret_glass_delta (interpret_cohens_d), [43](#)
- interpret_hedges_g, [24](#)
- interpret_hedges_g (interpret_cohens_d), [43](#)
- interpret_icc, [50](#)
- interpret_ifi (interpret_gfi), [47](#)
- interpret_kendalls_w, [51](#)
- interpret_nfi (interpret_gfi), [47](#)
- interpret_nnfi (interpret_gfi), [47](#)
- interpret_oddsratio, [52](#)
- interpret_omega_squared, [53](#)
- interpret_p, [54](#)
- interpret_parameters (effectsize_deprecated), [23](#)
- interpret_pd, [55](#)
- interpret_phi (interpret_r), [56](#)
- interpret_pnfi (interpret_gfi), [47](#)
- interpret_r, [24](#), [56](#)
- interpret_r(), [44](#)
- interpret_r2, [58](#)
- interpret_rank_biserial (interpret_r), [56](#)
- interpret_rfi (interpret_gfi), [47](#)
- interpret_rhat (interpret_ess), [46](#)
- interpret_rmsea (interpret_gfi), [47](#)
- interpret_rope, [59](#)
- interpret_srmr (interpret_gfi), [47](#)
- interpret_vif, [60](#)
- is.rules(rules), [86](#)
- is_effectsize_name, [61](#)
- Kendall's W, [21](#), [78](#), [84](#)
- kendalls_w (rank_epsilon_squared), [82](#)
- logical, [24](#)
- logoddsratio_to_d (d_to_r), [15](#)
- logoddsratio_to_r (d_to_r), [15](#)
- mad_pooled (sd_pooled), [87](#)
- mahalanobis_d, [10](#), [61](#), [78](#), [81](#)
- mahalanobis_d(), [88](#)
- Music_preferences, [34](#), [40](#), [64](#), [65](#), [86](#), [87](#), [89](#)
- Music_preferences2, [34](#), [40](#), [65](#), [65](#), [86](#), [87](#), [89](#)
- normalized_chi (effectsize_deprecated), [23](#)
- odds_to_probs, [14](#), [16](#), [28](#), [68](#), [68](#)
- oddsratio, [12](#), [65](#), [72](#)
- oddsratio(), [68](#)
- oddsratio_to_d (d_to_r), [15](#)
- oddsratio_to_d(), [52](#)
- oddsratio_to_r (d_to_r), [15](#)
- oddsratio_to_riskratio, [14](#), [16](#), [28](#), [67](#), [69](#)
- omega_squared (eta_squared), [28](#)
- Overlap, [21](#), [78](#), [84](#)
- p_direction(), [17](#)
- p_overlap (p_superiority), [74](#)
- p_superiority, [10](#), [63](#), [74](#), [81](#), [85](#)
- parameters::standardize_parameters(), [17](#)
- pearsons_c (phi), [69](#)
- phi, [12](#), [67](#), [69](#)
- phi(), [6](#)
- phi_to_chisq (chisq_to_phi), [3](#)
- plot.effectsize_table, [73](#)

- print.effectsize_difference
(plot.effectsize_table), 73
- print.effectsize_table
(plot.effectsize_table), 73
- print_html.effectsize_table
(plot.effectsize_table), 73
- print_md.effectsize_table
(plot.effectsize_table), 73
- probs_to_odds (odds_to_probs), 68
- probs_to_odds(), 77
- r_to_d (d_to_r), 15
- r_to_d(), 10
- r_to_oddsratio (d_to_r), 15
- rank_biserial, 10, 63, 78, 79, 85
- rank_biserial(), 13, 14, 76
- rank_epsilon_squared, 32, 78, 81, 82
- rank_eta_squared
(rank_epsilon_squared), 82
- rb_to_cles (diff_to_cles), 13
- rb_to_common_language
(effectsize_deprecated), 23
- rb_to_p_superiority (diff_to_cles), 13
- rb_to_p_superiority(), 77
- rb_to_vda (diff_to_cles), 13
- rb_to_wmw_odds (diff_to_cles), 13
- RCT_table, 34, 40, 65, 85, 87, 89
- riskratio (oddsratio), 65
- riskratio(), 68
- riskratio_to_oddsratio
(oddsratio_to_riskratio), 67
- rope(), 17
- rstantools::posterior_predict(), 31
- rules, 86
- rules(), 40–43, 45, 46, 48, 50–56, 58–60
- screening_test, 34, 40, 65, 86, 87, 89
- sd_pooled, 87
- sd_pooled(), 8, 10, 78
- signif(), 35, 73
- Smoking_FASD, 34, 40, 65, 86, 87, 88
- stats::chisq.test(), 5, 65, 69
- stats::fisher.test(), 65
- stats::friedman.test, 84
- stats::friedman.test(), 82
- stats::kruskal.test, 84
- stats::kruskal.test(), 82
- stats::mahalanobis(), 63
- stats::mcnemar.test(), 11
- stats::plogis(), 69
- stats::prop.test(), 12
- stats::t.test(), 7, 8, 74, 76, 88
- stats::wilcox.test, 80
- stats::wilcox.test(), 74, 79
- t_to_d, 6, 38, 89
- t_to_d(), 10
- t_to_epsilon2 (F_to_eta2), 35
- t_to_eta2 (F_to_eta2), 35
- t_to_eta2_adj (F_to_eta2), 35
- t_to_f (F_to_eta2), 35
- t_to_f2 (F_to_eta2), 35
- t_to_omega2 (F_to_eta2), 35
- t_to_r (t_to_d), 89
- tschuprows_t (phi), 69
- U2, 21, 78, 84
- vd_a (p_superiority), 74
- wmw_odds (p_superiority), 74
- z_to_d (t_to_d), 89
- z_to_r (t_to_d), 89