

Package ‘TOSTER’

April 5, 2023

Version 0.7.1

Title Two One-Sided Tests (TOST) Equivalence Testing

Description Two one-sided tests (TOST) procedure to test equivalence for t-tests, correlations, differences between proportions, and meta-analyses, including power analysis for t-tests and correlations. Allows you to specify equivalence bounds in raw scale units or in terms of effect sizes. See: Lakens (2017) <[doi:10.1177/1948550617697177](https://doi.org/10.1177/1948550617697177)>.

Maintainer Aaron Caldwell <aracaldwell149@gmail.com>

URL <https://aaroncaldwell.us/TOSTERpkg/>

License GPL-3

RoxygenNote 7.2.3

Imports stats, graphics, jmvcore (>= 0.9.6.4), ggplot2, ggdist, distributional, cowplot, tidyr, utils, R6

Suggests knitr, rmarkdown, broom, car, afex, testthat (>= 3.0.0)

VignetteBuilder knitr

Encoding UTF-8

LazyData true

Config/testthat/edition 3

Depends R (>= 3.5)

NeedsCompilation no

Author Daniel Lakens [aut],
Aaron Caldwell [aut, cre]

Repository CRAN

Date/Publication 2023-04-05 16:43:28 UTC

R topics documented:

as_hstest	3
boot_compare_cor	4
boot_compare_smd	5
boot_cor_test	6

boot_log_TOST	8
boot_t_test	10
boot_t_TOST	12
compare_cor	14
compare_smd	15
corsum_test	17
dataTOSTone	18
dataTOSTpaired	20
dataTOSTr	22
dataTOSTtwo	23
datatosttwoprop	25
equ_anova	26
equ_ftest	27
extract_r_paired	28
hawthorne	29
htest-helpers	29
log_TOST	30
plot_cor	32
plot_pes	33
plot_smd	34
powerTOSTone	35
powerTOSTone.raw	36
powerTOSTpaired	37
powerTOSTpaired.raw	38
powerTOSTr	40
powerTOSTtwo	41
powerTOSTtwo.prop	42
powerTOSTtwo.raw	43
power_eq_f	45
power_t_TOST	46
rbs	47
ses_calc	49
simple_htest	50
smd_calc	52
TOSTmeta	54
TOSTnp-methods	55
TOSTone	56
TOSTone.raw	57
TOSTpaired	58
TOSTpaired.raw	60
TOSTr	61
TOSTt-methods	62
TOSTtwo	63
TOSTtwo.prop	65
TOSTtwo.raw	66
tsum_TOST	68
t_TOST	70
wilcox_TOST	73

<code>as_htest</code>	3
<code>z_cor_test</code>	75
Index	77

<code>as_htest</code>	<i>Convert to class 'htest'</i>
-----------------------	---------------------------------

Description

Convert a TOSTER result object of class 'TOSTt' or 'TOSTnp' to a list of class 'htest'.

Usage

```
as_htest(TOST)
```

Arguments

TOST A TOSTER result object of class 'TOSTt' or 'TOSTnp'.

Value

Returns a list containing a list of class 'htest' for the result of each test with the following elements:

- `data.name` A character string giving the names of the data.
- `estimate` Estimated difference in raw units.
- `method` A character string indicating the performed test.
- `null.value` Equivalence bound.
- `alternative` A character string describing the alternative hypothesis.
- `parameter` The degrees of freedom of the distribution of the test statistic.
- `statistic` The value of the test statistic.
- `p.value` The p-value of the test.
- `conf.int` The confidence interval of the difference.

See Also

Other htest: [htest-helpers](#), [simple_htest\(\)](#)

Examples

```
res1 = t_TOST(formula = extra ~ group,data = sleep,eqb = .5,smd_ci = "goulet")
as_htest(res1)
```

boot_compare_cor	<i>Comparing Correlations between independent studies with Bootstrapping</i>
------------------	--

Description

A function to compare standardized mean differences (SMDs) between studies. This function is intended to be used to compare the compatibility of original studies with replication studies (lower p-values indicating lower compatibility)

Usage

```
boot_compare_cor(
  x1,
  y1,
  x2,
  y2,
  alternative = c("two.sided", "less", "greater", "equivalence", "minimal.effect"),
  method = c("pearson", "kendall", "spearman", "winsorized", "bendpercent"),
  alpha = 0.05,
  null = 0,
  R = 1999,
  ...
)
```

Arguments

x1, y1	numeric vectors of data values. x and y must have the same length from study 1.
x2, y2	numeric vectors of data values. x and y must have the same length from study 2.
alternative	a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "greater", "less", "equivalence" (TOST), or "minimal.effect" (TOST). You can specify just the initial letter.
method	a character string indicating which correlation coefficient is to be used for the test. One of "winsorized", "bendpercent", "pearson", "kendall", or "spearman", can be abbreviated.
alpha	alpha level (default = 0.05)
null	a number indicating the null hypothesis. Default is a correlation of zero.
R	number of bootstrap replicates
...	further arguments to be passed to or from methods.

Value

A list with class "htest" containing the following components:

"p.value" numeric scalar containing the p-value for the test under the null hypothesis.

"estimate" difference in correlations between studies.
 "conf.int" percentile (bootstrap) confidence interval for difference in correlations.
 "null.value" the specified hypothesized value for the null hypothesis.
 "alternative" character string indicating the alternative hypothesis (the value of the input argument alternative). Possible values are "greater", "less", or "two-sided".
 "method" a character string indicating how the association was measured.
 "data.name" Names of input values..
 "boot_res" List of bootstrapped results.
 "call" the matched call.

boot_compare_smd	<i>Comparing SMDs between independent studies with Bootstrapping</i>
------------------	--

Description

A function to compare standardized mean differences (SMDs) between studies. This function is intended to be used to compare the compatibility of original studies with replication studies (lower p-values indicating lower compatibility)

Usage

```
boot_compare_smd(
  x1,
  y1 = NULL,
  x2,
  y2 = NULL,
  null = 0,
  paired = FALSE,
  alternative = c("two.sided", "less", "greater"),
  R = 1999,
  alpha = 0.05
)
```

Arguments

x1	a (non-empty) numeric vector of data values from study 1.
y1	an optional (non-empty) numeric vector of data values from study 1.
x2	a (non-empty) numeric vector of data values from study 2.
y2	an optional (non-empty) numeric vector of data values from study 2.
null	a number indicating the null hypothesis. For TOST, this would be equivalence bound.
paired	a logical indicating whether the SMD is from a paired or independent samples design.

alternative a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "greater" or "less". You can specify just the initial letter.
R number of bootstrap replicates
alpha alpha level (default = 0.05)

Value

A list with class "htest" containing the following components:

"statistic" z-score
"p.value" numeric scalar containing the p-value for the test under the null hypothesis.
"estimate" difference in SMD between studies
"conf.int" percentile (bootstrap) confidence interval for difference in SMDs
"null.value" the specified hypothesized value for the null hypothesis.
"alternative" character string indicating the alternative hypothesis (the value of the input argument alternative). Possible values are "greater", "less", or "two-sided".
"method" Type of SMD
"data.name" "Bootstrapped" to denote summary statistics were utilized to obtain results.
"smd" SMDs input for the function.
"df_ci" Data frame of confidence intervals.
"boot_res" List of bootstrapped results.
"call" the matched call.

See Also

Other compare studies: [compare_smd\(\)](#)

boot_cor_test

Bootstrapped correlation coefficients

Description

A function for a bootstrap, percentile, method for correlation coefficients.

Usage

```

boot_cor_test(
  x,
  y,
  alternative = c("two.sided", "less", "greater", "equivalence", "minimal.effect"),
  method = c("pearson", "kendall", "spearman", "winsorized", "bendpercent"),
  alpha = 0.05,
  null = 0,
  R = 1999,
  ...
)
  
```

Arguments

x	a (non-empty) numeric vector of data values.
y	an optional (non-empty) numeric vector of data values.
alternative	a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "greater", "less", "equivalence" (TOST), or "minimal.effect" (TOST). You can specify just the initial letter.
method	a character string indicating which correlation coefficient is to be used for the test. One of "winsorized", "bendpercent", "pearson", "kendall", or "spearman", can be abbreviated.
alpha	alpha level (default = 0.05)
null	a number indicating the null hypothesis. Default is a correlation of zero.
R	number of bootstrap replicates
...	further arguments to be passed to or from methods.

Details

This function uses a percentile bootstrap methods for the confidence intervals. The returned p-values are calculated from a re-sampled null distribution (similar to `boot_t_TOST`).

The bootstrap correlation methods in this package offer two other correlations: a Winsorized correlation and a percentage bend correlation (see Wilcox 2017). These two can be modified by adding the `trim` (Winsorized) or `beta` (percentage bend) arguments. The default for both arguments is 0.2 but can be modified at the user's discretion. These calculations are based on Rand Wilcox's R functions for his book (Wilcox, 2017), and adapted from their implementation in Guillaume Rousselet's R package "bootcorci".

Value

A list with class "htest" containing the following components:

- "p.value" the p-value of the test.
- "estimate" the estimated measure of association, with name "pb", "wincor", "cor", "tau", or "rho" corresponding to the method employed.
- "null.value" the value of the association measure under the null hypothesis.
- "alternative" character string indicating the alternative hypothesis (the value of the input argument `alternative`).
- "method" a character string indicating how the association was measured.
- "data.name" a character string giving the names of the data.
- "call" the matched call.

References

- Wilcox, R.R. (2009) Comparing Pearson Correlations: Dealing with Heteroscedasticity and Non-normality. *Communications in Statistics - Simulation and Computation*, 38, 2220–2234.
- Wilcox, R.R. (2017) *Introduction to Robust Estimation and Hypothesis Testing*, 4th edition. Academic Press.

boot_log_TOST

*Bootstrapped TOST with log transformed t-tests***Description**

A function for a bootstrap method for TOST with all types of t-tests.

Usage

```
boot_log_TOST(x, ...)

## Default S3 method:
boot_log_TOST(
  x,
  y = NULL,
  hypothesis = c("EQU", "MET"),
  paired = FALSE,
  var.equal = FALSE,
  eqb = 1.25,
  alpha = 0.05,
  null = 1,
  R = 1999,
  ...
)

## S3 method for class 'formula'
boot_log_TOST(formula, data, subset, na.action, ...)
```

Arguments

x	a (non-empty) numeric vector of data values.
...	further arguments to be passed to or from methods.
y	an optional (non-empty) numeric vector of data values.
hypothesis	'EQU' for equivalence (default), or 'MET' for minimal effects test, the alternative hypothesis.
paired	a logical indicating whether you want a paired t-test.
var.equal	a logical variable indicating whether to treat the two variances as being equal. If TRUE then the pooled variance is used to estimate the variance otherwise the Welch (or Satterthwaite) approximation to the degrees of freedom is used.
eqb	Equivalence bound. Can provide 1 value (negative value is taken as the lower bound) or 2 specific values that represent the upper and lower equivalence bounds.
alpha	alpha level (default = 0.05)
null	Null hypothesis value for a two-tailed test (default is 1).
R	number of bootstrap replicates

formula	a formula of the form lhs ~ rhs where lhs is a numeric variable giving the data values and rhs either 1 for a one-sample or paired test or a factor with two levels giving the corresponding groups. If lhs is of class "Pair" and rhs is 1, a paired test is done.
data	an optional matrix or data frame (or similar: see model.frame) containing the variables in the formula formula. By default the variables are taken from environment(formula).
subset	an optional vector specifying a subset of observations to be used.
na.action	a function which indicates what should happen when the data contain NAs. Defaults to getOption("na.action").

Details

For details on the calculations in this function see vignette("robustTOST").

The implemented test(s) corresponds to the proposal of Chapter 16 of Efron and Tibshirani (1994), and recommended by He et al (2022). Returns TOSTt class object with bootstrapped based results. This is approximately equivalent to the percentile bootstrap method mentioned by He et al (2014).

For details on the calculations in this function see vignette("robustTOST").

Value

An S3 object of class "TOSTt" is returned containing the following slots:

"TOST" A table of class "data.frame" containing two-tailed t-test and both one-tailed results.

"eqb" A table of class "data.frame" containing equivalence bound settings.

"effsize" table of class "data.frame" containing effect size estimates

"hypothesis" String stating the hypothesis being tested

"smd" List containing the results of the means ratio calculation. Items include: d (means ratio estimate), dlow (lower CI bound), dhigh (upper CI bound), d_df (degrees of freedom for SMD), d_sigma (SE), d_lambda (non-centrality), J (bias correction), smd_label (type of SMD), d_denom (denominator calculation)

"alpha" Alpha level set for the analysis.

"method" Type of t-test.

"decision" List included text regarding the decisions for statistical inference.

"boot" List containing the bootstrap samples.

References

Efron, B., & Tibshirani, R. J. (1994). An introduction to the bootstrap. CRC press

He, Y., Deng, Y., You, C., & Zhou, X. H. (2022). Equivalence tests for ratio of means in bioequivalence studies under crossover design. *Statistical Methods in Medical Research*, 09622802221093721.

Food and Drug Administration (2014). Bioavailability and Bioequivalence Studies Submitted in NDAs or INDs — General Considerations. Center for Drug Evaluation and Research. Docket: FDA-2014-D-0204. <https://www.fda.gov/regulatory-information/search-fda-guidance-documents/bioavailability-and-bioequivalence-studies-submitted-ndas-or-inds-general-considerations>

See Also

Other Robust TOST: [boot_t_TOST\(\)](#), [boot_t_test\(\)](#), [log_TOST\(\)](#), [wilcox_TOST\(\)](#)

boot_t_test	<i>Bootstrapped t-test</i>
-------------	----------------------------

Description

A function for a bootstrap method for Tt-tests.

Usage

```
boot_t_test(x, ...)

## Default S3 method:
boot_t_test(
  x,
  y = NULL,
  var.equal = FALSE,
  paired = FALSE,
  alternative = c("two.sided", "less", "greater", "equivalence", "minimal.effect"),
  mu = 0,
  alpha = 0.05,
  R = 1999,
  ...
)

## S3 method for class 'formula'
boot_t_test(formula, data, subset, na.action, ...)
```

Arguments

x	a (non-empty) numeric vector of data values.
...	further arguments to be passed to or from methods.
y	an optional (non-empty) numeric vector of data values.
var.equal	a logical variable indicating whether to treat the two variances as being equal. If TRUE then the pooled variance is used to estimate the variance otherwise the Welch (or Satterthwaite) approximation to the degrees of freedom is used.
paired	a logical indicating whether you want a paired t-test.
alternative	a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "greater", "less", "equivalence" (TOST), or "minimal.effect" (TOST). You can specify just the initial letter.
mu	a number indicating the true value of the mean for the two tailed test (or difference in means if you are performing a two sample test).

alpha	alpha level (default = 0.05)
R	number of bootstrap replicates
formula	a formula of the form lhs ~ rhs where lhs is a numeric variable giving the data values and rhs either 1 for a one-sample or paired test or a factor with two levels giving the corresponding groups. If lhs is of class "Pair" and rhs is 1, a paired test is done.
data	an optional matrix or data frame (or similar: see model.frame) containing the variables in the formula formula. By default the variables are taken from environment(formula).
subset	an optional vector specifying a subset of observations to be used.
na.action	a function which indicates what should happen when the data contain NAs. Defaults to getOption("na.action").

Details

For details on the calculations in this function see vignette("robustTOST").

The implemented test(s) corresponds to the proposal of Chapter 16 of Efron and Tibshirani (1994).

For details on the calculations in this function see vignette("robustTOST").

Value

A list with class "htest" containing the following components:

statistic the value of the t-statistic.

parameter the degrees of freedom for the t-statistic.

p.value the p-value for the test.

conf.int a confidence interval for the mean appropriate to the specified alternative hypothesis.

estimate the estimated mean or difference in means depending on whether it was a one-sample test or a two-sample test.

null.value the specified hypothesized value of the mean or mean difference. May be 2 values.

stderr the standard error of the mean (difference), used as denominator in the t-statistic formula.

alternative a character string describing the alternative hypothesis.

method a character string indicating what type of t-test was performed.

data.name a character string giving the name(s) of the data..

References

Efron, B., & Tibshirani, R. J. (1994). An introduction to the bootstrap. CRC press.

See Also

Other Robust TOST: [boot_log_TOST\(\)](#), [boot_t_TOST\(\)](#), [log_TOST\(\)](#), [wilcox_TOST\(\)](#)

Examples

```
# example code

boot_t_test(extra ~ group, data = sleep)
```

```
boot_t_TOST           Bootstrapped TOST with t-tests
```

Description

A function for a bootstrap method for TOST with all types of t-tests.

Usage

```
boot_t_TOST(x, ...)

## Default S3 method:
boot_t_TOST(
  x,
  y = NULL,
  hypothesis = "EQU",
  paired = FALSE,
  var.equal = FALSE,
  eqb,
  low_eqbound,
  high_eqbound,
  eqbound_type = "raw",
  alpha = 0.05,
  bias_correction = TRUE,
  mu = 0,
  R = 1999,
  ...
)

## S3 method for class 'formula'
boot_t_TOST(formula, data, subset, na.action, ...)
```

Arguments

x	a (non-empty) numeric vector of data values.
...	further arguments to be passed to or from methods.
y	an optional (non-empty) numeric vector of data values.
hypothesis	'EQU' for equivalence (default), or 'MET' for minimal effects test, the alternative hypothesis.
paired	a logical indicating whether you want a paired t-test.

<code>var.equal</code>	a logical variable indicating whether to treat the two variances as being equal. If TRUE then the pooled variance is used to estimate the variance otherwise the Welch (or Satterthwaite) approximation to the degrees of freedom is used.
<code>eqb</code>	Equivalence bound. Can provide 1 value (negative value is taken as the lower bound) or 2 specific values that represent the upper and lower equivalence bounds.
<code>low_eqbound</code>	lower equivalence bounds (deprecated).
<code>high_eqbound</code>	upper equivalence bounds (deprecated).
<code>eqbound_type</code>	Type of equivalence bound. Can be set to "SMD" for standardized mean difference (i.e., Cohen's d) or "raw" for the mean difference. Default is "raw". Raw is strongly recommended as SMD bounds will produce biased results.
<code>alpha</code>	alpha level (default = 0.05)
<code>bias_correction</code>	Apply Hedges' correction for bias (default is TRUE).
<code>mu</code>	a number indicating the true value of the mean for the two tailed test (or difference in means if you are performing a two sample test).
<code>R</code>	number of bootstrap replicates
<code>formula</code>	a formula of the form lhs ~ rhs where lhs is a numeric variable giving the data values and rhs either 1 for a one-sample or paired test or a factor with two levels giving the corresponding groups. If lhs is of class "Pair" and rhs is 1, a paired test is done.
<code>data</code>	an optional matrix or data frame (or similar: see <code>model.frame</code>) containing the variables in the formula <code>formula</code> . By default the variables are taken from <code>environment(formula)</code> .
<code>subset</code>	an optional vector specifying a subset of observations to be used.
<code>na.action</code>	a function which indicates what should happen when the data contain NAs. Defaults to <code>getOption("na.action")</code> .

Details

For details on the calculations in this function see `vignette("robustTOST")`.

The implemented test(s) corresponds to the proposal of Chapter 16 of Efron and Tibshirani (1994). Returns TOSTt class object with bootstrapped based results. Please note that the repeated measures "corrected" effect size is not available.

For details on the calculations in this function see `vignette("robustTOST")`.

Value

An S3 object of class "TOSTt" is returned containing the following slots:

"TOST" A table of class "data.frame" containing two-tailed t-test and both one-tailed results.

"eqb" A table of class "data.frame" containing equivalence bound settings.

"effsize" table of class "data.frame" containing effect size estimates

"hypothesis" String stating the hypothesis being tested

"smd" List containing the results of the standardized mean difference calculations (e.g., Cohen's d). Items include: d (estimate), dlow (lower CI bound), dhigh (upper CI bound), d_df (degrees of freedom for SMD), d_sigma (SE), d_lambda (non-centrality), J (bias correction), smd_label (type of SMD), d_denom (denominator calculation)

"alpha" Alpha level set for the analysis.

"method" Type of t-test.

"decision" List included text regarding the decisions for statistical inference.

"boot" List containing the bootstrap samples.

References

Efron, B., & Tibshirani, R. J. (1994). An introduction to the bootstrap. CRC press.

See Also

Other Robust TOST: [boot_log_TOST\(\)](#), [boot_t_test\(\)](#), [log_TOST\(\)](#), [wilcox_TOST\(\)](#)

compare_cor

Comparing two independent correlation coefficients

Description

A function to compare correlations between studies. This function is intended to be used to compare the compatibility of original studies with replication studies (lower p-values indicating lower compatibility).

Usage

```
compare_cor(
  r1,
  df1,
  r2,
  df2,
  method = c("fisher", "kraatz"),
  alternative = c("two.sided", "less", "greater", "equivalence", "minimal.effect"),
  null = 0
)
```

Arguments

r1	Correlation study 1.
df1	Degrees of freedom from study 1 (if a simple correlation the df is N-2).
r2	Correlation study 2.
df2	Degrees of freedom from study 2 (if a simple correlation the df is N-2).

method	Method for determining differences. Default, "z", will use Fisher's transformation, while "Kraatz" will use the Kraatz method.
alternative	a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "greater", "less", "equivalence" (TOST), or "minimal.effect" (TOST). You can specify just the initial letter.
null	a number indicating the null hypothesis. For TOST, this would be equivalence bound.

Details

This function tests for differences between correlations. @return A list with class "htest" containing the following components:

"statistic" z-score

"p.value" numeric scalar containing the p-value for the test under the null hypothesis.

"estimate" difference in SMD between studies

"null.value" the specified hypothesized value for the null hypothesis.

"alternative" character string indicating the alternative hypothesis (the value of the input argument alternative). Possible values are "greater", "less", or "two-sided".

"method" Type of SMD

"data.name" "Summary Statistics" to denote summary statistics were utilized to obtain results.

"cor" Correlation input for the function.

"call" the matched call.

References

Counsell, A., & Cribbie, R. A. (2015). Equivalence tests for comparing correlation and regression coefficients. *The British journal of mathematical and statistical psychology*, 68(2), 292–309. <https://doi.org/10.1111/bmsp.12045>

Anderson, S., & Hauck, W. W. (1983). A new procedure for testing equivalence in comparative bioavailability and other clinical trials. *Communications in Statistics-Theory and Methods*, 12(23), 2663-2692.

compare_smd

Comparing SMDs between independent studies

Description

A function to compare standardized mean differences (SMDs) between studies. This function is intended to be used to compare the compatibility of original studies with replication studies (lower p-values indicating lower compatibility)

Usage

```
compare_smd(
  smd1,
  n1,
  se1 = NULL,
  smd2,
  n2,
  se2 = NULL,
  paired = FALSE,
  alternative = c("two.sided", "less", "greater", "equivalence", "minimal.effect"),
  null = 0,
  TOST = FALSE
)
```

Arguments

smd1, smd2	SMDs from study 1 & 2, respectively.
n1, n2	sample size(s) from study 1 & 2, respectively (can be 1 number or vector of 2 numbers).
se1, se2	User supplied standard errors (SEs). This will override the internal calculations.
paired	a logical indicating whether the SMD is from a paired or independent samples design. If a one-sample design, then paired must be set to TRUE.
alternative	a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "greater", "less", "equivalence" (TOST), or "minimal.effect" (TOST). You can specify just the initial letter.
null	a number indicating the null hypothesis. For TOST, this would be equivalence bound.
TOST	Defunct: use alternative argument. Logical indicator (default = FALSE) to perform two one-sided tests of equivalence (TOST).

Details

This function tests for differences between correlations.

Value

A list with class "htest" containing the following components:

"statistic" z-score

"p.value" numeric scalar containing the p-value for the test under the null hypothesis.

"estimate" difference in SMD between studies

"null.value" the specified hypothesized value for the null hypothesis.

"alternative" character string indicating the alternative hypothesis (the value of the input argument alternative). Possible values are "greater", "less", or "two-sided".

"method" Type of SMD

"data.name" "Summary Statistics" to denote summary statistics were utilized to obtain results.

"smd" SMDs input for the function.

"sample_sizes" Sample sizes input for the function.

"call" the matched call.

See Also

Other compare studies: [boot_compare_smd\(\)](#)

corsum_test

Association/Correlation Test from Summary Statistics

Description

Test for association between paired samples, using one of Pearson's product moment correlation coefficient, Kendall's τ or Spearman's ρ . This is the updated version of the TOSTr function.

Usage

```
corsum_test(
  r,
  n,
  alternative = c("two.sided", "less", "greater", "equivalence", "minimal.effect"),
  method = c("pearson", "kendall", "spearman"),
  alpha = 0.05,
  null = 0
)
```

Arguments

r	observed correlation
n	number of pairs of observations
alternative	a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "greater", "less", "equivalence" (TOST), or "minimal.effect" (TOST). You can specify just the initial letter.
method	a character string indicating which correlation coefficient is to be used for the test. One of "pearson", "kendall", or "spearman", can be abbreviated.
alpha	alpha level (default = 0.05)
null	a number indicating the null hypothesis. Default is a correlation of zero.

Details

This function uses Fisher's z transformation for the correlations, but uses Fieller's correction of the standard error for Spearman's rho and Kendall's tau.

Value

A list with class "htest" containing the following components:

"statistic" z-score

"p.value" the p-value of the test.

"estimate" the estimated measure of association, with name "cor", "tau", or "rho" corresponding to the method employed.

"null.value" the value of the association measure under the null hypothesis.

"alternative" character string indicating the alternative hypothesis (the value of the input argument alternative).

"method" a character string indicating how the association was measured.

"data.name" a character string giving the names of the data.

"call" the matched call.

References

Goertzen, J. R., & Cribbie, R. A. (2010). Detecting a lack of association: An equivalence testing approach. *British Journal of Mathematical and Statistical Psychology*, 63(3), 527-537. <https://doi.org/10.1348/000711009X475> formula page 531.

dataTOSTone

TOST One Sample T-Test

Description

TOST One Sample T-Test in jamovi. This function is not meant to be utilized in R. See `t_TOST` function.

Usage

```
dataTOSTone(
  data,
  vars,
  mu = 0,
  hypothesis = "EQU",
  low_eqbound = -0.5,
  high_eqbound = 0.5,
  eqbound_type = "raw",
  alpha = 0.05,
  desc = FALSE,
  plots = FALSE,
  low_eqbound_d = -999999999,
  high_eqbound_d = -999999999,
  smd_type = "g"
)
```

Arguments

<code>data</code>	the data as a data frame
<code>vars</code>	a vector of strings naming variables of interest in data
<code>mu</code>	a number (default: 0) to compare against
<code>hypothesis</code>	'EQU' for equivalence (default), or 'MET' for minimal effects test, the alternative hypothesis;
<code>low_eqbound</code>	a number (default: -0.5) the lower equivalence bounds
<code>high_eqbound</code>	a number (default: 0.5) the upper equivalence bounds
<code>eqbound_type</code>	'SMD' (default) or 'raw'; whether the bounds are specified in Cohen's d or raw units respectively
<code>alpha</code>	alpha level (default = 0.05)
<code>desc</code>	TRUE or FALSE (default), provide descriptive statistics
<code>plots</code>	TRUE or FALSE (default), provide plots
<code>low_eqbound_d</code>	deprecated
<code>high_eqbound_d</code>	deprecated
<code>smd_type</code>	'd' (default) or 'g'; whether the calculated effect size is biased (d) or bias-corrected (g).

Value

A results object containing:

<code>results\$text</code>	a html
<code>results\$tost</code>	a table
<code>results\$eqb</code>	a table
<code>results\$effsize</code>	a table
<code>results\$desc</code>	a table
<code>results\$plots</code>	an array of images

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$tost$asDF
as.data.frame(results$tost)
```

Examples

```
library("TOSTER")

dataTOSTone(data=iris, vars="Sepal.Width", mu=3, low_eqbound=-0.3, high_eqbound=0.3,
            alpha=0.05, desc=TRUE, plots=TRUE)

TOSTone(m=3.05733, mu=3, sd=0.4358663, n=150, low_eqbound_d=-0.3, high_eqbound_d=0.3, alpha=0.05)
```

dataTOSTpaired	<i>TOST Paired Samples T-Test</i>
----------------	-----------------------------------

Description

TOST Paired Samples T-Test in jamovi. This function is not meant to be utilized in R. See `t_TOST` function.

Usage

```
dataTOSTpaired(
  data,
  pair1,
  pair2,
  hypothesis = "EQU",
  low_eqbound = -0.5,
  high_eqbound = 0.5,
  eqbound_type = "raw",
  alpha = 0.05,
  desc = FALSE,
  plots = FALSE,
  low_eqbound_dz = -999999999,
  high_eqbound_dz = -999999999,
  indplot = FALSE,
  diffplot = FALSE,
  smd_type = "g"
)
```

Arguments

<code>data</code>	the data as a data frame
<code>pair1</code>	A string naming the first part of the pair
<code>pair2</code>	A string naming the second part of the pair
<code>hypothesis</code>	'EQU' for equivalence (default), or 'MET' for minimal effects test, the alternative hypothesis.
<code>low_eqbound</code>	a number (default: 0.5) the lower equivalence bounds
<code>high_eqbound</code>	a number (default: 0.5) the upper equivalence bounds
<code>eqbound_type</code>	'SMD' (default) or 'raw'; whether the bounds are specified in standardized mean difference (Cohen's dz) or raw units respectively
<code>alpha</code>	alpha level (default = 0.05)
<code>desc</code>	TRUE or FALSE (default), provide descriptive statistics
<code>plots</code>	TRUE or FALSE (default), provide plots
<code>low_eqbound_dz</code>	deprecated

high_eqbound_dz	deprecated
indplot	TRUE or FALSE (default), provide plot of paired data.
diffplot	TRUE or FALSE (default), provide plot of difference scores.
smd_type	'd' (default) or 'g'; whether the calculated effect size is biased (d) or bias-corrected (g).

Value

A results object containing:

results\$text	a html
results\$tost	a table
results\$eqb	a table
results\$effsize	a table
results\$desc	a table
results\$plots	an image
results\$indplot	an image
results\$diffplot	an image

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$tost$asDF
as.data.frame(results$tost)
```

References

Mara, C. A., & Cribbie, R. A. (2012). Paired-Samples Tests of Equivalence. *Communications in Statistics - Simulation and Computation*, 41(10), 1928-1943. <<https://doi.org/10.1080/03610918.2011.626545>>, formula page 1932. Note there is a typo in the formula: n-1 should be n (personal communication, 31-08-2016)

Examples

```
## Not run:
library("TOSTER")

dataTOSTpaired(data = randu, pair1 = "x", pair2="y", low_eqbound = -0.3,
               high_eqbound = 0.3, alpha = 0.05, desc = TRUE, plots = TRUE)

## End(Not run)
```

dataTOSTr	<i>TOST Correlation</i>
-----------	-------------------------

Description

TOST for correlations in jamovi. This function is not meant to be utilized in R.

Usage

```
dataTOSTr(
  data,
  pairs,
  cor_type = "pearson",
  hypothesis = "EQU",
  low_eqbound_r = -0.3,
  high_eqbound_r = 0.3,
  alpha = 0.05,
  desc = FALSE,
  plots = FALSE
)
```

Arguments

data	the data as a data frame
pairs	a list of vectors of strings naming variables to correlate from data
cor_type	a character string indicating which correlation coefficient is to be used for the test. One of "pearson", "kendall", or "spearman", can be abbreviated.
hypothesis	'EQU' for equivalence (default), or 'MET' for minimal effects test, the alternative hypothesis.
low_eqbound_r	lower equivalence bounds (e.g., -0.3) expressed in a correlation effect size
high_eqbound_r	upper equivalence bounds (e.g., 0.3) expressed in a correlation effect size
alpha	alpha level (default = 0.05)
desc	TRUE or FALSE (default), provide descriptive statistics
plots	TRUE or FALSE (default), provide plots

Value

A results object containing:

results\$text	a preformatted
results\$tost	a table
results\$desc	a table
results\$plots	an array of images

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$tost$asDF
as.data.frame(results$tost)
```

dataTOSTtwo	<i>TOST Independent Samples T-Test</i>
-------------	--

Description

TOST Independent Samples T-Test for jamovi. This function is not meant to be utilized in R. See `t_TOST` function.

Usage

```
dataTOSTtwo(
  data,
  deps,
  group,
  var_equal = FALSE,
  hypothesis = "EQU",
  low_eqbound = -0.5,
  high_eqbound = 0.5,
  eqbound_type = "raw",
  alpha = 0.05,
  desc = FALSE,
  plots = FALSE,
  descplots = FALSE,
  low_eqbound_d = -999999999,
  high_eqbound_d = -999999999,
  smd_type = "g"
)
```

Arguments

<code>data</code>	the data as a data frame
<code>deps</code>	a vector of strings naming dependent variables in data
<code>group</code>	a string naming the grouping variable in data; must have two levels
<code>var_equal</code>	TRUE or FALSE (default), assume equal variances
<code>hypothesis</code>	'EQU' for equivalence (default), or 'MET' for minimal effects test, the alternative hypothesis.
<code>low_eqbound</code>	a number (default: -0.5) the lower equivalence/MET bounds
<code>high_eqbound</code>	a number (default: 0.5) the upper equivalence/MET bounds
<code>eqbound_type</code>	'SMD' (default) or 'raw'; whether the bounds are specified in Cohen's d or raw units respectively

alpha	alpha level (default = 0.05)
desc	TRUE or FALSE (default), provide descriptive statistics
plots	TRUE or FALSE (default), provide effect size plots
descplots	TRUE or FALSE (default), provide plots
low_eqbound_d	deprecated
high_eqbound_d	deprecated
smd_type	'd' (default) or 'g'; whether the calculated effect size is biased (d) or bias-corrected (g).

Value

A results object containing:

results\$text	a html
results\$tost	a table
results\$eqb	a table
results\$effsize	a table
results\$desc	a table
results\$plots	an array of images
results\$descplots	an array of images

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$tost$asDF
as.data.frame(results$tost)
```

References

Berger, R. L., & Hsu, J. C. (1996). Bioequivalence Trials, Intersection-Union Tests and Equivalence Confidence Sets. *Statistical Science*, 11(4), 283-302.

Gruman, J. A., Cribbie, R. A., & Arpin-Cribbie, C. A. (2007). The effects of heteroscedasticity on tests of equivalence. *Journal of Modern Applied Statistical Methods*, 6(1), 133-140, formula for Welch's t-test on page 135

Examples

```
library(TOSTER)

## Load iris dataset, remove one of the three groups so two are left

data<-iris[which(iris$Species!="versicolor"),]

## TOST procedure on the raw data

dataTOSTtwo(data, deps="Sepal.Width", group="Species", var_equal = TRUE, low_eqbound = -0.5,
             high_eqbound = 0.5, alpha = 0.05, desc = TRUE, plots = TRUE)
```

 datatosttwoprop *TOST Two Proportions*

Description

TOST Two Proportions for jamovi. This function is not meant to be utilized in R.

Usage

```
datatosttwoprop(
  data,
  var,
  level,
  group,
  hypothesis = "EQU",
  low_eqbound = -0.1,
  high_eqbound = 0.1,
  alpha = 0.05,
  desc = FALSE,
  plot = FALSE
)
```

Arguments

data	.
var	.
level	.
group	.
hypothesis	'EQU' for equivalence (default), or 'MET' for minimal effects test, the alternative hypothesis.
low_eqbound	a number (default: -0.1) the lower equivalence bounds
high_eqbound	a number (default: 0.1) the upper equivalence bounds
alpha	alpha level (default = 0.05)
desc	TRUE or FALSE (default), provide descriptive statistics
plot	TRUE or FALSE (default), provide plot

Value

A results object containing:

results\$tost	a table
results\$eqb	a table
results\$desc	a table
results\$plot	an image

Tables can be converted to data frames with `asDF` or `as.data.frame`. For example:

```
results$tost$asDF
as.data.frame(results$tost)
```

equ_anova

Equivalence Test for ANOVA Results

Description

Performs equivalence test on the partial eta-squared (pes) value from ANOVA results.

Usage

```
equ_anova(object, eqbound, MET = FALSE, alpha = 0.05)
```

Arguments

object	an object of returned by either <code>Anova</code> , <code>aov</code> , or <code>afex_aov</code>
eqbound	Equivalence bound for the partial eta-squared.
MET	logical indicator to perform a minimal effect test rather than equivalence test (default is FALSE).
alpha	alpha used for the test (e.g., 0.05).

Details

For details on the calculations in this function see `vignette("the_ftestTOSTER")`.

Value

Returns a data frame containing the ANOVA results with equivalence tests added.

The following abbreviations are used in the table:

- effect: name of the effect.
- df1: Degrees of Freedom in the numerator (i.e. DF effect).
- df2: Degrees of Freedom in the denominator (i.e., DF error).
- F: F-value.
- p.null: p-value (probability of the data given the null hypothesis).
- pes: partial Eta-Squared measure of effect size.
- eqbound: equivalence bound.
- p.equ: p-value (probability of the data given the equivalence hypothesis)

References

Campbell, H., & Lakens, D. (2021). Can we disregard the whole model? Omnibus non-inferiority testing for R² in multi-variable linear regression and in ANOVA. *British Journal of Mathematical and Statistical Psychology*, 74(1), 64-89. doi: 10.1111/bmsp.12201

See Also

Other f-test: [equ_ftest\(\)](#)

equ_ftest

Equivalence Test using an F-test

Description

Performs equivalence test on the partial eta-squared (pes) value for using an F-test.

Usage

```
equ_ftest(Fstat, df1, df2, eqbound = NULL, eqb, MET = FALSE, alpha = 0.05)
```

Arguments

Fstat	The F-statistic from the F-test.
df1	Degrees of freedom for the numerator.
df2	Degrees of freedom for the denominator.
eqbound	Defunct argument for quivalence bound for the partial eta-squared. Default is NULL.
eqb	Defunct argument for quivalence bound for the partial eta-squared.
MET	logical indicator to perform a minimal effect test rather than equivalence test (default is FALSE).
alpha	alpha used for the test (e.g., 0.05).

Details

For details on the calculations in this function see vignette("the_ftestTOSTER").

Value

Object of class `"htest"`

`"statistic"` The value of the F-statistic.

`"parameter"` The degrees of freedom for the F-statistic.

`"p.value"` The he p-value for the test.

`"conf.int"` A confidence interval for the partial eta-squared statistic.

`"estimate"` Estimate of partial eta-squared.

"null.value" The specified for the equivalence test.

"method" A string indicating the type of F-test.

"data.name" A required string indicating that this was calculated from summary statistics.

References

Campbell, H., & Lakens, D. (2021). Can we disregard the whole model? Omnibus non-inferiority testing for R2 in multi-variable linear regression and in ANOVA. *British Journal of Mathematical and Statistical Psychology*, 74(1), 64-89. doi: 10.1111/bmsp.12201

See Also

Other f-test: [equ_anova\(\)](#)

extract_r_paired	<i>Extract Paired Correlation</i>
------------------	-----------------------------------

Description

A function for estimating the correlation from a paired samples t-test. Useful for tsum_TOST when the correlation is not available.

Usage

```
extract_r_paired(m1, sd1, m2, sd2 = NULL, n, tstat = NULL, pvalue = NULL)
```

Arguments

m1	mean of group 1
sd1	standard deviation of group 1
m2	mean of group 2
sd2	standard deviation of group 2
n	Sample size (number of pairs)
tstat	The t-value from a paired samples t-test
pvalue	The two-tailed p-value from a paired samples t-test

Value

An estimate of the correlation.

References

Lajeunesse, M. J. (2011). On the meta-analysis of response ratios for studies with correlated and multi-group designs. *Ecology*, 92(11), 2049-2055

hawthorne
*Data***Description**

A dataset from a study on the Hawthorne effect published by McCambridge et al. The dataset has 5 variables (participant_ID, totaldrinking.x, group, totaldrinking.y, totaldrinking.diff)

Usage

```
hawthorne
```

Format

An object of class `data.frame` with 5474 rows and 5 columns.

Source

McCambridge, J., Wilson, A., Attia, J., Weaver, N., & Kypri, K. (2019). Randomized trial seeking to induce the Hawthorne effect found no evidence for any effect on self-reported alcohol consumption online. *Journal of Clinical Epidemiology*, 108, 102–109.

hstest-helpers
*Helpers for hstest objects***Description**

Functions to help interpret or display objects of the class `hstest`

Usage

```
df_hstest(hstest, test_statistics = TRUE, show_ci = TRUE, extract_names = TRUE)
```

```
describe_hstest(hstest, alpha = NULL, digits = 3)
```

Arguments

<code>hstest</code>	A S3 object of the class <code>hstest</code>
<code>test_statistics</code>	A logical variable to display the test statistics.
<code>show_ci</code>	A logical variable to display the confidence interval.
<code>extract_names</code>	A logical variable to take the names from the S3 object (i.e., statistic for <code>t.test</code> would be "t")
<code>alpha</code>	alpha level (default = 0.05)
<code>digits</code>	integer indicating the number of decimal places.

See Also

Other htest: [as_hctest\(\)](#), [simple_hctest\(\)](#)

Examples

```
# simple example with t-test
tres = t.test(extra ~ group, data = sleep)

# As a data frame
df_hctest(tres)

# Describe t-test results
describe_hctest(tres)
```

log_TOST

TOST with log transformed t-tests

Description

A function for TOST on the log-transformed data using parametric t-tests.

Usage

```
log_TOST(
  x,
  ...,
  hypothesis = "EQU",
  paired = FALSE,
  var.equal = FALSE,
  eqb = 1.25,
  alpha = 0.05,
  null = 1
)

## Default S3 method:
log_TOST(
  x,
  y = NULL,
  hypothesis = c("EQU", "MET"),
  var.equal = FALSE,
  paired = FALSE,
  eqb = 1.25,
  alpha = 0.05,
  null = 1,
  ...
)
```

```
## S3 method for class 'formula'
log_TOST(formula, data, subset, na.action, ...)
```

Arguments

x	a (non-empty) numeric vector of data values.
...	further arguments to be passed to or from methods.
hypothesis	'EQU' for equivalence (default), or 'MET' for minimal effects test, the alternative hypothesis.
paired	a logical indicating whether you want a paired t-test.
var.equal	a logical variable indicating whether to treat the two variances as being equal. If TRUE then the pooled variance is used to estimate the variance otherwise the Welch (or Satterthwaite) approximation to the degrees of freedom is used.
eqb	Equivalence bound; default is 1.25 (FDA guidelines). Can provide 1 value (reciprocal value is taken as the lower bound) or 2 specific values that represent the upper and lower equivalence bounds.
alpha	alpha level (default = 0.05)
null	Null hypothesis value for a two-tailed test (default is 1).
y	an optional (non-empty) numeric vector of data values.
formula	a formula of the form lhs ~ rhs where lhs is a numeric variable giving the data values and rhs either 1 for a one-sample or paired test or a factor with two levels giving the corresponding groups. If lhs is of class "Pair" and rhs is 1, a paired test is done.
data	an optional matrix or data frame (or similar: see model.frame) containing the variables in the formula formula. By default the variables are taken from environment(formula).
subset	an optional vector specifying a subset of observations to be used.
na.action	a function which indicates what should happen when the data contain NAs. Defaults to getOption("na.action").

Details

For details on the calculations in this function see vignette("robustTOST").

Value

An S3 object of class "TOSTt" is returned containing the following slots:

"TOST" A table of class "data.frame" containing two-tailed t-test and both one-tailed results.

"eqb" A table of class "data.frame" containing equivalence bound settings.

"effsize" table of class "data.frame" containing effect size estimates

"hypothesis" String stating the hypothesis being tested

"smd" List containing the results of the means ratio calculation. Items include: d (means ratio estimate), dlow (lower CI bound), dhigh (upper CI bound), d_df (degrees of freedom for SMD), d_sigma (SE), d_lambda (non-centrality), J (bias correction), smd_label (type of SMD), d_denom (denominator calculation)

"alpha" Alpha level set for the analysis.

"method" Type of t-test.

"decision" List included text regarding the decisions for statistical inference.

References

He, Y., Deng, Y., You, C., & Zhou, X. H. (2022). Equivalence tests for ratio of means in bioequivalence studies under crossover design. *Statistical Methods in Medical Research*, 09622802221093721.

Food and Drug Administration (2014). Bioavailability and Bioequivalence Studies Submitted in NDAs or INDs — General Considerations. Center for Drug Evaluation and Research. Docket: FDA-2014-D-0204. <https://www.fda.gov/regulatory-information/search-fda-guidance-documents/bioavailability-and-bioequivalence-studies-submitted-ndas-or-inds-general-considerations>

See Also

Other Robust TOST: [boot_log_TOST\(\)](#), [boot_t_TOST\(\)](#), [boot_t_test\(\)](#), [wilcox_TOST\(\)](#)

Examples

```
data(mtcars)
# Default FDA bioequivalence bounds
log_TOST(mpg ~ am,
data = mtcars)
```

plot_cor	<i>Function to produce plots of the distribution of standard correlation coefficients</i>
----------	---

Description

Function to produce plots of the distribution of standard correlation coefficients

Usage

```
plot_cor(
  r,
  n,
  method = c("pearson", "spearman", "kendall"),
  type = c("c", "cd"),
  levels = c(0.68, 0.9, 0.95, 0.999)
)
```


Arguments

r	The observed correlation coefficient.
n	Total number of observations (sample size).
method	The method by which the coefficient was calculated: pearson, spearman, or kendall (default is "pearson")
type	Choose whether to plot a "consonance" function ("c"), consonance density ("cd"), or both (c("c","cd"); default option).
levels	Numeric vector of confidence levels to display

Details

This function was created so that users could create consonance plots of Pearson's correlation coefficient. These types of plots are discussed by Schweder T, Hjort NL. (2016, ISBN:9781316445051) and Rafi Z, Greenland S. (2020) <doi:10.1186/s12874-020-01105-9>.

Value

Returns plot of the distribution of the correlation coefficient.

See Also

Other plotting functions: [plot_pes\(\)](#), [plot_smd\(\)](#)

plot_pes

Function to produce plots of the distribution of partial eta-squared

Description

Function to produce plots of the distribution of partial eta-squared

Usage

```
plot_pes(  
  Fstat,  
  df1,  
  df2,  
  type = c("c", "cd"),  
  levels = c(0.68, 0.9, 0.95, 0.999)  
)
```

Arguments

Fstat	The F-statistic from the F-test.
df1	Degrees of freedom for the numerator.
df2	Degrees of freedom for the denominator.
type	Choose whether to plot a "consonance" function ("c"), consonance density ("cd"), or both (c("c","cd"); default option).
levels	Numeric vector of confidence levels to display

Details

This function was created so that users could create consonance plots of partial eta-squared from ANOVA-level effects. These types of plots are discussed by Schweder T, Hjort NL. (2016, ISBN:9781316445051) and Rafi Z, Greenland S. (2020) <doi:10.1186/s12874-020-01105-9>.

Value

Returns plot of the distribution of partial eta-squared

See Also

Other plotting functions: [plot_cor\(\)](#), [plot_smd\(\)](#)

plot_smd	<i>Function to produce plots of the distribution of the standardized mean difference</i>
----------	--

Description

Function to produce plots of the distribution of the standardized mean difference

Usage

```
plot_smd(
  d,
  df,
  lambda = NULL,
  sigma = NULL,
  smd_ci = c("goulet", "nct", "t", "z"),
  smd_label = "SMD",
  type = c("c", "cd"),
  levels = c(0.5, 0.9, 0.95, 0.999)
)
```

Arguments

d	Estimate of the standardized mean difference
df	degrees of freedom for the standardized mean difference
lambda	The non-centrality parameter for the standardized mean difference
sigma	The standard error for the standardized mean difference
smd_ci	Method for calculating SMD confidence intervals. Methods include Goulet, noncentral t (nct), central t (t), and normal method (z).
smd_label	Label for the x-axis indicating the SMD measure
type	Choose whether to plot a "consonance" function ("c"), consonance density ("cd"), or both (c("c","cd"); default option).
levels	Numeric vector of confidence levels to display

Details

This function was created so that users could create plots from their own SMD calculations and were inspired by the `concurve` R package. The difficulty is that specific information must be past onto this function. The calculations for the standardized mean difference can be found in the vignettes of this package. These types of plots are discussed by Schweder T, Hjort NL. (2016, ISBN:9781316445051) and Rafi Z, Greenland S. (2020) <doi:10.1186/s12874-020-01105-9>.

Value

Returns plot of the distribution of the standardized mean difference.

See Also

Other plotting functions: [plot_cor\(\)](#), [plot_pes\(\)](#)

powerTOSTone

Power analysis for TOST for one-sample t-test (Cohen's d).

Description

Power analysis for TOST for one-sample t-test (Cohen's d).

Usage

```
powerTOSTone(alpha, statistical_power, N, low_eqbound_d, high_eqbound_d)
```

Arguments

alpha alpha used for the test (e.g., 0.05)
 statistical_power desired power (e.g., 0.8)
 N sample size (e.g., 108)
 low_eqbound_d lower equivalence bounds (e.g., -0.5) expressed in standardized mean difference (Cohen's d)
 high_eqbound_d upper equivalence bounds (e.g., 0.5) expressed in standardized mean difference (Cohen's d)

Value

Calculate either achieved power, equivalence bounds, or required N, assuming a true effect size of 0. Returns a string summarizing the power analysis, and a numeric variable for number of observations, equivalence bounds, or power.

References

Chow, S.-C., Wang, H., & Shao, J. (2007). Sample Size Calculations in Clinical Research, Second Edition - CRC Press Book. Formula 3.1.9

Examples

```
## Sample size for alpha = 0.05, 90% power, equivalence bounds of
## Cohen's d = -0.3 and Cohen's d = 0.3, and assuming true effect = 0
powerTOSTone(alpha=0.05, statistical_power=0.9, low_eqbound_d=-0.3, high_eqbound_d=0.3)

## Power for sample size of 121, alpha = 0.05, equivalence bounds of
## Cohen's d = -0.3 and Cohen's d = 0.3, and assuming true effect = 0

powerTOSTone(alpha=0.05, N=121, low_eqbound_d=-0.3, high_eqbound_d=0.3)

## Equivalence bounds for sample size of 121, alpha = 0.05, statistical power of
## 0.9, and assuming true effect = 0

powerTOSTone(alpha=0.05, N=121, statistical_power=.9)
```

powerTOSTone.raw *Power analysis for TOST for one-sample t-test (raw scores).*

Description

Power analysis for TOST for one-sample t-test (raw scores).

Usage

```
powerTOSTone.raw(alpha, statistical_power, N, sd, low_eqbound, high_eqbound)
```

Arguments

alpha	alpha used for the test (e.g., 0.05)
statistical_power	desired power (e.g., 0.8)
N	sample size (e.g., 108)
sd	population standard deviation
low_eqbound	lower equivalence bounds (e.g., -0.5) expressed in raw scores
high_eqbound	upper equivalence bounds (e.g., 0.5) expressed in raw scores

Value

Calculate either achieved power, equivalence bounds, or required N, assuming a true effect size of 0. Returns a string summarizing the power analysis, and a numeric variable for number of observations, equivalence bounds, or power.

References

Chow, S.-C., Wang, H., & Shao, J. (2007). Sample Size Calculations in Clinical Research, Second Edition - CRC Press Book. Formula 3.1.9

Examples

```
## Sample size for alpha = 0.05, 90% power, equivalence bounds of -0.3 and 0.3 in
## raw units, assuming pooled standard deviation of 1, and assuming true effect = 0
powerTOSTone.raw(alpha=0.05, statistical_power=0.9, sd = 1, low_eqbound=-0.3, high_eqbound=0.3)

## Power for sample size of 121, alpha = 0.05, equivalence bounds of
## -0.3 and 0.3 in raw units, assuming pooled standard deviation of 1, and assuming true effect = 0
powerTOSTone.raw(alpha=0.05, N=121, sd = 1, low_eqbound=-0.3, high_eqbound=0.3)

## Power for sample size of 121, alpha = 0.05, statistical power of
## 0.9, and assuming true effect = 0
powerTOSTone.raw(alpha=0.05, N=121, statistical_power=.9, sd=1)
```

powerTOSTpaired *Power analysis for TOST for dependent t-test (Cohen's dz).*

Description

Power analysis for TOST for dependent t-test (Cohen's dz).

Usage

```
powerTOSTpaired(alpha, statistical_power, N, low_eqbound_dz, high_eqbound_dz)
```

Arguments

alpha alpha used for the test (e.g., 0.05)
 statistical_power desired power (e.g., 0.8)
 N number of pairs (e.g., 96)
 low_eqbound_dz lower equivalence bounds (e.g., -0.5) expressed in standardized mean difference (Cohen's dz)
 high_eqbound_dz upper equivalence bounds (e.g., 0.5) expressed in standardized mean difference (Cohen's dz)

Value

Calculate either achieved power, equivalence bounds, or required N, assuming a true effect size of 0. Returns a string summarizing the power analysis, and a numeric variable for number of observations, equivalence bounds, or power.

References

Chow, S.-C., Wang, H., & Shao, J. (2007). Sample Size Calculations in Clinical Research, Second Edition - CRC Press Book. Formula 3.1.9

Examples

```
## Sample size for alpha = 0.05, 80% power, equivalence bounds of
## Cohen's dz = -0.3 and Cohen's d = 0.3, and assuming true effect = 0
powerTOSTpaired(alpha=0.05,statistical_power=0.8,low_eqbound_dz=-0.3,high_eqbound_dz=0.3)

## Sample size for alpha = 0.05, N = 96 pairs, equivalence bounds of
## Cohen's dz = -0.3 and Cohen's d = 0.3, and assuming true effect = 0
powerTOSTpaired(alpha=0.05,N=96,low_eqbound_dz=-0.3,high_eqbound_dz=0.3)

## Equivalence bounds for alpha = 0.05, N = 96 pairs, statistical power of
## 0.8, and assuming true effect = 0
powerTOSTpaired(alpha=0.05,N=96,statistical_power=0.8)
```

powerTOSTpaired.raw *Power analysis for TOST for dependent t-test (raw scores).*

Description

Power analysis for TOST for dependent t-test (raw scores).

Usage

```
powerTOSTpaired.raw(
  alpha,
  statistical_power,
  N,
  sdif,
  low_eqbound,
  high_eqbound
)
```

Arguments

alpha	alpha used for the test (e.g., 0.05)
statistical_power	desired power (e.g., 0.8)
N	number of pairs (e.g., 96)
sdif	standard deviation of the difference scores
low_eqbound	lower equivalence bounds (e.g., -0.5) expressed in raw mean difference
high_eqbound	upper equivalence bounds (e.g., 0.5) expressed in raw mean difference

Value

Calculate either achieved power, equivalence bounds, or required N, assuming a true effect size of 0. Returns a string summarizing the power analysis, and a numeric variable for number of observations, equivalence bounds, or power.

References

Chow, S.-C., Wang, H., & Shao, J. (2007). Sample Size Calculations in Clinical Research, Second Edition - CRC Press Book. Formula 3.1.9

Examples

```
## Sample size for alpha = 0.05, 80% power, equivalence bounds of -3 and 3 in raw units
## and assuming a standard deviation of the difference scores of 10, and assuming a true effect = 0
powerTOSTpaired.raw(alpha=0.05,statistical_power=0.8,low_eqbound=-3, high_eqbound=3, sdif=10)

## Sample size for alpha = 0.05, N = 96 pairs, equivalence bounds of -3 and 3 in raw units
## and assuming a standard deviation of the difference scores of 10, and assuming a true effect = 0
powerTOSTpaired.raw(alpha=0.05,N=96,low_eqbound=-3, high_eqbound=3, sdif=10)

## Equivalence bounds for alpha = 0.05, N = 96 pairs, statistical power of 0.8
## and assuming a standard deviation of the difference scores of 10, and assuming a true effect = 0
powerTOSTpaired.raw(alpha=0.05,N=96, statistical_power=0.8, sdif=10)
```

powerTOSTr

Power analysis for TOST for correlations.

Description

Power analysis for TOST for correlations.

Usage

```
powerTOSTr(alpha, statistical_power, N, low_eqbound_r, high_eqbound_r)
```

Arguments

alpha	alpha used for the test (e.g., 0.05)
statistical_power	desired power (e.g., 0.8)
N	number of pairs (e.g., 96)
low_eqbound_r	lower equivalence bounds (e.g., -0.3) expressed in a correlation effect size
high_eqbound_r	upper equivalence bounds (e.g., 0.3) expressed in a correlation effect size

Value

Calculate either achieved power, equivalence bounds, or required N, assuming a true effect size of 0. Returns a string summarizing the power analysis, and a numeric variable for number of observations, equivalence bounds, or power.

Examples

```
## Sample size for alpha = 0.05, 90% power, equivalence bounds of
## r = -0.1 and r = 0.1, assuming true effect = 0
powerTOSTr(alpha=0.05, statistical_power=0.9, low_eqbound_r=-0.1, high_eqbound_r=0.1)

## Sample size for alpha = 0.05, N=536, equivalence bounds of
## r = -0.1 and r = 0.1, assuming true effect = 0
powerTOSTr(alpha=0.05, N=536, low_eqbound_r=-0.1, high_eqbound_r=0.1)

## Equivalence bounds for alpha = 0.05, N=536, statistical power of
## 0.9, assuming true effect = 0
powerTOSTr(alpha=0.05, N=536, statistical_power=0.9)
```

powerTOSTtwo *Power analysis for TOST for independent t-test (Cohen's d).*

Description

Power analysis for TOST for independent t-test (Cohen's d).

Usage

```
powerTOSTtwo(alpha, statistical_power, N, low_eqbound_d, high_eqbound_d)
```

Arguments

alpha	alpha used for the test (e.g., 0.05)
statistical_power	desired power (e.g., 0.8)
N	sample size per group (e.g., 108)
low_eqbound_d	lower equivalence bounds (e.g., -0.5) expressed in standardized mean difference (Cohen's d)
high_eqbound_d	upper equivalence bounds (e.g., 0.5) expressed in standardized mean difference (Cohen's d)

Value

Calculate either achieved power, equivalence bounds, or required N, assuming a true effect size of 0. Returns a string summarizing the power analysis, and a numeric variable for number of observations, equivalence bounds, or power.

References

Chow, S.-C., Wang, H., & Shao, J. (2007). *Sample Size Calculations in Clinical Research*, Second Edition - CRC Press Book. Formula 3.2.4 with $k = 1$

Examples

```
## Sample size for alpha = 0.05, 80% power, equivalence bounds of
## Cohen's d = -0.4 and Cohen's d = 0.4, assuming true effect = 0
powerTOSTtwo(alpha=0.05, statistical_power=0.8, low_eqbound_d=-0.4, high_eqbound_d=0.4)

## Statistical power for alpha = 0.05, N = 108 per group, equivalence bounds of
## Cohen's d = -0.4 and Cohen's d = 0.4, assuming true effect = 0
powerTOSTtwo(alpha=0.05, N=108, low_eqbound_d=-0.4, high_eqbound_d=0.4)

## Equivalence bounds for alpha = 0.05, N = 108 per group, statistical power of
## 0.8, assuming true effect = 0
powerTOSTtwo(alpha=0.05, N=108, statistical_power=0.8)
```

powerTOSTtwo.prop	<i>Power analysis for TOST for difference between two proportions using Z-test (pooled)</i>
-------------------	---

Description

Power analysis for TOST for difference between two proportions using Z-test (pooled)

Usage

```
powerTOSTtwo.prop(
  alpha,
  statistical_power,
  prop1,
  prop2,
  N,
  low_eqbound_prop,
  high_eqbound_prop
)
```

Arguments

alpha	alpha used for the test (e.g., 0.05)
statistical_power	desired power (e.g., 0.8)
prop1	expected proportion in control condition
prop2	expected proportion in the experimental condition
N	sample size (e.g., 108)
low_eqbound_prop	lower equivalence bounds (e.g., -0.05) expressed in proportion
high_eqbound_prop	upper equivalence bounds (e.g., 0.05) expressed in proportion

Value

Calculate either achieved power, equivalence bounds, or required N, assuming a true effect size of 0. Returns a string summarizing the power analysis, and a numeric variable for number of observations, equivalence bounds, or power.

References

Silva, G. T. da, Logan, B. R., & Klein, J. P. (2008). Methods for Equivalence and Noninferiority Testing. *Biology of Blood and Marrow Transplantation: Journal of the American Society for Blood and Marrow Transplantation*, 15(1 Suppl), 120-127. <https://doi.org/10.1016/j.bbmt.2008.10.004>

Julious, S. A. & Campbell, M. J. (2012). Tutorial in biostatistics: sample sizes for parallel group clinical trials with binary data. *Statistics in Medicine*, 31:2904-2936. Chow, S.-C., Wang, H., &

Shao, J. (2007). *Sample Size Calculations in Clinical Research*, Second Edition (2 edition). Boca Raton: Chapman and Hall/CRC.

Examples

```
## Sample size for alpha = 0.05, 90% power, assuming true effect prop1 = prop 2 = 0.5,
## equivalence bounds of 0.4 and 0.6 (so low_eqbound_prop = -0.1 and high_eqbound_prop = 0.1)

powerTOSTtwo.prop(alpha = 0.05, statistical_power = 0.9, prop1 = 0.5, prop2 = 0.5,
  low_eqbound_prop = -0.1, high_eqbound_prop = 0.1)

## Power for alpha = 0.05, N 542 , assuming true effect prop1 = prop 2 = 0.5,
## equivalence bounds of 0.4 and 0.6 (so low_eqbound_prop = -0.1 and high_eqbound_prop = 0.1)

powerTOSTtwo.prop(alpha = 0.05, N = 542, prop1 = 0.5, prop2 = 0.5,
  low_eqbound_prop = -0.1, high_eqbound_prop = 0.1)

## Equivalence bounds for alpha = 0.05, N 542 , assuming true effect prop1 = prop 2 = 0.5,
## and 90% power

powerTOSTtwo.prop(alpha=0.05, statistical_power=0.9, N=542, prop1 = 0.5, prop2 = 0.5)

#Example 4.2.4 from Chow, Wang, & Shao (2007, p. 93)
powerTOSTtwo.prop(alpha=0.05, statistical_power=0.8, prop1 = 0.75, prop2 = 0.8,
  low_eqbound_prop = -0.2, high_eqbound_prop = 0.2)

# Example 5 from Julious & Campbell (2012, p. 2932)
powerTOSTtwo.prop(alpha=0.025, statistical_power=0.9, prop1 = 0.8, prop2 = 0.8,
  low_eqbound_prop=-0.1, high_eqbound_prop=0.1)
# From Machin, D. (Ed.). (2008). Sample size tables for clinical studies (3rd ed).

# Example 9.4b equivalence of two proportions (p. 113) #
powerTOSTtwo.prop(alpha=0.010, statistical_power=0.8, prop1 = 0.5, prop2 = 0.5,
  low_eqbound_prop = -0.2, high_eqbound_prop = 0.2)/2
```

powerTOSTtwo.raw *Power analysis for TOST for independent t-test (raw scores).*

Description

Power analysis for TOST for independent t-test (raw scores).

Usage

```
powerTOSTtwo.raw(
  alpha,
  statistical_power,
  N,
  sdpooled,
  low_eqbound,
```

```

    high_eqbound,
    delta = 0
)

```

Arguments

alpha	alpha used for the test (e.g., 0.05)
statistical_power	desired power (e.g., 0.8)
N	sample size per group (e.g., 108)
sdpooled	specify the pooled standard deviation
low_eqbound	lower equivalence bounds (e.g., -0.5) expressed in raw scale units (e.g., scale-points)
high_eqbound	upper equivalence bounds (e.g., 0.5) expressed in raw scale units (e.g., scale-points)
delta	hypothesized true value for the difference between the 2 means. Default is zero.

Value

Calculate either achieved power, equivalence bounds, or required N, assuming a true effect size of 0. Returns a string summarizing the power analysis, and a numeric variable for number of observations, equivalence bounds, or power.

References

Chow, S.-C., Wang, H., & Shao, J. (2007). Sample Size Calculations in Clinical Research, Second Edition - CRC Press Book. Formula 3.2.4 with $k = 1$

Examples

```

## Sample size for alpha = 0.05, 80% power, equivalence bounds of -200 and 200 in raw
## units, assuming pooled standard deviation of 350, and assuming true effect = 0
powerTOSTtwo.raw(alpha=0.05,statistical_power=0.8,low_eqbound=-200,high_eqbound=200,sdpooled=350)

## Power for alpha = 0.05, N = 53 per group, equivalence bounds of
## -200 and 200 in raw units, assuming sdpooled = 350 and true effect = 0
powerTOSTtwo.raw(alpha=0.05, N=53, low_eqbound=-200, high_eqbound=200, sdpooled=350)

## Equivalence bounds for alpha = 0.05, N = 108 per group, statistical power of
## 0.8, assuming true effect = 0
powerTOSTtwo.raw(alpha=0.05, N=108, statistical_power=0.8, sdpooled=350)

```

Description

Power analysis for TOST for an F-test

Usage

```
power_eq_f(alpha = 0.05, df1, df2, eqbound)
```

Arguments

alpha	alpha used for the test (e.g., 0.05)
df1	Degrees of freedom for the numerator
df2	Degrees of freedom for the denominator
eqbound	Equivalence bound for the partial eta-squared

Value

Object of class "power.htest"

References

Campbell, H., & Lakens, D. (2021). Can we disregard the whole model? Omnibus non-inferiority testing for R² in multi-variable linear regression and in ANOVA. *British Journal of Mathematical and Statistical Psychology*, 74(1), 64-89. doi: 10.1111/bmsp.12201

See Also

Other power: [power_t_TOST\(\)](#)

Examples

```
## Statistical power for alpha = 0.05, 3 groups, n = 80 per group, equivalence bound of
## partial eta squared = 0.01, assuming true effect = 0.
## df1 = number of groups - 1 = 3 - 1 = 2.
## df2 = Total N - number of groups = 240 - 3 = 237.
power_eq_f(alpha=0.05, df1=3, df2 = 237, eqbound = 0.01)
```

 power_t_TOST

Power calculations for TOST for t-tests

Description

Calculates the exact power of two one sided t-tests (TOST) for one, two, and paired samples.

Usage

```
power_t_TOST(
  n = NULL,
  delta = 0,
  sd = 1,
  eqb,
  low_eqbound = NULL,
  high_eqbound = NULL,
  alpha = NULL,
  power = NULL,
  type = "two.sample"
)
```

Arguments

n	number of observations per group. 2 sample sizes, in a vector, can be provided for the two sample case.
delta	true difference in means (default is 0)
sd	population standard deviation. Standard deviation of the differences for paired samples
eqb	Equivalence bound. Can provide 1 value (negative value is taken as the lower bound) or 2 specific values that represent the upper and lower equivalence bounds.
low_eqbound	Lower equivalence bounds. Deprecated use eqb.
high_eqbound	Upper equivalence bounds. Deprecated use eqb.
alpha	a priori alpha-level (i.e., significance level)
power	power of the TOST procedure (1-beta)
type	string specifying the type of t-test.

Details

The exact calculations of power are based on Owen's Q-function or by direct integration of the bivariate non-central t-distribution (inspired by the PowerTOST package). Approximate power is implemented via the non-central t-distribution or the 'shifted' central t-distribution.

Note

The power function in this package is limited. Please see the PowerTOST R package for more options.

References

Phillips KF. Power of the Two One-Sided Tests Procedure in Bioequivalence. *J Pharmacokin Biopharm.* 1990;18(2):137–44. doi: 10.1007/BF01063556

Diletti D, Hauschke D, Steinijans VW. Sample Size Determination for Bioequivalence Assessment by Means of Confidence Intervals. *Int J Clin Pharmacol Ther Toxicol.* 1991;29(1):1–8.

See Also

Other power: [power_eq_f\(\)](#)

 rbs

Non-parametric standardized effect sizes (replicates of ses_calc)

Description

Effect sizes for simple (one or two sample) non-parametric tests.

Usage

```
rbs(x, y = NULL, mu = 0, conf.level = 0.95, paired = FALSE)
```

```
np_ses(
  x,
  y = NULL,
  mu = 0,
  conf.level = 0.95,
  paired = FALSE,
  ses = c("rb", "odds", "cstat")
)
```

Arguments

x	a (non-empty) numeric vector of data values.
y	an optional (non-empty) numeric vector of data values.
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].
conf.level	confidence level of the interval.
paired	a logical indicating whether you want to calculate a paired test.
ses	Rank-biserial (rb), odds (odds), and concordance probability (cstat).

Details

This method was adapted from the `effectsize R` package. 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' indicating that all values of the second sample are smaller than the first sample, to '+1' indicating that all values of the second sample are larger than the first sample.

In addition, the rank-biserial correlation can be transformed into a concordance probability (i.e., probability of superiority) or into a generalized odds (WMW odds or Agresti's generalized odds ratio).

Ties When tied values occur, they are each given the average of the ranks that would have been given had no ties occurred. No other corrections have been implemented yet.

Confidence Intervals Confidence intervals for the standardized effect sizes are estimated using the normal approximation (via Fisher's transformation).

Value

Returns a list of results including the rank biserial correlation, logical indicator if it was a paired method, setting for mu, and confidence interval.

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.
- Kendall, M.G. (1948) Rank correlation methods. London: Griffin.
- 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.

 ses_calc

SES Calculation

Description

Standardized effect size (SES), these are the effect sizes not considered SMDs.

Usage

```
ses_calc(x, ..., paired = FALSE, ses = "rb", alpha = 0.05)
```

```
## Default S3 method:
```

```
ses_calc(
  x,
  y = NULL,
  paired = FALSE,
  ses = c("rb", "odds", "cstat"),
  alpha = 0.05,
  mu = 0,
  ...
)
```

```
## S3 method for class 'formula'
```

```
ses_calc(formula, data, subset, na.action, ...)
```

Arguments

x	a (non-empty) numeric vector of data values.
...	further arguments to be passed to or from methods.
paired	a logical indicating whether you want to calculate a paired test.
ses	Standardized effect size. Default is "rb" for rank-biserial correlation. Options also include "cstat" for concordance probability, or "odds" for Wilcoxon-Mann-Whitney odds (otherwise known as Agresti's generalized odds ratio).
alpha	alpha level (default = 0.05)
y	an optional (non-empty) numeric vector of data values.
mu	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].
formula	a formula of the form lhs ~ rhs where lhs is a numeric variable giving the data values and rhs either 1 for a one-sample or paired test or a factor with two levels giving the corresponding groups. If lhs is of class "Pair" and rhs is 1, a paired test is done.
data	an optional matrix or data frame (or similar: see model.frame) containing the variables in the formula formula. By default the variables are taken from environment(formula).

subset an optional vector specifying a subset of observations to be used.
na.action a function which indicates what should happen when the data contain NAs. Defaults to `getOption("na.action")`.

Details

For details on the calculations in this function see `vignette("robustTOST")`.

Value

A data frame containing the standardized effect size.

See Also

Other effect sizes: `smd_calc()`

Examples

```
## Not run:
ses_calc(formula = extra ~ group, data = sleep, paired = TRUE, ses = "r")

## End(Not run)
```

simple_htest *One, two, and paired samples hypothesis tests*

Description

Performs one or two sample t-tests or Wilcoxon-Mann-Whitney rank-based tests with expanded options compared to `t.test` or `wilcox.test`.

Usage

```
simple_htest(
  x,
  ...,
  paired = FALSE,
  alternative = c("two.sided", "less", "greater", "equivalence", "minimal.effect"),
  mu = 0,
  alpha = 0.05
)

## Default S3 method:
simple_htest(
  x,
  y = NULL,
  test = c("t.test", "wilcox.test"),
  paired = FALSE,
```

```

    alternative = c("two.sided", "less", "greater", "equivalence", "minimal.effect"),
    mu = 0,
    alpha = 0.05,
    ...
)

## S3 method for class 'formula'
simple_hctest(formula, data, subset, na.action, ...)

```

Arguments

x	a (non-empty) numeric vector of data values.
...	further arguments to be passed to or from methods.
paired	a logical indicating whether you want a paired t-test.
alternative	a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "greater", "less", "equivalence" (TOST), or "minimal.effect" (TOST). You can specify just the initial letter.
mu	a number indicating the true value of the mean for the two tailed test (or difference in means if you are performing a two sample test).
alpha	alpha level (default = 0.05)
y	an optional (non-empty) numeric vector of data values.
test	a character string specifying what type of hypothesis test to use. Options are limited to "wilcox.test" and "t.test". You can specify just the initial letter.
formula	a formula of the form lhs ~ rhs where lhs is a numeric variable giving the data values and rhs either 1 for a one-sample or paired test or a factor with two levels giving the corresponding groups. If lhs is of class "Pair" and rhs is 1, a paired test is done.
data	an optional matrix or data frame (or similar: see model.frame) containing the variables in the formula formula. By default the variables are taken from environment(formula).
subset	an optional vector specifying a subset of observations to be used.
na.action	a function which indicates what should happen when the data contain NAs. Defaults to getOption("na.action").

Details

Currently, this function allows for traditional or TOST hypothesis tests using "t.test" or "wilcox.test". The type of test, t-test or Wilcoxon-Mann-whitney, can be selected with the "test" argument. More information on the tests can be found in the documentation for the "t.test" and "wilcox.test" functions.

alternative = "greater" is the alternative that x is larger than y (on average). If alternative = "equivalence" then the alternative is that the difference between x and y is between the two null values (mu). If alternative = "minimal.effect" then the alternative is that the difference between x and y is less than the lowest null value or greater than the highest.

Value

A list with class "htest" containing the following components:

`statistic` the value of the t-statistic.

`parameter` the degrees of freedom for the t-statistic.

`p.value` the p-value for the test.

`conf.int` a confidence interval for the mean appropriate to the specified alternative hypothesis.

`estimate` the estimated mean or difference in means depending on whether it was a one-sample test or a two-sample test.

`null.value` the specified hypothesized value of the mean or mean difference. May be 2 values.

`stderr` the standard error of the mean (difference), used as denominator in the t-statistic formula.

`alternative` a character string describing the alternative hypothesis.

`method` a character string indicating what type of t-test was performed.

`data.name` a character string giving the name(s) of the data..

See Also

Other TOST: [t_TOST\(\)](#), [tsum_TOST\(\)](#)

Other htest: [as_hptest\(\)](#), [htest-helpers](#)

Examples

```
data(mtcars)
simple_hptest(mpg ~ am,
data = mtcars,
alternative = "e",
mu = 3)
```

smd_calc

SMD Calculation

Description

A function to only calculate standardized mean differences.

Usage

```
smd_calc(
  x,
  ...,
  paired = FALSE,
  var.equal = FALSE,
  alpha = 0.05,
  bias_correction = TRUE,
  rm_correction = FALSE,
```

```

    glass = NULL,
    smd_ci = c("nct", "goulet", "t", "z")
  )

## Default S3 method:
smd_calc(
  x,
  y = NULL,
  paired = FALSE,
  var.equal = FALSE,
  alpha = 0.05,
  mu = 0,
  bias_correction = TRUE,
  rm_correction = FALSE,
  glass = NULL,
  smd_ci = c("nct", "goulet", "t", "z"),
  ...
)

## S3 method for class 'formula'
smd_calc(formula, data, subset, na.action, ...)

```

Arguments

x	a (non-empty) numeric vector of data values.
...	further arguments to be passed to or from methods.
paired	a logical indicating whether you want a paired t-test.
var.equal	a logical variable indicating whether to treat the two variances as being equal. If TRUE then the pooled variance is used to estimate the variance otherwise the Welch (or Satterthwaite) approximation to the degrees of freedom is used.
alpha	alpha level (default = 0.05)
bias_correction	Apply Hedges' correction for bias (default is TRUE).
rm_correction	Repeated measures correction to make standardized mean difference Cohen's d(rm). This only applies to repeated/paired samples. Default is FALSE.
glass	A option to calculate Glass's delta as an alternative to Cohen's d type SMD. Default is NULL to not calculate Glass's delta, "glass1" will use the first group's SD as the denominator whereas "glass2" will use the 2nd group's SD.
smd_ci	Method for calculating SMD confidence intervals. Methods include Goulet, noncentral t (nct), central t (t), and normal method (z).
y	an optional (non-empty) numeric vector of data values.
mu	Null value. Deviating from zero will give the x-y-mu.
formula	a formula of the form lhs ~ rhs where lhs is a numeric variable giving the data values and rhs either 1 for a one-sample or paired test or a factor with two levels giving the corresponding groups. If lhs is of class "Pair" and rhs is 1, a paired test is done.

data	an optional matrix or data frame (or similar: see <code>model.frame</code>) containing the variables in the formula. By default the variables are taken from <code>environment(formula)</code> .
subset	an optional vector specifying a subset of observations to be used.
na.action	a function which indicates what should happen when the data contain NAs. Defaults to <code>getOption("na.action")</code> .

Details

For details on the calculations in this function see `vignette("SMD_calcs")`.

Value

A data frame containing the SMD estimates.

See Also

Other effect sizes: `ses_calc()`

Examples

```
## Not run:
smd_calc(formula = extra ~ group, data = sleep, paired = TRUE, smd_ci = "nct")

## End(Not run)
```

TOSTmeta

TOST function for meta-analysis

Description

TOST function for meta-analysis

Usage

```
TOSTmeta(
  ES,
  var,
  se,
  low_eqbound_d,
  high_eqbound_d,
  alpha,
  plot = TRUE,
  verbose = TRUE
)
```

Arguments

ES	meta-analytic effect size
var	meta-analytic variance
se	standard error
low_eqbound_d	lower equivalence bounds (e.g., -0.5) expressed in standardized mean difference (Cohen's d)
high_eqbound_d	upper equivalence bounds (e.g., 0.5) expressed in standardized mean difference (Cohen's d)
alpha	alpha level (default = 0.05)
plot	set whether results should be plotted (plot = TRUE) or not (plot = FALSE) - defaults to TRUE
verbose	logical variable indicating whether text output should be generated (verbose = TRUE) or not (verbose = FALSE) - default to TRUE

Value

Returns TOST Z-value 1, TOST p-value 1, TOST Z-value 2, TOST p-value 2, alpha, low equivalence bound d, high equivalence bound d, Lower limit confidence interval TOST, Upper limit confidence interval TOST

References

Rogers, J. L., Howard, K. I., & Vessey, J. T. (1993). Using significance tests to evaluate equivalence between two experimental groups. *Psychological Bulletin*, 113(3), 553, formula page 557.

Examples

```
## Run TOSTmeta by specifying the standard error
TOSTmeta(ES=0.12, se=0.09, low_eqbound_d=-0.2, high_eqbound_d=0.2, alpha=0.05)
## Run TOSTmeta by specifying the variance
TOSTmeta(ES=0.12, var=0.0081, low_eqbound_d=-0.2, high_eqbound_d=0.2, alpha=0.05)
## If both variance and se are specified, TOSTmeta will use standard error and ignore variance
TOSTmeta(ES=0.12, var=9999, se = 0.09, low_eqbound_d=-0.2, high_eqbound_d=0.2, alpha=0.05)
```

TOSTnp-methods

Methods for TOSTnp objects

Description

Methods defined for objects returned from the wilcox_TOST function.

Usage

```
## S3 method for class 'TOSTnp'
print(x, digits = 4, ...)

## S3 method for class 'TOSTnp'
describe(x, digits = 3, ...)
```

Arguments

x object of class TOSTnp
 digits Number of digits to print for p-values
 ... further arguments passed through, see description of return value for details.
[TOSTnp-methods](#).

Value

print Prints short summary of the tests.
 describe Verbose description of results.

Examples

```
# example code
data(mtcars)
res1 = wilcox_TOST(mpg ~ am,data = mtcars,eqb = 3)

# PRINT
print(res1)

# DESCRIPTION
describe(res1)
```

TOSTone

TOST function for a one-sample t-test (Cohen's d)

Description

TOST function for a one-sample t-test (Cohen's d)

Usage

```
TOSTone(
  m,
  mu,
  sd,
  n,
  low_eqbound_d,
  high_eqbound_d,
  alpha,
  plot = TRUE,
  verbose = TRUE
)
```


Arguments

m	mean
mu	value to compare against
sd	standard deviation
n	sample size
low_eqbound_d	lower equivalence bounds (e.g., -0.5) expressed in standardized mean difference (Cohen's d)
high_eqbound_d	upper equivalence bounds (e.g., 0.5) expressed in standardized mean difference (Cohen's d)
alpha	alpha level (default = 0.05)
plot	set whether results should be plotted (plot = TRUE) or not (plot = FALSE) - defaults to TRUE
verbose	logical variable indicating whether text output should be generated (verbose = TRUE) or not (verbose = FALSE) - default to TRUE

Value

Returns TOST t-value 1, TOST p-value 1, TOST t-value 2, TOST p-value 2, degrees of freedom, low equivalence bound, high equivalence bound, Lower limit confidence interval TOST, Upper limit confidence interval TOST

Examples

```
## Test observed mean of 0.54 and standard deviation of 1.2 in sample of 100 participants
## against 0.5 given equivalence bounds of Cohen's d = -0.3 and 0.3, with an alpha = 0.05.
TOSTone(m=0.54,mu=0.5,sd=1.2,n=100,low_eqbound_d=-0.3, high_eqbound_d=0.3, alpha=0.05)
```

TOSTone.raw

TOST function for a one-sample t-test (raw scores)

Description

TOST function for a one-sample t-test (raw scores)

Usage

```
TOSTone.raw(
  m,
  mu,
  sd,
  n,
  low_eqbound,
  high_eqbound,
  alpha,
  plot = TRUE,
  verbose = TRUE
)
```

Arguments

m	mean
mu	value to compare against
sd	standard deviation
n	sample size
low_eqbound	lower equivalence bounds (e.g., -0.5) expressed in raw units
high_eqbound	upper equivalence bounds (e.g., 0.5) expressed in raw units
alpha	alpha level (default = 0.05)
plot	set whether results should be plotted (plot = TRUE) or not (plot = FALSE) - defaults to TRUE
verbose	logical variable indicating whether text output should be generated (verbose = TRUE) or not (verbose = FALSE) - default to TRUE

Value

Returns TOST t-value 1, TOST p-value 1, TOST t-value 2, TOST p-value 2, degrees of freedom, low equivalence bound, high equivalence bound, Lower limit confidence interval TOST, Upper limit confidence interval TOST

Examples

```
## Test observed mean of 0.52 and standard deviation of 0.52 in sample of 300 participants
## against 0.5 given equivalence bounds in raw units of -0.1 and 0.1, with an alpha = 0.05.
TOSTone.raw(m=0.52,mu=0.5,sd=0.5,n=300,low_eqbound=-0.1, high_eqbound=0.1, alpha=0.05)
```

TOSTpaired

TOST function for a dependent t-test (Cohen's dz)

Description

TOST function for a dependent t-test (Cohen's dz)

Usage

```
TOSTpaired(
  n,
  m1,
  m2,
  sd1,
  sd2,
  r12,
  low_eqbound_dz,
  high_eqbound_dz,
  alpha,
  plot = TRUE,
  verbose = TRUE
)
```

Arguments

n	sample size (pairs)
m1	mean of group 1
m2	mean of group 2
sd1	standard deviation of group 1
sd2	standard deviation of group 2
r12	correlation of dependent variable between group 1 and group 2
low_eqbound_dz	lower equivalence bounds (e.g., -0.5) expressed in standardized mean difference (Cohen's dz)
high_eqbound_dz	upper equivalence bounds (e.g., 0.5) expressed in standardized mean difference (Cohen's dz)
alpha	alpha level (default = 0.05)
plot	set whether results should be plotted (plot = TRUE) or not (plot = FALSE) - defaults to TRUE
verbose	logical variable indicating whether text output should be generated (verbose = TRUE) or not (verbose = FALSE) - default to TRUE

Value

Returns TOST t-value 1, TOST p-value 1, TOST t-value 2, TOST p-value 2, degrees of freedom, low equivalence bound, high equivalence bound, low equivalence bound in dz, high equivalence bound in dz, Lower limit confidence interval TOST, Upper limit confidence interval TOST

References

Mara, C. A., & Cribbie, R. A. (2012). Paired-Samples Tests of Equivalence. *Communications in Statistics - Simulation and Computation*, 41(10), 1928-1943. <https://doi.org/10.1080/03610918.2011.626545>, formula page 1932. Note there is a typo in the formula: n-1 should be n (personal communication, 31-8-2016)

Examples

```
## Test means of 5.83 and 5.75, standard deviations of 1.17 and 1.29 in sample of 65 pairs
## with correlation between observations of 0.75 using equivalence bounds in Cohen's dz of
## -0.4 and 0.4 (with default alpha setting of = 0.05).
TOSTpaired(n=65,m1=5.83,m2=5.75,sd1=1.17,sd2=1.29,r12=0.75,low_eqbound_dz=-0.4,high_eqbound_dz=0.4)
```

TOSTpaired.raw *TOST function for a dependent t-test (raw scores)*

Description

TOST function for a dependent t-test (raw scores)

Usage

```
TOSTpaired.raw(
  n,
  m1,
  m2,
  sd1,
  sd2,
  r12,
  low_eqbound,
  high_eqbound,
  alpha,
  plot = TRUE,
  verbose = TRUE
)
```

Arguments

n	sample size (pairs)
m1	mean of group 1
m2	mean of group 2
sd1	standard deviation of group 1
sd2	standard deviation of group 2
r12	correlation of dependent variable between group 1 and group 2
low_eqbound	lower equivalence bounds (e.g., -0.5) expressed in raw scores
high_eqbound	upper equivalence bounds (e.g., 0.5) expressed in raw scores
alpha	alpha level (default = 0.05)
plot	set whether results should be plotted (plot = TRUE) or not (plot = FALSE) - defaults to TRUE
verbose	logical variable indicating whether text output should be generated (verbose = TRUE) or not (verbose = FALSE) - default to TRUE

Value

Returns TOST t-value 1, TOST p-value 1, TOST t-value 2, TOST p-value 2, degrees of freedom, low equivalence bound, high equivalence bound, Lower limit confidence interval TOST, Upper limit confidence interval TOST

References

Mara, C. A., & Cribbie, R. A. (2012). Paired-Samples Tests of Equivalence. *Communications in Statistics - Simulation and Computation*, 41(10), 1928-1943. <https://doi.org/10.1080/03610918.2011.626545>, formula page 1932. Note there is a typo in the formula: n-1 should be n (personal communication, 31-8-2016)

Examples

```
## Test means of 5.83 and 5.75, standard deviations of 1.17 and 1.30 in sample of 65 pairs
## with correlation between observations of 0.745 using equivalence bounds in raw units of
## -0.34 and 0.34, (with default alpha setting of = 0.05).
TOSTpaired.raw(n=65,m1=5.83,m2=5.75,sd1=1.17,sd2=1.30,r12=0.745,low_eqbound=-0.34,high_eqbound=0.34)
```

TOSTr *TOST function for a correlations*

Description

TOST function for a correlations

Usage

```
TOSTr(n, r, low_eqbound_r, high_eqbound_r, alpha, plot = TRUE, verbose = TRUE)
```

Arguments

n	number of pairs of observations
r	observed correlation
low_eqbound_r	lower equivalence bounds (e.g., -0.3) expressed in a correlation effect size
high_eqbound_r	upper equivalence bounds (e.g., 0.3) expressed in a correlation effect size
alpha	alpha level (default = 0.05)
plot	set whether results should be plotted (plot = TRUE) or not (plot = FALSE) - defaults to TRUE
verbose	logical variable indicating whether text output should be generated (verbose = TRUE) or not (verbose = FALSE) - default to TRUE

Value

Returns TOST p-value 1, TOST p-value 2, alpha, low equivalence bound r, high equivalence bound r, Lower limit confidence interval TOST, Upper limit confidence interval TOST

References

Goertzen, J. R., & Cribbie, R. A. (2010). Detecting a lack of association: An equivalence testing approach. *British Journal of Mathematical and Statistical Psychology*, 63(3), 527-537. <https://doi.org/10.1348/000711009X475>, formula page 531.

Examples

```
TOSTr(n=100, r = 0.02, low_eqbound_r=-0.3, high_eqbound_r=0.3, alpha=0.05)
```

TOSTt-methods

Methods for TOSTt objects

Description

Methods defined for objects returned from the `t_TOST` and `boot_t_TOST` functions.

Usage

```
## S3 method for class 'TOSTt'
print(x, digits = 4, ...)

## S3 method for class 'TOSTt'
plot(
  x,
  type = c("cd", "c", "tnull"),
  estimates = c("raw", "SMD"),
  ci_lines,
  ci_shades,
  ...
)

describe(x, ...)

## S3 method for class 'TOSTt'
describe(x, digits = 3, ...)
```

Arguments

<code>x</code>	object of class <code>TOSTt</code>
<code>digits</code>	Number of digits to print for p-values
<code>...</code>	further arguments passed through, see description of return value for details. TOSTt-methods .
<code>type</code>	Type of plot to produce. Default is a consonance density plot "cd". Consonance plots (<code>type = "cd"</code>) and null distribution plots (<code>type = "tnull"</code>) can also be produced. Note: null distribution plots only available for <code>estimates = "raw"</code> .
<code>estimates</code>	indicator of what estimates to plot; options include "raw" or "SMD". Default is both: <code>c("raw","SMD")</code> .
<code>ci_lines</code>	Confidence interval lines for plots. Default is $1-\alpha*2$ (e.g., $\alpha = 0.05$ is 90%)
<code>ci_shades</code>	Confidence interval shades when plot type is "cd".

Value

print Prints short summary of the tests.
 plot Returns a plot of the effects.
 describe Verbose description of results.

Examples

```
# example code
# Print

res1 = t_TOST(mpg ~ am, data = mtcars, eqb = 3)

res1
# Print with more digits
print(res1, digits = 6)
## Not run:
# Plot with density plot - only raw values (SLOW)
plot(res1, type = "cd", estimates = "raw")
# Plot with consonance - only raw values (SLOW)
plot(res1, type = "c", estimates = "raw")
# Plot null distribution - only raw values
plot(res1, type = "tnull", estimates = "raw")

## End(Not run)
# Get description of the results
describe(res1)
```

TOSTtwo

TOST function for an independent t-test (Cohen's d)

Description

TOST function for an independent t-test (Cohen's d)

Usage

```
TOSTtwo(
  m1,
  m2,
  sd1,
  sd2,
  n1,
  n2,
  low_eqbound_d,
  high_eqbound_d,
  alpha,
  var.equal,
```

```

    plot = TRUE,
    verbose = TRUE
  )

```

Arguments

m1	mean of group 1
m2	mean of group 2
sd1	standard deviation of group 1
sd2	standard deviation of group 2
n1	sample size in group 1
n2	sample size in group 2
low_eqbound_d	lower equivalence bounds (e.g., -0.5) expressed in standardized mean difference (Cohen's d)
high_eqbound_d	upper equivalence bounds (e.g., 0.5) expressed in standardized mean difference (Cohen's d)
alpha	alpha level (default = 0.05)
var.equal	logical variable indicating whether equal variances assumption is assumed to be TRUE or FALSE. Defaults to FALSE.
plot	set whether results should be plotted (plot = TRUE) or not (plot = FALSE) - defaults to TRUE
verbose	logical variable indicating whether text output should be generated (verbose = TRUE) or not (verbose = FALSE) - default to TRUE

Value

Returns TOST t-value 1, TOST p-value 1, TOST t-value 2, TOST p-value 2, degrees of freedom, low equivalence bound, high equivalence bound, low equivalence bound in Cohen's d, high equivalence bound in Cohen's d, Lower limit confidence interval TOST, Upper limit confidence interval TOST

References

Berger, R. L., & Hsu, J. C. (1996). Bioequivalence Trials, Intersection-Union Tests and Equivalence Confidence Sets. *Statistical Science*, 11(4), 283-302.

Gruman, J. A., Cribbie, R. A., & Arpin-Cribbie, C. A. (2007). The effects of heteroscedasticity on tests of equivalence. *Journal of Modern Applied Statistical Methods*, 6(1), 133-140, formula for Welch's t-test on page 135

Examples

```

## Eskine (2013) showed that participants who had been exposed to organic
## food were substantially harsher in their moral judgments relative to
## those exposed to control (d = 0.81, 95% CI: [0.19, 1.45]). A
## replication by Moery & Calin-Jageman (2016, Study 2) did not observe
## a significant effect (Control: n = 95, M = 5.25, SD = 0.95, Organic
## Food: n = 89, M = 5.22, SD = 0.83). Following Simonsohn's (2015)

```



```
## recommendation the equivalence bound was set to the effect size the
## original study had 33% power to detect (with n = 21 in each condition,
## this means the equivalence bound is d = 0.48, which equals a
## difference of 0.384 on a 7-point scale given the sample sizes and a
## pooled standard deviation of 0.894). Using a TOST equivalence test
## with default alpha = 0.05, not assuming equal variances, and equivalence
## bounds of d = -0.43 and d = 0.43 is significant, t(182) = -2.69,
## p = 0.004. We can reject effects larger than d = 0.43.
```

```
TOSTtwo(m1=5.25,m2=5.22,sd1=0.95,sd2=0.83,n1=95,n2=89,low_eqbound_d=-0.43,high_eqbound_d=0.43)
```

TOSTtwo.prop

TOST function for two proportions (raw scores)

Description

TOST function for two proportions (raw scores)

Usage

```
TOSTtwo.prop(
  prop1,
  prop2,
  n1,
  n2,
  low_eqbound,
  high_eqbound,
  alpha,
  ci_type = "normal",
  plot = TRUE,
  verbose = TRUE
)
```

Arguments

prop1	proportion of group 1
prop2	proportion of group 2
n1	sample size in group 1
n2	sample size in group 2
low_eqbound	lower equivalence bounds (e.g., -0.1) expressed in proportions
high_eqbound	upper equivalence bounds (e.g., 0.1) expressed in proportions
alpha	alpha level (default = 0.05)
ci_type	confidence interval type (default = "normal"). "wilson" produces Wilson score intervals with a Yates continuity correction while "normal" calculates the simple asymptotic method with no continuity correction.

plot set whether results should be plotted (plot = TRUE) or not (plot = FALSE) - defaults to TRUE

verbose logical variable indicating whether text output should be generated (verbose = TRUE) or not (verbose = FALSE) - default to TRUE

Value

Returns TOST z-value 1, TOST p-value 1, TOST z-value 2, TOST p-value 2, low equivalence bound, high equivalence bound, Lower limit confidence interval TOST, Upper limit confidence interval TOST

References

Tunes da Silva, G., Logan, B. R., & Klein, J. P. (2008). Methods for Equivalence and Noninferiority Testing. *Biology of Blood Marrow Transplant*, 15(1 Suppl), 120-127. Yin, G. (2012). *Clinical Trial Design: Bayesian and Frequentist Adaptive Methods*. Hoboken, New Jersey: John Wiley & Sons, Inc.

Examples

```
## Equivalence test for two independent proportions equal to .65 and .70, with 100 samples
## per group, lower equivalence bound of -0.1, higher equivalence bound of 0.1, and alpha of 0.05.

TOSTtwo.prop(prop1 = .65, prop2 = .70, n1 = 100, n2 = 100,
             low_eqbound = -0.1, high_eqbound = 0.1, alpha = .05)
```

TOSTtwo.raw

TOST function for an independent t-test (raw scores)

Description

TOST function for an independent t-test (raw scores)

Usage

```
TOSTtwo.raw(
  m1,
  m2,
  sd1,
  sd2,
  n1,
  n2,
  low_eqbound,
  high_eqbound,
  alpha,
  var.equal,
  plot = TRUE,
```

```

    verbose = TRUE
  )

```

Arguments

m1	mean of group 1
m2	mean of group 2
sd1	standard deviation of group 1
sd2	standard deviation of group 2
n1	sample size in group 1
n2	sample size in group 2
low_eqbound	lower equivalence bounds (e.g., -0.5) expressed in raw scale units (e.g., scale-points)
high_eqbound	upper equivalence bounds (e.g., 0.5) expressed in raw scale units (e.g., scale-points)
alpha	alpha level (default = 0.05)
var.equal	logical variable indicating whether equal variances assumption is assumed to be TRUE or FALSE. Defaults to FALSE.
plot	set whether results should be plotted (plot = TRUE) or not (plot = FALSE) - defaults to TRUE
verbose	logical variable indicating whether text output should be generated (verbose = TRUE) or not (verbose = FALSE) - default to TRUE

Value

Returns TOST t-value 1, TOST p-value 1, TOST t-value 2, TOST p-value 2, degrees of freedom, low equivalence bound, high equivalence bound, Lower limit confidence interval TOST, Upper limit confidence interval TOST

References

Berger, R. L., & Hsu, J. C. (1996). Bioequivalence Trials, Intersection-Union Tests and Equivalence Confidence Sets. *Statistical Science*, 11(4), 283-302.

Gruman, J. A., Cribbie, R. A., & Arpin-Cribbie, C. A. (2007). The effects of heteroscedasticity on tests of equivalence. *Journal of Modern Applied Statistical Methods*, 6(1), 133-140, formula for Welch's t-test on page 135

Examples

```

## Eskine (2013) showed that participants who had been exposed to organic
## food were substantially harsher in their moral judgments relative to
## those exposed to control (d = 0.81, 95% CI: [0.19, 1.45]). A
## replication by Moery & Calin-Jageman (2016, Study 2) did not observe
## a significant effect (Control: n = 95, M = 5.25, SD = 0.95, Organic
## Food: n = 89, M = 5.22, SD = 0.83). Following Simonsohn's (2015)
## recommendation the equivalence bound was set to the effect size the

```

```
## original study had 33% power to detect (with n = 21 in each condition,
## this means the equivalence bound is d = 0.48, which equals a
## difference of 0.384 on a 7-point scale given the sample sizes and a
## pooled standard deviation of 0.894). Using a TOST equivalence test
## with alpha = 0.05, assuming equal variances, and equivalence
## bounds of d = -0.43 and d = 0.43 is significant, t(182) = -2.69,
## p = 0.004. We can reject effects larger than d = 0.43.
```

```
TOSTtwo.raw(m1=5.25,m2=5.22,sd1=0.95,sd2=0.83,n1=95,n2=89,low_eqbound=-0.384,high_eqbound=0.384)
```

tsum_TOST

TOSTt with Summary Statistics

Description

A function for TOST with all types of t-tests from summary statistics.

Usage

```
tsum_TOST(
  m1,
  sd1,
  n1,
  m2 = NULL,
  sd2 = NULL,
  n2 = NULL,
  r12 = NULL,
  hypothesis = c("EQU", "MET"),
  paired = FALSE,
  var.equal = FALSE,
  eqb,
  low_eqbound,
  high_eqbound,
  mu = 0,
  eqbound_type = c("raw", "SMD"),
  alpha = 0.05,
  bias_correction = TRUE,
  rm_correction = FALSE,
  glass = NULL,
  smd_ci = c("nct", "goulet", "t", "z")
)
```

Arguments

m1	mean of group 1
sd1	standard deviation of group 1
n1	sample size in group 1

m2	mean of group 2
sd2	standard deviation of group 2
n2	sample size in group 2
r12	correlation of dependent variable between group 1 and group 2
hypothesis	'EQU' for equivalence (default), or 'MET' for minimal effects test, the alternative hypothesis.
paired	a logical indicating whether you want a paired t-test.
var.equal	a logical variable indicating whether to treat the two variances as being equal. If TRUE then the pooled variance is used to estimate the variance otherwise the Welch (or Satterthwaite) approximation to the degrees of freedom is used.
eqb	Equivalence bound. Can provide 1 value (negative value is taken as the lower bound) or 2 specific values that represent the upper and lower equivalence bounds.
low_eqbound	lower equivalence bounds (deprecated).
high_eqbound	upper equivalence bounds (deprecated).
mu	a number indicating the true value of the mean for the two tailed test (or difference in means if you are performing a two sample test).
eqbound_type	Type of equivalence bound. Can be set to "SMD" for standardized mean difference (i.e., Cohen's d) or "raw" for the mean difference. Default is "raw". Raw is strongly recommended as SMD bounds will produce biased results.
alpha	alpha level (default = 0.05)
bias_correction	Apply Hedges' correction for bias (default is TRUE).
rm_correction	Repeated measures correction to make standardized mean difference Cohen's d(rm). This only applies to repeated/paired samples. Default is FALSE.
glass	A option to calculate Glass's delta as an alternative to Cohen's d type SMD. Default is NULL to not calculate Glass's delta, "glass1" will use the first group's SD as the denominator whereas "glass2" will use the 2nd group's SD.
smd_ci	Method for calculating SMD confidence intervals. Methods include Goulet, noncentral t (nct), central t (t), and normal method (z).

Details

For details on the calculations in this function see vignette("IntroTOSTt") & vignette("SMD_calcs").

Value

An S3 object of class "TOSTt" is returned containing the following slots:

"TOST" A table of class "data.frame" containing two-tailed t-test and both one-tailed results.

"eqb" A table of class "data.frame" containing equivalence bound settings.

"effsize" table of class "data.frame" containing effect size estimates

"hypothesis" String stating the hypothesis being tested

"smd" List containing the results of the standardized mean difference calculations (e.g., Cohen's d). Items include: d (estimate), dlow (lower CI bound), dhigh (upper CI bound), d_df (degrees of freedom for SMD), d_sigma (SE), d_lambda (non-centrality), J (bias correction), smd_label (type of SMD), d_denom (denominator calculation)

"alpha" Alpha level set for the analysis.

"method" Type of t-test.

"decision" List included text regarding the decisions for statistical inference.

See Also

Other TOST: [simple_hstest\(\)](#), [t_TOST\(\)](#)

Examples

```
# example code
# One sample test
tsum_TOST(m1 = 0.55, n1 = 18, sd1 = 4, eqb = 2)
```

t_TOST

TOST with t-tests

Description

A function for TOST with all types of t-tests.

Usage

```
t_TOST(
  x,
  ...,
  hypothesis = "EQU",
  paired = FALSE,
  var.equal = FALSE,
  eqb,
  low_eqbound,
  high_eqbound,
  eqbound_type = "raw",
  alpha = 0.05,
  bias_correction = TRUE,
  rm_correction = FALSE,
  glass = NULL,
  smd_ci = c("nct", "goulet", "t", "z")
)

## Default S3 method:
t_TOST(
  x,
```

```

y = NULL,
hypothesis = c("EQU", "MET"),
paired = FALSE,
var.equal = FALSE,
eqb,
low_eqbound,
high_eqbound,
eqbound_type = c("raw", "SMD"),
alpha = 0.05,
mu = 0,
bias_correction = TRUE,
rm_correction = FALSE,
glass = NULL,
smd_ci = c("nct", "goulet", "t", "z"),
...
)

## S3 method for class 'formula'
t_TOST(formula, data, subset, na.action, ...)

```

Arguments

x	a (non-empty) numeric vector of data values.
...	further arguments to be passed to or from methods.
hypothesis	'EQU' for equivalence (default), or 'MET' for minimal effects test, the alternative hypothesis.
paired	a logical indicating whether you want a paired t-test.
var.equal	a logical variable indicating whether to treat the two variances as being equal. If TRUE then the pooled variance is used to estimate the variance otherwise the Welch (or Satterthwaite) approximation to the degrees of freedom is used.
eqb	Equivalence bound. Can provide 1 value (negative value is taken as the lower bound) or 2 specific values that represent the upper and lower equivalence bounds.
low_eqbound	lower equivalence bounds (deprecated).
high_eqbound	upper equivalence bounds (deprecated).
eqbound_type	Type of equivalence bound. Can be set to "SMD" for standardized mean difference (i.e., Cohen's d) or "raw" for the mean difference. Default is "raw". Raw is strongly recommended as SMD bounds will produce biased results.
alpha	alpha level (default = 0.05)
bias_correction	Apply Hedges' correction for bias (default is TRUE).
rm_correction	Repeated measures correction to make standardized mean difference Cohen's d(rm). This only applies to repeated/paired samples. Default is FALSE.
glass	A option to calculate Glass's delta as an alternative to Cohen's d type SMD. Default is NULL to not calculate Glass's delta, "glass1" will use the first group's SD as the denominator whereas "glass2" will use the 2nd group's SD.

smd_ci	Method for calculating SMD confidence intervals. Methods include Goulet, noncentral t (nct), central t (t), and normal method (z).
y	an optional (non-empty) numeric vector of data values.
mu	a number indicating the true value of the mean for the two tailed test (or difference in means if you are performing a two sample test).
formula	a formula of the form lhs ~ rhs where lhs is a numeric variable giving the data values and rhs either 1 for a one-sample or paired test or a factor with two levels giving the corresponding groups. If lhs is of class "Pair" and rhs is 1, a paired test is done.
data	an optional matrix or data frame (or similar: see model.frame) containing the variables in the formula formula. By default the variables are taken from environment(formula).
subset	an optional vector specifying a subset of observations to be used.
na.action	a function which indicates what should happen when the data contain NAs. Defaults to getOption("na.action").

Details

For details on the calculations in this function see vignette("IntroTOSTt") & vignette("SMD_calcs").

Value

An S3 object of class "TOSTt" is returned containing the following slots:

"TOST" A table of class "data.frame" containing two-tailed t-test and both one-tailed results.

"eqb" A table of class "data.frame" containing equivalence bound settings.

"effsize" table of class "data.frame" containing effect size estimates

"hypothesis" String stating the hypothesis being tested

"smd" List containing the results of the standardized mean difference calculations (e.g., Cohen's d). Items include: d (estimate), dlow (lower CI bound), dhigh (upper CI bound), d_df (degrees of freedom for SMD), d_sigma (SE), d_lambda (non-centrality), J (bias correction), smd_label (type of SMD), d_denom (denominator calculation)

"alpha" Alpha level set for the analysis.

"method" Type of t-test.

"decision" List included text regarding the decisions for statistical inference.

See Also

Other TOST: [simple_hptest\(\)](#), [tsum_TOST\(\)](#)

Examples

```
data(mtcars)
t_TOST(mpg ~ am,
data = mtcars,
eqb = 3)
```

 wilcox_TOST

TOST with Wilcoxon-Mann-Whitney tests

Description

A function for TOST using the non-parametric methods of the Wilcoxon-Mann-Whitney family of tests. This function uses the normal approximation and applies a continuity correction automatically.

Usage

```
wilcox_TOST(
  x,
  ...,
  hypothesis = "EQU",
  paired = FALSE,
  eqb,
  low_eqbound,
  high_eqbound,
  ses = "rb",
  alpha = 0.05
)

## Default S3 method:
wilcox_TOST(
  x,
  y = NULL,
  hypothesis = "EQU",
  paired = FALSE,
  eqb,
  low_eqbound,
  high_eqbound,
  ses = c("rb", "odds", "cstat"),
  alpha = 0.05,
  mu = 0,
  ...
)

## S3 method for class 'formula'
wilcox_TOST(formula, data, subset, na.action, ...)
```

Arguments

x	a (non-empty) numeric vector of data values.
...	further arguments to be passed to or from methods.
hypothesis	'EQU' for equivalence (default), or 'MET' for minimal effects test, the alternative hypothesis.

<code>paired</code>	a logical indicating whether you want a paired t-test.
<code>eqb</code>	Equivalence bound. Can provide 1 value (negative value is taken as the lower bound) or 2 specific values that represent the upper and lower equivalence bounds.
<code>low_eqbound</code>	lower equivalence bounds (deprecated).
<code>high_eqbound</code>	upper equivalence bounds (deprecated).
<code>ses</code>	Standardized effect size. Default is "rb" for rank-biserial correlation. Options also include "cstat" for concordance probability, or "odds" for Wilcoxon-Mann-Whitney odds (otherwise known as Agresti's generalized odds ratio).
<code>alpha</code>	alpha level (default = 0.05)
<code>y</code>	an optional (non-empty) numeric vector of data values.
<code>mu</code>	number indicating the value around which (a-)symmetry (for one-sample or paired samples) or shift (for independent samples) is to be estimated. See <code>[stats::wilcox.test]</code> .
<code>formula</code>	a formula of the form <code>lhs ~ rhs</code> where <code>lhs</code> is a numeric variable giving the data values and <code>rhs</code> either 1 for a one-sample or paired test or a factor with two levels giving the corresponding groups. If <code>lhs</code> is of class "Pair" and <code>rhs</code> is 1, a paired test is done.
<code>data</code>	an optional matrix or data frame (or similar: see <code>model.frame</code>) containing the variables in the formula <code>formula</code> . By default the variables are taken from <code>environment(formula)</code> .
<code>subset</code>	an optional vector specifying a subset of observations to be used.
<code>na.action</code>	a function which indicates what should happen when the data contain NAs. Defaults to <code>getOption("na.action")</code> .

Details

For details on the calculations in this function see `vignette("robustTOST")`.

Value

An S3 object of class "TOSTnp" is returned containing the following slots:

"TOST" A table of class "data.frame" containing two-tailed wilcoxon signed rank test and both one-tailed results.

"eqb" A table of class "data.frame" containing equivalence bound settings.

"effsize" table of class "data.frame" containing effect size estimates.

"hypothesis" String stating the hypothesis being tested.

"smd" List containing information on standardized effect size.

"alpha" Alpha level set for the analysis.

"method" Type of non-parametric test.

"decision" List included text regarding the decisions for statistical inference.

References

David F. Bauer (1972). Constructing confidence sets using rank statistics. *Journal of the American Statistical Association* 67, 687–690. doi: 10.1080/01621459.1972.10481279.

Myles Hollander and Douglas A. Wolfe (1973). *Nonparametric Statistical Methods*. New York: John Wiley & Sons. Pages 27–33 (one-sample), 68–75 (two-sample). Or second edition (1999).

See Also

Other Robust TOST: [boot_log_TOST\(\)](#), [boot_t_TOST\(\)](#), [boot_t_test\(\)](#), [log_TOST\(\)](#)

Examples

```
data(mtcars)
wilcox_TOST(mpg ~ am,
data = mtcars,
eqb = 3)
```

z_cor_test

Test for Association/Correlation Between Paired Samples

Description

Test for association between paired samples, using one of Pearson's product moment correlation coefficient, Kendall's τ or Spearman's ρ . Unlike the stats version of `cor.test`, this function allows users to set the null to a value other than zero.

Usage

```
z_cor_test(
  x,
  y,
  alternative = c("two.sided", "less", "greater", "equivalence", "minimal.effect"),
  method = c("pearson", "kendall", "spearman"),
  alpha = 0.05,
  null = 0
)
```

Arguments

x, y	numeric vectors of data values. x and y must have the same length.
alternative	a character string specifying the alternative hypothesis, must be one of "two.sided" (default), "greater", "less", "equivalence" (TOST), or "minimal.effect" (TOST). You can specify just the initial letter.
method	a character string indicating which correlation coefficient is to be used for the test. One of "pearson", "kendall", or "spearman", can be abbreviated.
alpha	alpha level (default = 0.05)
null	a number indicating the null hypothesis. Default is a correlation of zero.

Details

This function uses Fisher's z transformation for the correlations, but uses Fieller's correction of the standard error for Spearman's rho and Kendall's tau.

Value

A list with class "htest" containing the following components:

"p.value" the p-value of the test.

"estimate" the estimated measure of association, with name "pb", "wincor", "cor", "tau", or "rho" corresponding to the method employed.

"null.value" the value of the association measure under the null hypothesis.

"alternative" character string indicating the alternative hypothesis (the value of the input argument alternative).

"method" a character string indicating how the association was measured.

"data.name" a character string giving the names of the data.

"call" the matched call.

References

Goertzen, J. R., & Cribbie, R. A. (2010). Detecting a lack of association: An equivalence testing approach. *British Journal of Mathematical and Statistical Psychology*, 63(3), 527-537. <https://doi.org/10.1348/000711009X475> formula page 531.

Examples

```
# example code
x <- c(44.4, 45.9, 41.9, 53.3, 44.7, 44.1, 50.7, 45.2, 60.1)
y <- c( 2.6,  3.1,  2.5,  5.0,  3.6,  4.0,  5.2,  2.8,  3.8)
# Sig test
z_cor_test(x, y, method = "kendall", alternative = "t", null = 0)
# MET test
z_cor_test(x, y, method = "kendall", alternative = "min", null = .2)
```

Index

- * **Robust TOST**
 - boot_log_TOST, 8
 - boot_t_test, 10
 - boot_t_TOST, 12
 - log_TOST, 30
 - wilcox_TOST, 73
 - * **TOST**
 - simple_htest, 50
 - t_TOST, 70
 - tsum_TOST, 68
 - * **compare studies**
 - boot_compare_smd, 5
 - compare_smd, 15
 - * **datasets**
 - hawthorne, 29
 - * **effect sizes**
 - ses_calc, 49
 - smd_calc, 52
 - * **f-test**
 - equ_anova, 26
 - equ_ftest, 27
 - * **htest**
 - as_htest, 3
 - htest-helpers, 29
 - simple_htest, 50
 - * **plotting functions**
 - plot_cor, 32
 - plot_pes, 33
 - plot_smd, 34
 - * **power**
 - power_eq_f, 45
 - power_t_TOST, 46
- as.data.frame, 19, 21, 23, 24, 26
- as_htest, 3, 30, 52
- boot_compare_cor, 4
- boot_compare_smd, 5, 17
- boot_cor_test, 6
- boot_log_TOST, 8, 11, 14, 32, 75
- boot_t_test, 10, 10, 14, 32, 75
- boot_t_TOST, 10, 11, 12, 32, 75
- compare_cor, 14
- compare_smd, 6, 15
- corsum_test, 17
- dataTOSTone, 18
- dataTOSTpaired, 20
- dataTOSTr, 22
- dataTOSTtwo, 23
- datatosttwoprop, 25
- describe (TOSTt-methods), 62
- describe.TOSTnp (TOSTnp-methods), 55
- describe_htest (htest-helpers), 29
- df_htest (htest-helpers), 29
- equ_anova, 26, 28
- equ_ftest, 27, 27
- extract_r_paired, 28
- hawthorne, 29
- htest-helpers, 29
- log_TOST, 10, 11, 14, 30, 75
- np_ses (rbs), 47
- plot.TOSTt (TOSTt-methods), 62
- plot_cor, 32, 34, 35
- plot_pes, 33, 33, 35
- plot_smd, 33, 34, 34
- power_eq_f, 45, 47
- power_t_TOST, 45, 46
- powerTOSTone, 35
- powerTOSTone.raw, 36
- powerTOSTpaired, 37
- powerTOSTpaired.raw, 38
- powerTOSTr, 40
- powerTOSTtwo, 41
- powerTOSTtwo.prop, 42

powerTOSTtwo.raw, 43
print.TOSTnp (TOSTnp-methods), 55
print.TOSTt (TOSTt-methods), 62

rbs, 47

ses_calc, 49, 54
simple_hstest, 3, 30, 50, 70, 72
smd_calc, 50, 52

t_TOST, 52, 70, 70
TOSTmeta, 54
TOSTnp-methods, 55
TOSTone, 56
TOSTone.raw, 57
TOSTpaired, 58
TOSTpaired.raw, 60
TOSTr, 61
TOSTt-methods, 62
TOSTtwo, 63
TOSTtwo.prop, 65
TOSTtwo.raw, 66
tsum_TOST, 52, 68, 72

wilcox_TOST, 10, 11, 14, 32, 73

z_cor_test, 75