PREPRINT

Exploring Equivalence Testing with the Updated TOSTER R Package

Aaron R. Caldwell[a]

[a]Natick, MA, https://orcid.org/0000-0002-4541-6283

ARTICLE HISTORY
Compiled November 13, 2022

ABSTRACT
This is an article detailing the "avocado TOST" update to the TOSTER R package.

KEYWORDS
statistics, bootstrapping, minimal effects test, NHST, TOST

1. Introduction

Researchers often erroneously declare that no statistical effect exists based on a single "non-significant" p-value (Altman and Bland 1995). In many of these cases, the data may corroborate the researchers claim but the interpretation of a null hypothesis significance test (NHST), wherein the null hypothesis is "no effect", is nonetheless incorrect. In order to statistically test for whether there is "no effect" or "no difference" researchers could explore using equivalence testing. A very simple equivalence testing approach is the use of "two one-sided tests" (TOST) (Schuirmann 1987). In TOST procedures, an upper (Δ_U) and lower (Δ_L) equivalence bound is specified based on the smallest effect size of interest (SESOI). If the TOST is below a pre-specified alpha level, then the effect can be considered close enough to zero to be practically equivalent (Lakens 2017).

Both the complaints about erroneous conclusions regarding equivalence (Altman and Bland 1995) and proposed statistical solutions (Schuirmann 1987) have existed for decades now. Yet the problem appears to persist in many applied disciplines. I estimate that the cause of this continued dissonance is due to a lack of education on equivalence testing and struggle for many applied researchers to implement equivalence testing. In my experience, most researchers have received some degree of statistical training in their doctoral or master's studies, but it is rare that any have idea of equivalence testing. It may also be difficult to implement equivalence testing for many researchers. This may be caused by most statistical software defaulting to a null hypothesis of zero, or even completely lacking an ability to change the null hypothesis.

The TOSTER R package was originally developed in by Lakens (2017) to introduce experimental psychologists to the concept of equivalence testing and provide an easy-to-use implementation in R. Since then I have made a significant update to the

CONTACT Aaron R. Caldwell. Email: arcaldwell49@gmail.com

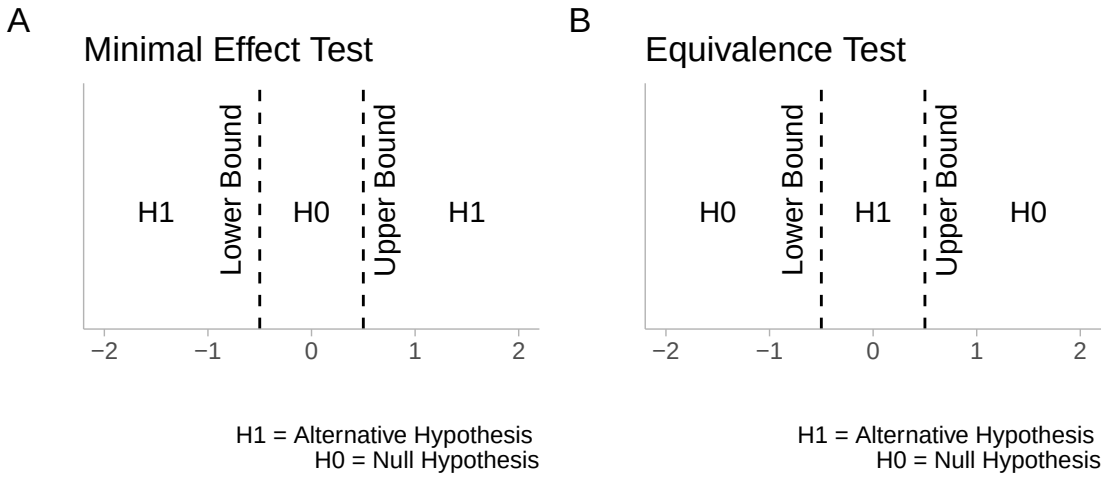

Figure 1. Type of Hypothesis

package in order to improve the user interface and expand the tools available within the package.

An experienced R programmer may have no problem performing equivalence testing within R but beginners may struggle with both writing the code and interpreting the output.

2. Basics of Equivalence Testing

2.1. *The TOSTER R Package*

In an effort to make `TOSTER` more informative and easier to use, a new function `t_TOST` was created. This function operates very similarly to base R's `t.test` function, but performs 3 t-tests (one two-tailed and two one-tailed tests). In addition, this function has a generic method where two vectors can be supplied or a formula can be given (e.g.,`y ~ group`). This function also makes it easier to switch between types of t-tests. All three types (two sample, one sample, and paired samples) can be performed/calculated from the same function. Moreover, the output from this function is verbose, and should make the decisions derived from the function more informative and user-friendly.

Also, `t_TOST` is not limited to equivalence tests. Minimal effects testing (MET) is possible. MET is useful for situations where the hypothesis is about a minimal effect and the null hypothesis *is* equivalence @ref(fig:hypplot) .

2.2. *Vignettes with TOSTER*

In the general introduction to this package we detailed how to look at *old* results and how to apply TOST to interpreting those results. However, in many cases, users may have new data that needs to be analyzed. Therefore, `t_TOST` can be applied to new data. This vignette will use the `bugs` data from the `jmv` R package and the `sleep` data.

```
data('sleep')
library(jmv)
data('bugs')
```

2.2.1. Independent Groups

For this example, we will use the sleep data. In this data there is a `group` variable and an outcome `extra`.

```
head(sleep,2)
```

```
##   extra group ID
## 1   0.7     1  1
## 2  -1.6     1  2
```

We will assume the data are independent, and that we have equivalence bounds of +/- 0.5. All we need to do is provide the `formula`, `data`, and `eqb` arguments for the function to run appropriately. In addition, we can set the `var.equal` argument (to assume equal variance), and the `paired` argument (sets if the data is paired or not). Both are logical indicators that can be set to TRUE or FALSE. The `alpha` is automatically set to 0.05 but this can also be adjusted by the user. Hedges's correction (Hedges 1981) is also automatically calculated, but this can be overridden with the `bias_correction` argument[1]. The `hypothesis` is automatically set to "EQU" for equivalence but if a minimal effect is of interest then "MET" can be supplied.

```
# Formula Interface
res1 = t_TOST(formula = extra ~ group, data = sleep,
              eqb = .5, smd_ci = "t")
# x & y Interface
res1a = t_TOST(x = subset(sleep,group==1)$extra,
               y = subset(sleep,group==2)$extra, eqb =.5)
```

[1] Glass's delta can also be produced in the output by using the `glass` argument

Once the function has run, we can print the results with the `print` method. This provides a verbose summary of the results.

```
print(res1)
```

```
##
## Welch Two Sample t-test
##
## The equivalence test was non-significant, t(17.78) = -1.272, p = 8.9e-01
## The null hypothesis test was non-significant, t(17.78) = -1.861, p = 7.94e-02
## NHST: don't reject null significance hypothesis that the effect is equal to zero
## TOST: don't reject null equivalence hypothesis
##
## TOST Results
##                 t    df p.value
## t-test     -1.861 17.78   0.079
## TOST Lower -1.272 17.78   0.890
## TOST Upper -2.450 17.78   0.012
##
## Effect Sizes
##                Estimate     SE               C.I. Conf. Level
## Raw             -1.5800 0.8491 [-3.0534, -0.1066]         0.9
## Hedges's g(av)  -0.7965 0.5992  [-1.8362, 0.2433]         0.9
## Note: SMD confidence intervals are an approximation. See vignette("SMD_calcs").
```

Another nice feature is the generic `plot` method that can provide a visual summary of the results. All of the plots in this package were inspired by the concurve R package. There are two types of plots that can be produced. The first, and default, is the consonance density plot (`type = "cd"`) @ref(fig:cdplot).

```
plot(res1, type = "cd")
```

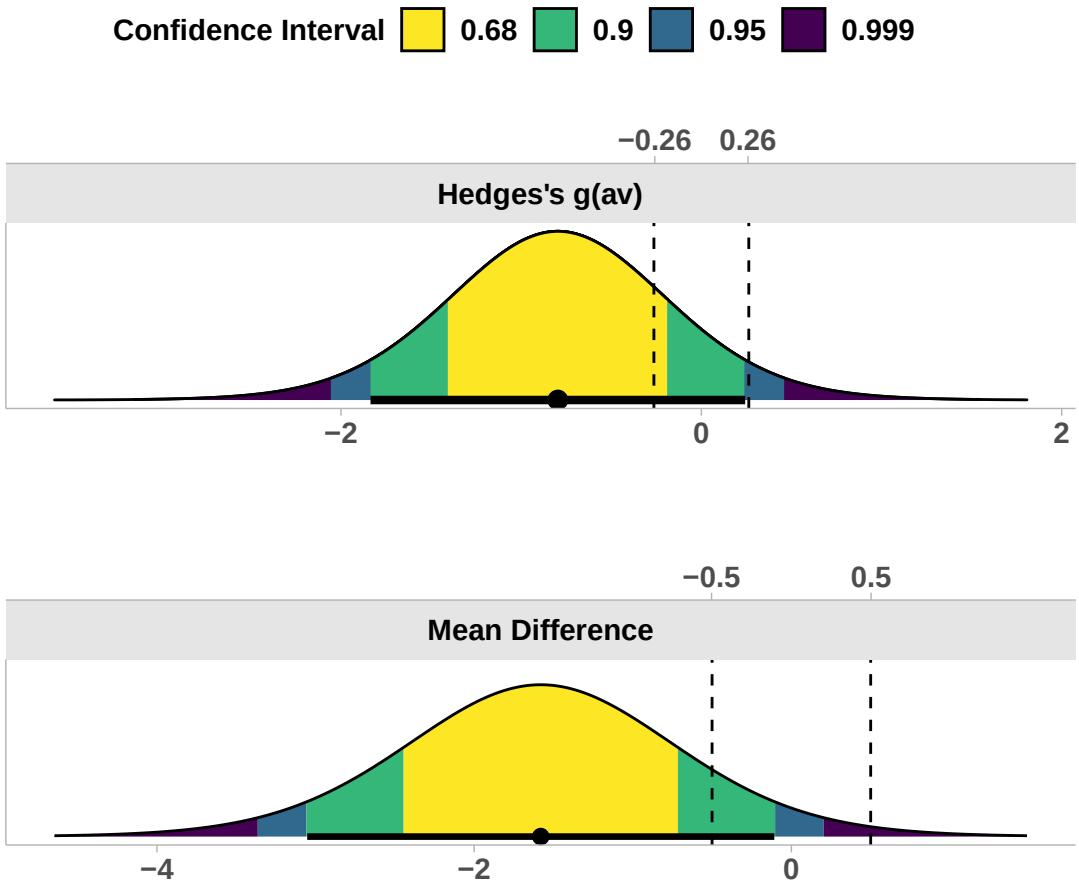

Figure 2. Example of consonance density plot.

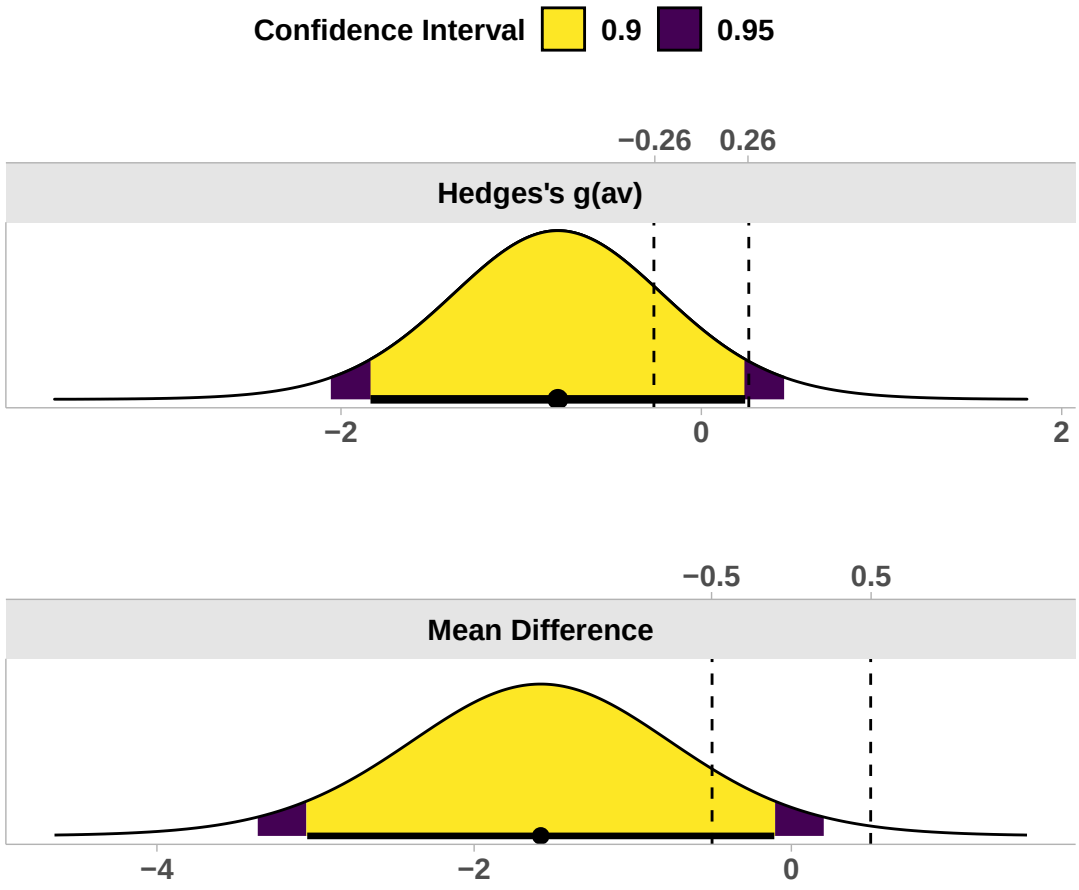

Figure 3. Demonstrating the shading in plot method.

The shading pattern can be modified with the `ci_shades` @ref(fig:shadeplot).

```
plot(res1, type = "cd",
     ci_shades = c(.9,.95))
```

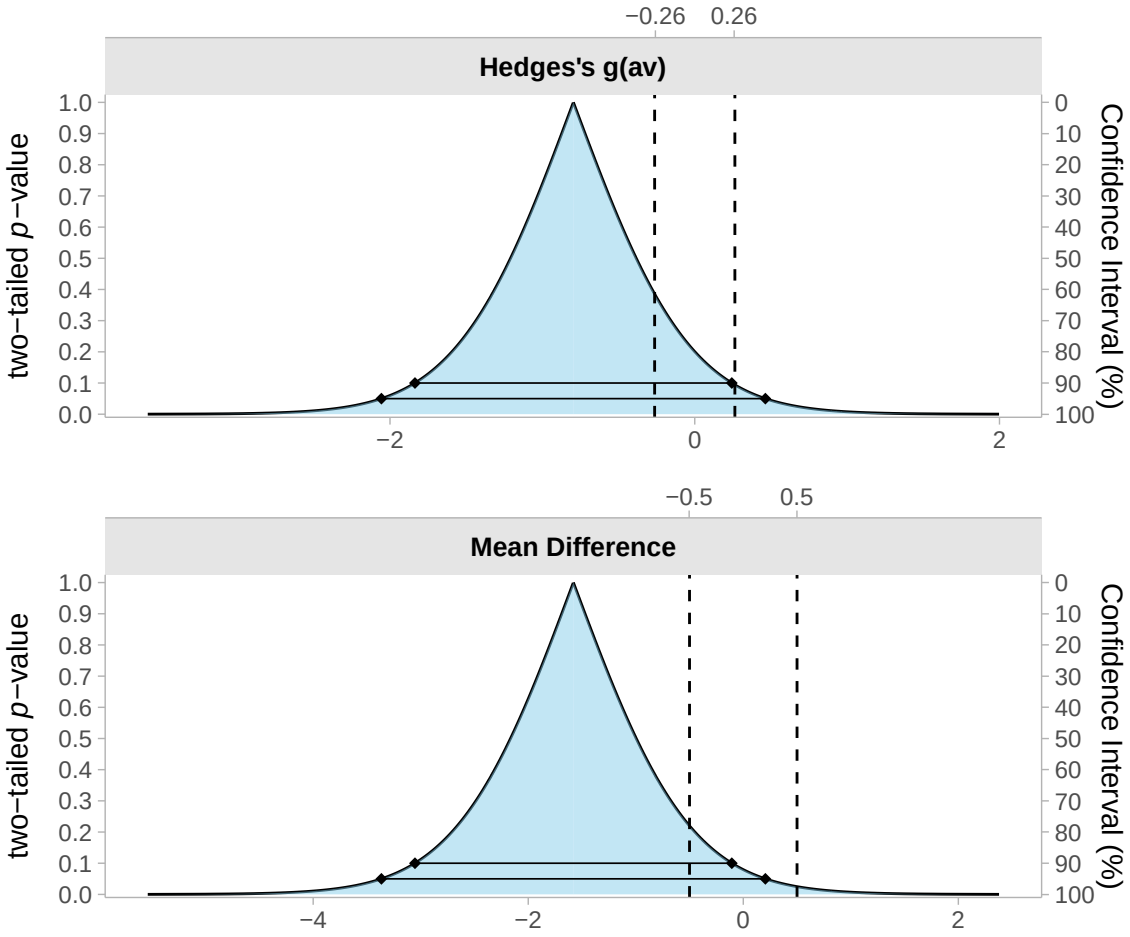

Figure 4. Example of consonance plot.

Consonance plots, where all confidence intervals can be simultaneous plotted, can also be produced. The advantage here is multiple confidence interval lines can plotted at once @ref(fig:conplot).

```
plot(res1, type = "c",
     ci_lines =  c(.9,.95))
```

2.2.2. Paired Samples

To perform a paired samples TOST, the process does not change much. We could process the test the same way by providing a formula. All we would need to then is change `paired` to TRUE.

```
res2 = t_TOST(formula = extra ~ group,
              data = sleep,
              paired = TRUE,
              eqb = .5)
res2
```

```
##
## Paired t-test
##
## The equivalence test was non-significant, t(9) = -2.777, p = 9.89e-01
## The null hypothesis test was significant, t(9) = -4.062, p = 2.83e-03
## NHST: reject null significance hypothesis that the effect is equal to zero
## TOST: don't reject null equivalence hypothesis
##
## TOST Results
##                 t df p.value
## t-test     -4.062  9   0.003
## TOST Lower -2.777  9   0.989
## TOST Upper -5.348  9 < 0.001
##
## Effect Sizes
##               Estimate    SE               C.I. Conf. Level
## Raw             -1.580 0.389   [-2.293, -0.867]         0.9
## Hedges's g(z)   -1.174 0.411 [-1.8046, -0.4977]         0.9
## Note: SMD confidence intervals are an approximation. See vignette("SMD_calcs").
```

However, we may have two vectors of data that are paired. So instead we may want to just provide those separately rather than using a data set and setting the formula. This can be demonstrated with the "bugs" data.

```
res3 = t_TOST(x = bugs$LDHF,
              y = bugs$LDLF,
              paired = TRUE,
              eqb = 1)
res3
```

```
##
## Paired t-test
##
## The equivalence test was non-significant, t(90) = 2.655, p = 9.95e-01
## The null hypothesis test was significant, t(90) = 6.649, p = 2.22e-09
## NHST: reject null significance hypothesis that the effect is equal to zero
## TOST: don't reject null equivalence hypothesis
##
## TOST Results
##                 t df p.value
## t-test      6.649 90 < 0.001
## TOST Lower 10.642 90 < 0.001
## TOST Upper  2.655 90   0.995
##
## Effect Sizes
##               Estimate     SE             C.I. Conf. Level
## Raw             1.6648 0.2504  [1.2487, 2.081]         0.9
## Hedges's g(z)   0.6911 0.1167 [0.4987, 0.8802]         0.9
## Note: SMD confidence intervals are an approximation. See vignette("SMD_calcs").
```

We may want to perform a Minimal Effect Test with the `hypothesis` argument set to "MET".

```
res3a = t_TOST(x = bugs$LDHF,
               y = bugs$LDLF,
               paired = TRUE,
               hypothesis = "MET",
               eqb = 1)
res3a
```

```
##
## Paired t-test
##
## The minimal effect test was significant, t(90) = 10.642, p = 4.69e-03
## The null hypothesis test was significant, t(90) = 6.649, p = 2.22e-09
## NHST: reject null significance hypothesis that the effect is equal to zero
## TOST: reject null MET hypothesis
##
## TOST Results
##                 t df p.value
## t-test      6.649 90 < 0.001
## TOST Lower 10.642 90       1
## TOST Upper  2.655 90   0.005
##
## Effect Sizes
##               Estimate     SE             C.I. Conf. Level
## Raw             1.6648 0.2504  [1.2487, 2.081]         0.9
## Hedges's g(z)   0.6911 0.1167 [0.4987, 0.8802]         0.9
## Note: SMD confidence intervals are an approximation. See vignette("SMD_calcs").
```

2.2.3. One Sample t-test

In other cases we may just have a one sample test. If that is the case all we have to do is supply the **x** argument for the data. For this test we may hypothesis that the mean of LDHF is not more than 1.5 points greater or less than 7.

```
res4 = t_TOST(x = bugs$LDHF,
              hypothesis = "EQU",
              eqb = c(5.5,8.5))
res4
```

```
##
## One Sample t-test
##
## The equivalence test was significant, t(90) = -4.244, p = 2.66e-05
## The null hypothesis test was significant, t(90) = 27.942, p = 3.91e-46
## NHST: reject null significance hypothesis that the effect is equal to zero
## TOST: reject null equivalence hypothesis
##
## TOST Results
##                 t df p.value
## t-test     27.942 90 < 0.001
## TOST Lower  7.116 90 < 0.001
## TOST Upper -4.244 90 < 0.001
##
## Effect Sizes
##            Estimate     SE             C.I. Conf. Level
## Raw           7.379 0.2641  [6.9402, 7.818]         0.9
## Hedges's g    2.905 0.2395 [2.5058, 3.2949]         0.9
## Note: SMD confidence intervals are an approximation. See vignette("SMD_calcs").
```

2.2.4. Using Summary Statistics

In some cases you may only have access to the summary statistics. Therefore, I created a function, `tsum_TOST`, to perform the same tests just based on the summary statistics. This involves providing the function with a number of different arguments.

- `n1 & n2` the sample sizes (only n1 needs to be provided for one sample case)
- `m1 & m2` the sample means
- `sd1 & sd2` the sample standard deviation
- `r12` the correlation between each if paired is set to TRUE

The results from above can be replicated with the `tsum_TOST`:

```
res_tsum = tsum_TOST(
  m1 = mean(bugs$LDHF, na.rm=TRUE), sd1 = sd(bugs$LDHF, na.rm=TRUE),
  n1 = length(na.omit(bugs$LDHF)),
  hypothesis = "EQU", smd_ci = "t", eqb = c(5.5, 8.5)
)

res_tsum
```

```
##
## One-sample t-Test
##
## The equivalence test was significant, t(90) = -4.244, p = 2.66e-05
## The null hypothesis test was significant, t(90) = 27.942, p = 3.91e-46
## NHST: reject null significance hypothesis that the effect is equal to zero
## TOST: reject null equivalence hypothesis
##
## TOST Results
##                 t df p.value
## t-test     27.942 90 < 0.001
## TOST Lower  7.116 90 < 0.001
## TOST Upper -4.244 90 < 0.001
##
## Effect Sizes
##            Estimate     SE             C.I. Conf. Level
## Raw           7.379 0.2641  [6.9402, 7.818]         0.9
## Hedges's g    2.905 0.2395 [2.4289, 3.3804]         0.9
## Note: SMD confidence intervals are an approximation. See vignette("SMD_calcs").
```

3. Robust Methods for Equivalence Testing

Why this may be useful Describe these aren't tests of medians but symmetry tests (myth busting) Paired samples may want to be rank transformed

In this package there are many functions that provide robust alternatives to the `t_TOST` function.

3.1. *Tests of Symmetry (rank based tests)*

The Wilcoxon-Mann-Whitney (WMW) group of tests (includes Mann-Whitney U-test) provide a non-parametric test of differences between groups, or within samples, based on *ranks*. This provides a test of location shift, which is a fancy way of saying differences in the center of the distribution (i.e., in parametric tests the location is mean). With TOST, there are two separate tests of directional location shift to determine if the location shift is within (equivalence) or outside (minimal effect). Many researchers mistakenly think these are tests of medians, but this is not the case (See Divine et al. (2018) for details). The exact calculations can be explored via the documentation of the `wilcox.test` function.

In the TOSTER package, we accomplish this with the `wilcox_TOST` function. This function operates in an extremely similar implementation to the `t_TOST` function. However, the standardized mean difference (SMD) is *not* calculated since this would be an inappropriate measure of effect size alongside the non-parametric test statistics. Instead, a standardized effect size (SES) is calculated for *all* types of comparisons (e.g., two sample, one sample, and paired samples). The function can produce a rank-biserial correlation (CITE), a WMW Odds (CITE), or a concordance probability (CITE) (i.e., non-parametric probability of superiority).[2]

As an example, we can use the sleep data to make a non-parametric comparison of equivalence.

```
data('sleep')
library(TOSTER)

test1 = wilcox_TOST(formula = extra ~ group,
                        data = sleep,
                        paired = FALSE,
                        eqb = .5)


print(test1)
```

```
##
## Wilcoxon rank sum test with continuity correction
##
## The equivalence test was non-significant W = 20.000, p = 8.94e-01
## The null hypothesis test was non-significant W = 25.500, p = 6.93e-02
## NHST: don't reject null significance hypothesis that the effect is equal to zero
## TOST: don't reject null equivalence hypothesis
##
```

[2]There is no plotting capability at this time for the output of this function.

```
## TOST Results
##            Test Statistic p.value
## NHST                 25.5   0.069
## TOST Lower           34.0   0.894
## TOST Upper           20.0   0.013
##
## Effect Sizes
##                            Estimate               C.I. Conf. Level
## Median of Differences        -1.346       [-3.4, -0.1]         0.9
## Rank-Biserial Correlation    -0.490 [-0.7493, -0.1005]         0.9
```

3.2. *Bootstrap TOST*

The bootstrap refers to resampling with replacement and can be used statistical estimation and inference. Bootsrapping techniques are very useful because they are considered somewhat robust to the violations of assumptions for a simple t-test. Therefore we added a bootstrap option, `boot_t_TOST` to the package to provide another robust alternative to the `t_TOST` function.

In this function we provide a percentile bootstrap solution outlined by Efron and Tibshirani (1993) (see chapter 16, page 220). The bootstrapped p-values are derived from the "studentized" version of a test of mean differences (Efron and Tibshirani 1993). Overall, the results should be similar to the results of `t_TOST`. **However**, for paired samples, the Cohen's d(rm) effect size *cannot* be calculated at this time.

3.2.1. Two Sample Algorithm

1. Form B bootstrap data sets from x* and y* wherein x* is sampled with replacement from $\tilde{x}_1, \tilde{x}_2, ...\tilde{x}_n$ and y* is sampled with replacement from $\tilde{y}_1, \tilde{y}_2, ...\tilde{y}_n$
2. t is then evaluated on each sample, but the mean of each sample (y or x) and the overall average (z) are subtracted from each

$$t(z^{*b}) = \frac{(\bar{x}^* - \bar{x} - \bar{z}) - (\bar{y}^* - \bar{y} - \bar{z})}{\sqrt{sd_y^*/n_y + sd_x^*/n_x}}$$

3. An approximate p-value can then be calculated as the number of bootstrapped results greater than the observed t-statistic from the sample.

$$p_{boot} = \frac{\#t(z^{*b}) \geq t_{sample}}{B}$$

The same process is completed for the one sample case but with the one sample solution for the equation outlined by $t(z^{*b})$. The paired sample case in this bootstrap procedure is equivalent to the one sample solution because the test is based on the difference scores.

3.2.2. Example of Bootrapping

Again, we can use the sleep data to see the bootstrapped results. Notice that the plots show how the resampling via bootstrapping indicates the instability of Hedges' d(z).

```
data('sleep')

test1 = boot_t_TOST(formula = extra ~ group,
                    data = sleep,
                    paired = TRUE,
                    eqb = .5,
                    R = 999)


print(test1)
```

```
##
## Bootstrapped Paired t-test
##
## The equivalence test was non-significant, t(9) = -2.777, p = 1e+00
## The null hypothesis test was significant, t(9) = -4.062, p = 0e+00
## NHST: reject null significance hypothesis that the effect is equal to zero
## TOST: don't reject null equivalence hypothesis
##
## TOST Results
##                 t df p.value
## t-test     -4.062  9 < 0.001
## TOST Lower -2.777  9       1
## TOST Upper -5.348  9 < 0.001
##
## Effect Sizes
##               Estimate     SE               C.I. Conf. Level
## Raw             -1.580 0.3611    [-2.181, -1.02]         0.9
## Hedges's g(z)   -1.174 0.6300 [-2.7711, -0.9172]         0.9
## Note: percentile bootstrap method utilized.
```

4. Equivalence Testing with ANOVAs

For an open access tutorial paper explaining how to set equivalence bounds, and how to perform and report equivalence for ANOVA models see Campbell and Lakens (2021). These functions are meant to be omnibus tests, and additional testing may be necessary. For example, comparison of the estimated marginal means, in addition to or as an alternative of with may be prudent.

4.1. *F-test Calculations*

Statistical equivalence testing (or "omnibus non-inferiority testing" by Campbell and Lakens (2021)) for F-tests are special use case of the cumulative distribution function of the non-central F distribution. As Campbell and Lakens (2021) states, these type of

questions answer the question: "Can we reject the hypothesis that the total proportion of variance in outcome Y attributable to X is greater than or equal to the equivalence bound Δ?"

4.1.1. Hypothesis Tests

$$H_0 = 1 > \eta_p^2 \geq \Delta$$

$$H_1 = 0 \geq \eta_p^2 < \Delta$$

In `TOSTER` we go a tad farther and calculate a generalization of the non-centrality parameter that allows the equivalence test for F-tests to be applied to variety of designs.

Campbell and Lakens (2021) calculate the p-value as:

$$p = p_f(F; J-1, N-J, \frac{N \cdot \Delta}{1-\Delta})$$

However, this approach could not be applied to factorial ANOVA and the paper only outlines how to apply this approach to a one-way ANOVA and an extension to Welch's one-way ANOVA.

The non-centrality parameter (ncp = λ) can be calculated with the equivalence bound and the degrees of freedom:

$$\lambda_{eq} = \frac{\Delta}{1-\Delta} \cdot (df_1 + df_2 + 1)$$

The p-value for the equivalence test (p_{eq}) could then be calculated from traditional ANOVA results and the distribution function:

$$p_{eq} = p_f(F; df_1, df_2, \lambda_{eq})$$

4.2. *Example of Equivalence ANOVA Test*

Using the `InsectSprays` data set in R and the base R `aov` function we can demonstrate how this omnibus equivalence testing can be applied with `TOSTER`.

From the initial analysis we an see a clear "significant" effect (the p-value listed is zero but it just very small) of the factor spray. However, we *may* be interested in testing if the effect is practically equivalent. I will arbitrarily set the equivalence bound to a partial eta-squared of 0.35 ($H_0 : \eta_p^2 > 0.35$)

```
library(TOSTER)
# Get Data
data("InsectSprays")
```

```r
# Build ANOVA
aovtest = aov(count ~ spray,
              data = InsectSprays)

# Display overall results
knitr::kable(broom::tidy(aovtest),
             caption = "Traditional ANOVA Test")
```

Table 1.: Traditional ANOVA Test

() term	df	sumsq	meansq	statistic	p.value
()					
spray	5	2668.833	533.76667	34.70228	0
Residuals	66	1015.167	15.38131	NA	NA
()					

We can then use the information in the table above to perform an equivalence test using the `equ_ftest` function. This function returns an object of the S3 class `htest` and the output will look very familiar to the the t-test. The main difference is the estimates, and confidence interval, are for partial η_p^2.

```r
equ_ftest(Fstat = 34.70228,
          df1 = 5,
          df2 = 66,
          eqb = 0.35)
```

```
##
##  Equivalence Test from F-test
##
## data:  Summary Statistics
## F = 34.702, df1 = 5, df2 = 66, p-value = 1
## 95 percent confidence interval:
##  0.5806263 0.7804439
## sample estimates:
## [1] 0.724439
```

Based on the results above we would conclude there is a significant effect of "spray" and the differences due to spray are *not* statistically equivalent. In essence, we reject the traditional null hypothesis of "no effect" but accept the null hypothesis of the equivalence test.

The `equ_ftest` is very useful because all you need is very basic summary statistics. However, if you are doing all your analyses in R then you can use the `equ_anova` function. This function accepts objects produced from `stats::aov`, `car::Anova` and `afex::aov_car` (or any ANOVA from derived from `afex`).

```r
equ_anova(aovtest,
          eqb = 0.35)
```

```
##        effect df1 df2  F.value       p.null      pes eqbound     p.equ
## 1 spray         5  66 34.70228 3.182584e-17 0.724439    0.35 0.9999965
```

5. Conclusions

6. Additional Information

Acknowledgement(s)

I'd would like to thank everyone from the Lakens' laboratory group for their input and suggestions.

Disclosure statement

The author of this manuscript is the author of the `TOSTER` package. Citations of this manuscript will benefit their citation count.

Funding

No funding was provided for this work.

Notes on contributor(s)

Daniel Lakens provided a review of many of the materials that have been incorporated into the update of `TOSTER`, and was the original author of this package.

Nomenclature/Notation

- ANOVA: Analysis of Variance
- MET: Minimal Effects Test
- ncp: non-centrality parameter
- SMD: Standardized mean difference (e.g., Cohen's d)
- TOST: Two-one sided tests
- WMW: Wilcoxon-Mann-Whitney

Notes

The R package is also (partially) implemented in jamovi as the TOSTER module.

References

Altman, Douglas G, and J Martin Bland. 1995. "Statistics notes: Absence of evidence is not evidence of absence." *BMJ* 311 (7003): 485.

Campbell, Harlan, and Daniël Lakens. 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): e12201. `https://bpspsychub.onlinelibrary.wiley.com/doi/abs/10.1111/bmsp.12201`.

Divine, George W, H James Norton, Anna E Barón, and Elizabeth Juarez-Colunga. 2018. "The Wilcoxon–Mann–Whitney procedure fails as a test of medians." *The American Statistician* 72 (3): 278–286. `https://doi.org/10.1080/00031305.2017.1305291`.

Efron, Bradley, and Robert J. Tibshirani. 1993. *An Introduction to the Bootstrap*. Monographs on Statistics and Applied Probability 57. Boca Raton, Florida, USA: Chapman & Hall/CRC.

Hedges, Larry V. 1981. "Distribution theory for Glass's estimator of effect size and related estimators." *journal of Educational Statistics* 6 (2): 107–128.

Lakens, Daniel. 2017. "Equivalence tests: A practical primer for t-tests, correlations, and meta-analyses." *Social Psychological and Personality Science* 1: 1–8.

Schuirmann, Donald J. 1987. "A comparison of the two one-sided tests procedure and the power approach for assessing the equivalence of average bioavailability." *Journal of pharmacokinetics and biopharmaceutics* 15 (6): 657–680.