An Extended Commentary On Post-Publication Peer Review In Organizational Neuroscience

In press at Meta-Psychology

Guy A. Prochilo

April 25, 2019

- This R Markdown document reproduces the analyses found in the above publication and provides instructions for independent reproduction.
- A pre-print of this publication can be found at: https://psyarxiv.com/hv3rm (https://psyarxiv.com/hv3rm)
- Click here (https://osf.io/ayuxz/) for a full history of the submission to *Meta-Psychology* (https://open.lnu.se/index.php/metapsychology/about)

1 Instructions

Follow these instructions to independently reproduce all analyses in the above publication via an R Markdown file.

1. Install requisite software:

- R (v-3.5.1)
- R Studio (v-1.1.463)

Note: These analyses were performed using using Windows 10 and have not been tested on Unix systems.

2. Download the analysis pipeline:

- Download or clone the git repository for this analysis: https://github.com/gprochilo/org_neuro_com (https://github.com/gprochilo/org_neuro_com)
- The files within this repository must be saved to a single directory

3. Open the R Studio project file:

- Open `analysis.Rproj` to begin an R Studio session in your current working directory

4. Install the required packages

- This analysis pipeline requires the following packages:

```
install.packages("ggplot2")
install.packages("scales")
install.packages("ggrepel")
install.packages("ggpubr")
install.packages("psych")
install.packages("cocor")
install.packages("pbapply")
install.packages("reshape2")
install.packages("userfriendlyscience")
install.packages("devtools")
install.packages("SuppDists")
install.packages("here")
```

- The cannonball package can be installed from Github after installing devtools.

```
install_github("janhove/cannonball")
```

5. Knit the R Markdown file to reproduce all analyses:

- Open `markdown.rmd` in R Studio and select **Knit**
- This action will reproduce this document and all associated analyses

2 Prepare R Workspace

The scripts below are evaluated by R Markdown to reproduce all analyses.

1. Load and attach required packages

- R Markdown will load and attach the required packages by sourcing the `libraries.R` file

```
source("libraries.R")
```

2. Source the analysis scripts

- R Markdown will source the analysis script file for this project: `ProchFun_ONC-v2.R`
- This file includes all scripts required for reproducing all analyses

```
source("ProcFun_ONC-v2.R")
```

```
## # MIT License
## #
## # ProcFun_ONC-v2.R: R functions for organizational neuroscience commentary.
## # Copyright (c) 2019 Guy A. Prochilo
## #
## # Permission is hereby granted, free of charge, to any person obtaining a copy
## # of this software and associated documentation files (the "Software"), to deal
## # in the Software without restriction, including without limitation the rights
## # to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
## # copies of the Software, and to permit persons to whom the Software is
## # furnished to do so, subject to the following conditions:
## #
## #   The above copyright notice and this permission notice shall be included in all
## # copies or substantial portions of the Software.
## #
## # THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
## # IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
## # FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
## # AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
## # LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
## # OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
## # SOFTWARE.
## #
## # Contact
## # Name: Guy A. Prochilo
## # Email: guy.prochilo@gmail.com
## #
## # Last update: April 2019
## #
## # Cite as:
## # Prochilo, G. A. (2019). ProcFun_ONC-v2.R: R functions for organizational neurosci
ence commentary.
## # Retrieved from https://github.com/gprochilo
```

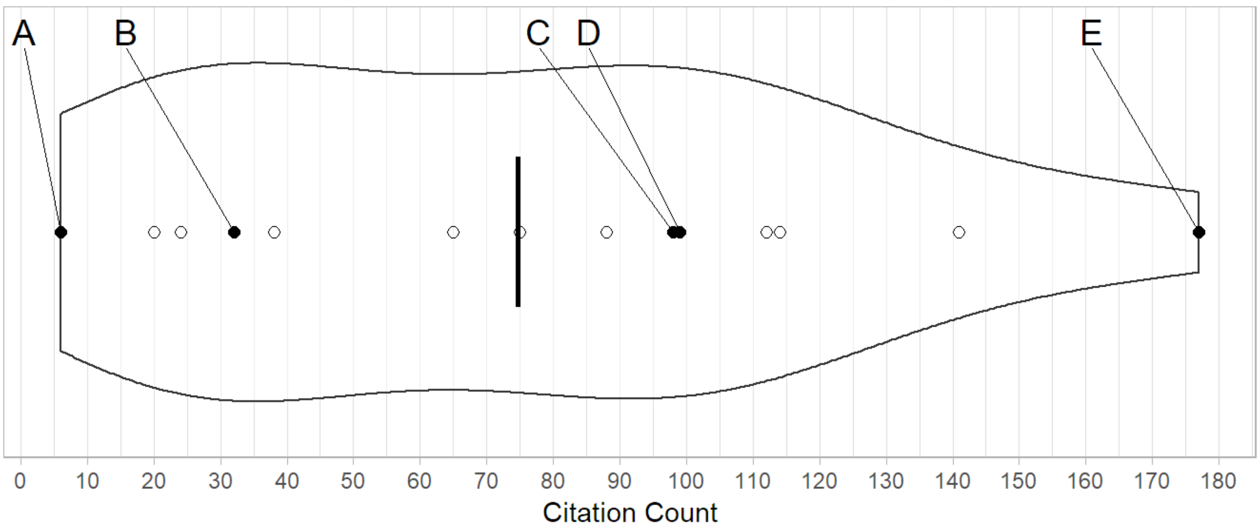
3 Reproduce Figures

3.1 Figure 1

- Reproduce the dot plot presented in Figure 1

```
cite.plot()
```

```
## $mu
## [1] 74.73333
##
## $stdev
## [1] 49.62497
##
## $range
## [1] "6 to 177"
##
## $quants
##   10%   20%   30%   40%   50%   60%   70%   80%   90%
##  21.6  30.4  33.2  54.2  75.0  92.0  98.8 112.4 130.2
```

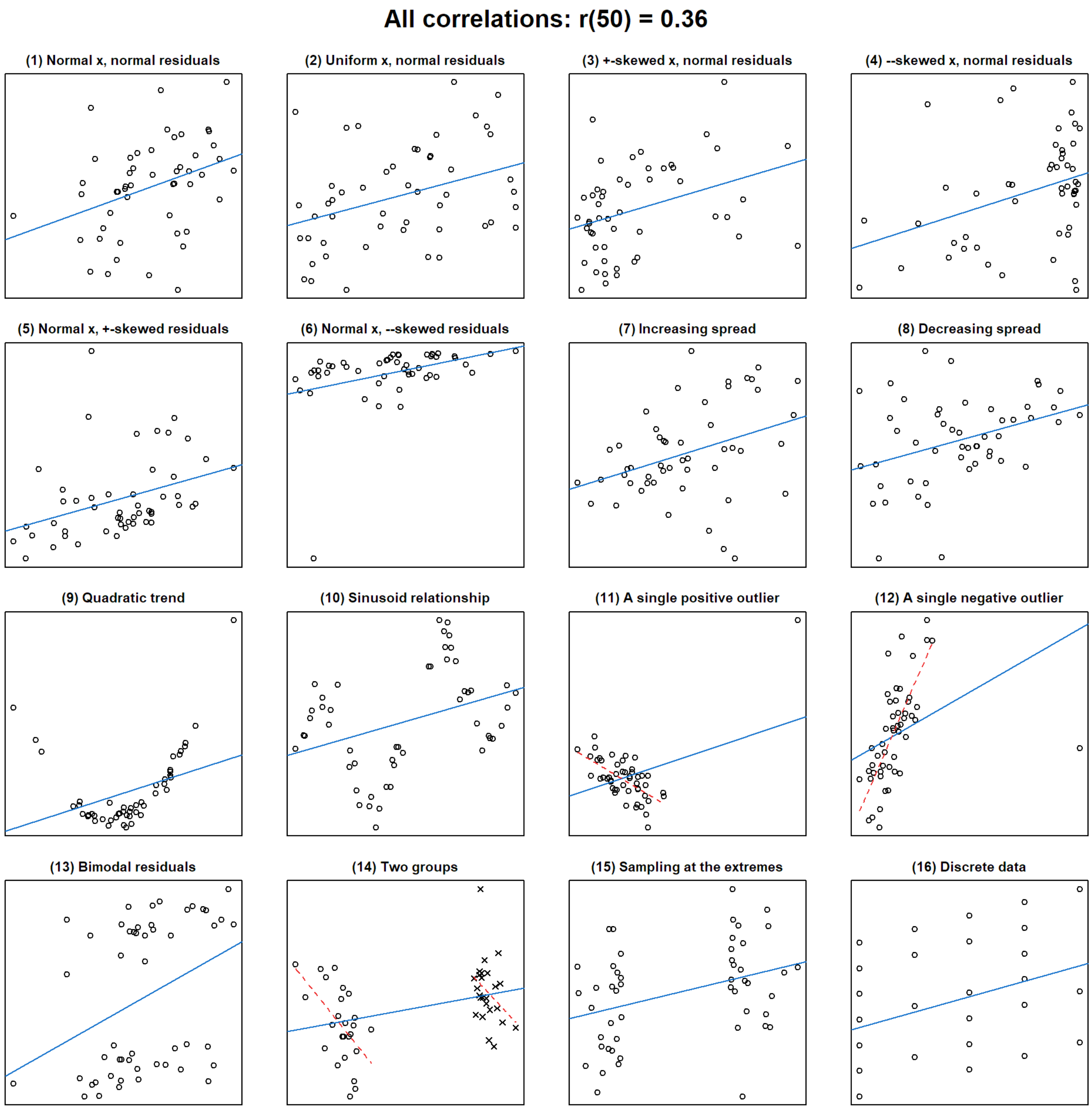



```
## [1] "EPS and PNG files generated."
```

3.2 Figure 2

- Reproduce scatterplots consistent with $r = 0.36$ and $N = 50$
- Note 1: The software outputs *N* rather than *df* in the title (this has been adjusted in the associated publication)
- Note 2: While the software outputs 16 scatterplots, eight scatterplots were included in the final publication due to space constraints

```
plot.r(r = 0.36, n = 50, plot = TRUE)
```

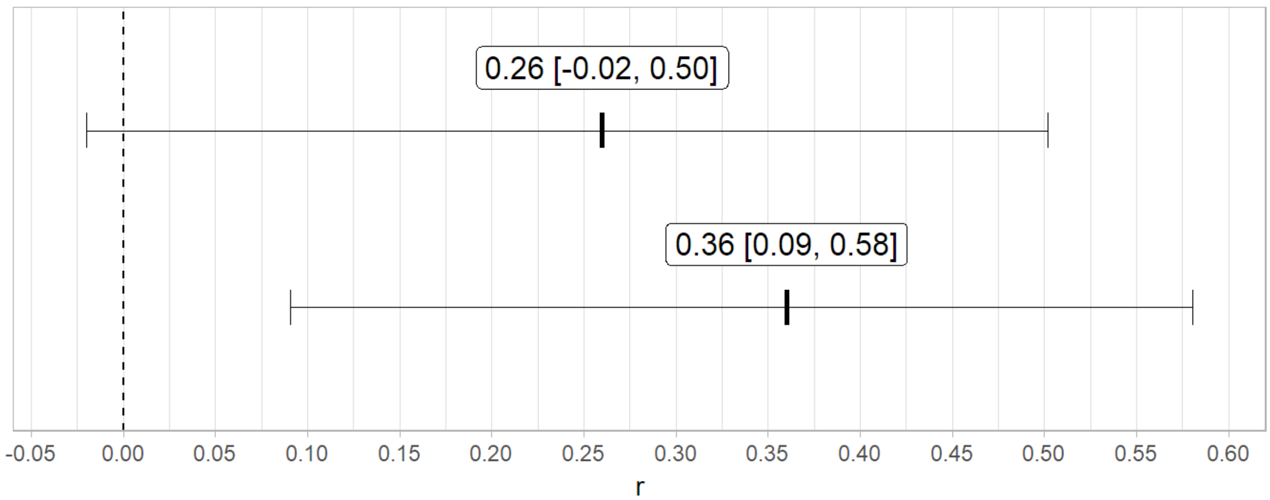


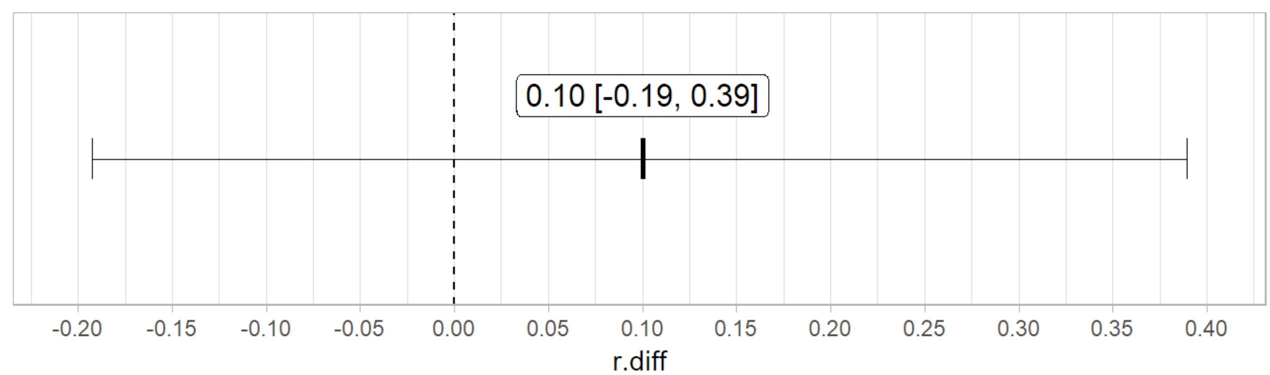

```
## [1] "EPS and PNG files generated."
```

3.3 Figure 3

- Reproduce Figure 3A and Figure 3B
- Figure 3A (upper) is graphical representation of $r = .26$ and $r = .36$ and their 95% CIs with a sample of $N = 50$
- Figure 3B (lower) is the results of Zou's test.

```
zou.test(
  r.jk=.36,
  r.jh=.26,
  r.kh=.39,
  n=50,
  twotailed = TRUE,
  alpha=0.05,
  conf.level=0.95,
  null.value=0,
  plot = TRUE
)
```






```
## [1] "EPS and PNG files generated."
```

```
## $r.jk
## [1] "0.36 [0.09, 0.58]"
##
## $r.jh
## [1] "0.26 [-0.02, 0.50]"
##
## $r.diff
## [1] "0.10 [-0.19, 0.39]"
```

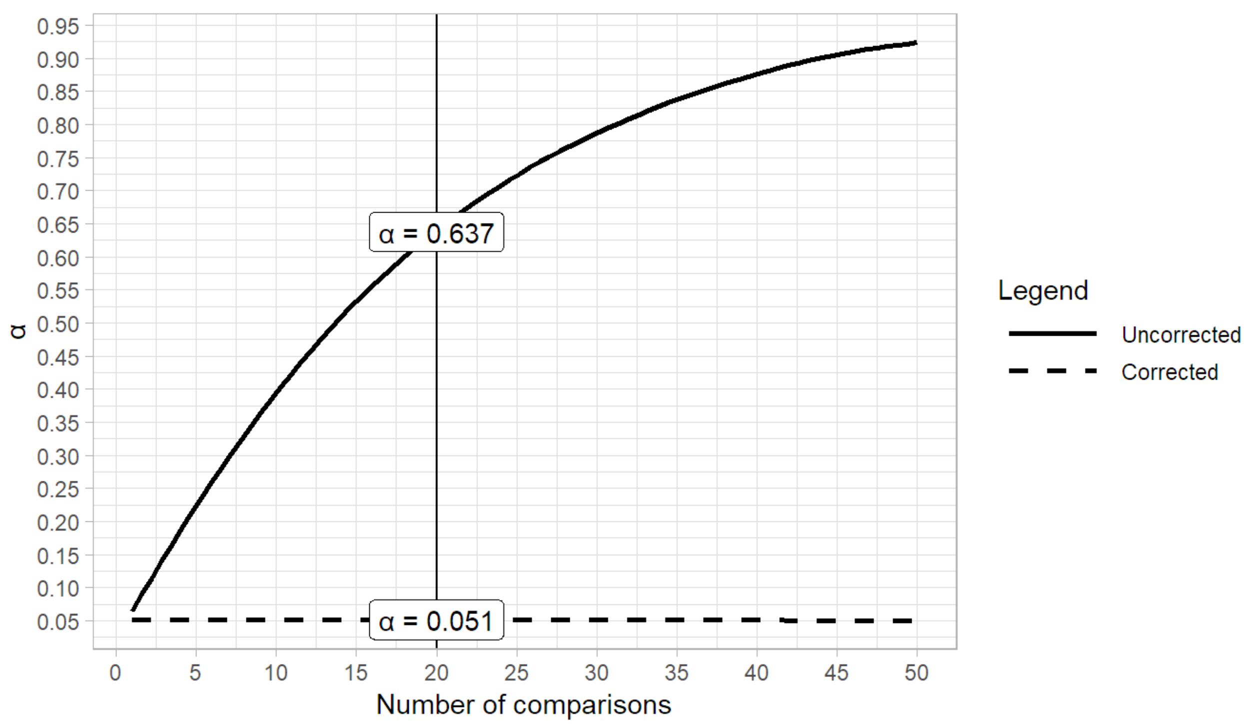
3.4 Figure 4

- Reproduce the simulation presented in Figure 4 and its associated dataset.
- The code in this R Markdown file uses a CSV file that contains the data generated by the simulation instead of running the simulation itself
- To run the full simulation you may run the following scripts outside of the R Markdown environment (runtime approx. 2-3 min):

```
source("libraries.R")
source("ProchFun_ONC-v2.R")
errorPlot(comparisons = 50,sampleSize = 17,reps = 10000,plotname = "errorPlot", n.comp
arison.label = 20)
```

- The following code generates Figure 4 using a CSV file of data saved from the above simulation:

```
errorPlot.lazy("figure_4_data.csv", n.comparison.label = 20)
```




```
## [1] "EPS and PNG files generated."
```

```
##     comparisons      status  alpha
## 1            1 uncorrected 0.0494
## 2            2 uncorrected 0.0990
## 3            3 uncorrected 0.1445
## 4            4 uncorrected 0.1884
## 5            5 uncorrected 0.2276
## 6            6 uncorrected 0.2691
## 7            7 uncorrected 0.3057
## 8            8 uncorrected 0.3366
## 9            9 uncorrected 0.3673
## 10          10 uncorrected 0.3978
## 11          11 uncorrected 0.4261
## 12          12 uncorrected 0.4545
## 13          13 uncorrected 0.4848
## 14          14 uncorrected 0.5106
## 15          15 uncorrected 0.5360
## 16          16 uncorrected 0.5586
## 17          17 uncorrected 0.5793
## 18          18 uncorrected 0.6034
## 19          19 uncorrected 0.6221
## 20          20 uncorrected 0.6373
## 21          21 uncorrected 0.6562
## 22          22 uncorrected 0.6743
## 23          23 uncorrected 0.6901
## 24          24 uncorrected 0.7038
## 25          25 uncorrected 0.7199
## 26          26 uncorrected 0.7366
## 27          27 uncorrected 0.7505
## 28          28 uncorrected 0.7631
## 29          29 uncorrected 0.7768
## 30          30 uncorrected 0.7899
## 31          31 uncorrected 0.7995
## 32          32 uncorrected 0.8095
## 33          33 uncorrected 0.8182
## 34          34 uncorrected 0.8271
## 35          35 uncorrected 0.8347
## 36          36 uncorrected 0.8431
## 37          37 uncorrected 0.8519
## 38          38 uncorrected 0.8593
## 39          39 uncorrected 0.8672
## 40          40 uncorrected 0.8740
## 41          41 uncorrected 0.8798
## 42          42 uncorrected 0.8858
## 43          43 uncorrected 0.8921
## 44          44 uncorrected 0.8984
## 45          45 uncorrected 0.9042
## 46          46 uncorrected 0.9093
## 47          47 uncorrected 0.9137
## 48          48 uncorrected 0.9178
## 49          49 uncorrected 0.9223
## 50          50 uncorrected 0.9270
## 51           1   corrected 0.0494
## 52           2   corrected 0.0533
```

```
## 53           3   corrected 0.0522
## 54           4   corrected 0.0541
## 55           5   corrected 0.0515
## 56           6   corrected 0.0513
## 57           7   corrected 0.0524
## 58           8   corrected 0.0510
## 59           9   corrected 0.0529
## 60          10   corrected 0.0512
## 61          11   corrected 0.0518
## 62          12   corrected 0.0511
## 63          13   corrected 0.0507
## 64          14   corrected 0.0510
## 65          15   corrected 0.0519
## 66          16   corrected 0.0517
## 67          17   corrected 0.0530
## 68          18   corrected 0.0524
## 69          19   corrected 0.0523
## 70          20   corrected 0.0514
## 71          21   corrected 0.0516
## 72          22   corrected 0.0504
## 73          23   corrected 0.0498
## 74          24   corrected 0.0497
## 75          25   corrected 0.0503
## 76          26   corrected 0.0520
## 77          27   corrected 0.0512
## 78          28   corrected 0.0513
## 79          29   corrected 0.0519
## 80          30   corrected 0.0520
## 81          31   corrected 0.0522
## 82          32   corrected 0.0523
## 83          33   corrected 0.0517
## 84          34   corrected 0.0512
## 85          35   corrected 0.0509
## 86          36   corrected 0.0517
## 87          37   corrected 0.0509
## 88          38   corrected 0.0511
## 89          39   corrected 0.0514
## 90          40   corrected 0.0509
## 91          41   corrected 0.0508
## 92          42   corrected 0.0494
## 93          43   corrected 0.0491
## 94          44   corrected 0.0488
## 95          45   corrected 0.0489
## 96          46   corrected 0.0494
## 97          47   corrected 0.0505
## 98          48   corrected 0.0503
## 99          49   corrected 0.0505
## 100         50   corrected 0.0509
```

4 Waldman et al. (2011a)

4.1 Confidence interval computations

4.1.1 *r*(48) = 0.36, *p* < .05

```
r.confidence(0.36,50,twotailed = TRUE)
```

```
## $result
## [1] "r(48) = 0.36 [0.09, 0.58], p = 0.010"
```

```
## $r
## [1] 0.36
##
## $n
## [1] 50
##
## $df
## [1] 48
##
## $ci.lower
## [1] 0.09074545
##
## $ci.upper
## [1] 0.5802079
##
## $ci.width
## [1] 0.4894624
##
## $MarginOfError
## [1] 0.2447312
##
## $t.val
## [1] 2.673398
##
## $p.val
## [1] 0.01023068
```

4.1.2 *r*(48) = 0.26, *p* < .10

```
r.confidence(0.26,50,twotailed = TRUE)
```

```
## $result
## [1] "r(48) = 0.26 [-0.02, 0.50], p = 0.068"
```

```
## $r
## [1] 0.26
##
## $n
## [1] 50
##
## $df
## [1] 48
##
## $ci.lower
## [1] -0.01977914
##
## $ci.upper
## [1] 0.5020166
##
## $ci.width
## [1] 0.5217957
##
## $MarginOfError
## [1] 0.2608979
##
## $t.val
## [1] 1.86549
##
## $p.val
## [1] 0.06823097
```

4.2 Zou's test

```
zou.test(
  r.jk=.36,
  r.jh=.26,
  r.kh=.39,
  n=50,
  twotailed = TRUE,
  alpha=0.05,
  conf.level=0.95,
  null.value=0,
  plot = FALSE,
  save.plot = FALSE
)
```

```
## $r.jk
## [1] "0.36 [0.09, 0.58]"
##
## $r.jh
## [1] "0.26 [-0.02, 0.50]"
##
## $r.diff
## [1] "0.10 [-0.19, 0.39]"
```

4.3 Precision for planning

4.3.1 $r = 0.36$ and margin of error = 0.18 with 95% confidence

```
pwr.confIntR(r = 0.36, w = 0.18, conf.level = 0.95)
```

```
##          w = 0.18
## r = 0.36       92
## attr(,"conf.level")
## [1] 0.95
```

4.3.2 $r = 0.09$ and margin of error = 0.045 with 95% confidence

```
pwr.confIntR(r = 0.09, w = 0.045, conf.level = 0.95)
```

```
##          w = 0.045
## r = 0.09      1867
## attr(,"conf.level")
## [1] 0.95
```

5 Waldman et al. (2013a)

5.1 Confidence interval computations

5.1.1 $r(24) = 0.32$, $p < .05$ (Two-tailed test)

```
r.confidence(0.32,26,twotailed = TRUE)
```

```
## $result
## [1] "r(24) = 0.32 [-0.08, 0.63], p = 0.111"
```

```
## $r
## [1] 0.32
##
## $n
## [1] 26
##
## $df
## [1] 24
##
## $ci.lower
## [1] -0.07688162
##
## $ci.upper
## [1] 0.6293432
##
## $ci.width
## [1] 0.7062248
##
## $MarginOfError
## [1] 0.3531124
##
## $t.val
## [1] 1.654681
##
## $p.val
## [1] 0.1110101
```

5.1.2 $r(24) = 0.32$, $p < .05$ (One-tailed test)

```
r.confidence(0.32,26,twotailed = FALSE)
```

```
## $result
## [1] "r(24) = 0.32 [-0.01, 0.59], p = 0.056"
```

```
## $r
## [1] 0.32
##
## $n
## [1] 26
##
## $df
## [1] 24
##
## $ci.lower
## [1] -0.0113281
##
## $ci.upper
## [1] 0.5880125
##
## $ci.width
## [1] 0.5993406
##
## $MarginOfError
## [1] 0.2996703
##
## $t.val
## [1] 1.654681
##
## $p.val
## [1] 0.05550505
```

6 Kim and James (2015)

6.1 Confidence interval computations

6.1.1 $r(48) = 0.49$, $p < .05$

```
r.confidence(0.49,17,twotailed = TRUE)
```

```
## $result
## [1] "r(15) = 0.49 [0.01, 0.79], p = 0.046"
```

```
## $r
## [1] 0.49
##
## $n
## [1] 17
##
## $df
## [1] 15
##
## $ci.lower
## [1] 0.01223732
##
## $ci.upper
## [1] 0.785619
##
## $ci.width
## [1] 0.7733817
##
## $MarginOfError
## [1] 0.3866908
##
## $t.val
## [1] 2.177025
##
## $p.val
## [1] 0.04585992
```

6.2 Critical *r* computations

6.2.1 Critical *r* value for a study with *N* = 17

```
critical.r(n = 17, alpha = .05, twotailed = TRUE)
```

```
## $t.crit
## [1] 2.13145
##
## $r.crit
## [1] 0.482146
```