

Supplementary Materials

[Code ▼](#)

In this document, we present our additional analyses. All computations were done using the statistical programming language R¹; regression models were built using the lme4 package²; the conditional inference tree was built with the party package³.

Attempt to replicate significant effects fails

The original paper⁴ reports the following statistically significant findings:

1. An effect of SEX OF PARTICIPANT indicating that males are more likely to share. On p. 2, the authors state, “we find that males are more likely to share their science than females ($\beta = 0.88$, $SE = 0.5$, $F_{1,384} = 8.16$, $P = 0.005$; Fig. 1)”.
2. An effect of the three-way interaction between SEX OF PARTICIPANT, SEX OF REQUESTER, and CONDITION (paper request vs. data request), which—in the authors’ interpretation—“shows that the difference between males and females becomes particularly evident when those males got a request from a male requester.” (p. 2)
3. An effect of the two-way interaction between SEX OF PARTICIPANT and H-INDEX, which is described as follows: “the interaction effect between the sex of the participants and their h-index ($F_{2,384} = 3.27$, $P = 0.049$), shows that for males the likelihood of responding positively to a request for a paper or data decreases with increasing h-index.” (p. 2)

Before we built a regression model, we applied two transformations, shown in Listing 1: (1) we renamed the columns and some of the factor levels to increase the readability of the model summaries that follow, and (2) we log-transformed and centered the numerical predictors H-INDEX and H-INDEX DIFFERENCE to avoid violating model assumptions.

[Hide](#)

```
# Reading data
data <- fread("../data/RawDataMassenEtAl2017.csv")
# Simplifying variable names
names(data) <- c("PARTICIPANT", "SEX_OF_PARTICIPANT", "HINDEX",
                 "SEX_OF_REQUESTER", "STATUS_REQUESTER", "RESPONSE",
                 "CONDITION", "HINDEX_DIFF")
# Factorizing categorical variables
data[, "]:= (PARTICIPANT      = factor(PARTICIPANT),
             SEX_OF_PARTICIPANT = factor(SEX_OF_PARTICIPANT),
             SEX_OF_REQUESTER  = factor(SEX_OF_REQUESTER),
             STATUS_REQUESTER  = factor(STATUS_REQUESTER),
             RESPONSE          = factor(RESPONSE),
             CONDITION          = factor(CONDITION))
# Using meaningful names for levels of categorical variables
levels(data$STATUS_REQUESTER) <- c("student", "postdoc")
levels(data$RESPONSE)         <- c("no", "yes")
levels(data$CONDITION)        <- c("paper", "data")
# Log-transforming numerical predictors
data[, "]:= (sHINDEX          = log(HINDEX + 1),
             sHINDEX_DIFF     = log(data$HINDEX_DIFF + 9))
# Centering numerical predictors
data[, "]:= (sHINDEX          = sHINDEX - mean(sHINDEX),
             sHINDEX_DIFF     = sHINDEX_DIFF - mean(sHINDEX_DIFF))
```

Listing 1: Renaming data and transforming numeric predictors in preparation for regression modeling.

After setting up the data, the significant effects reported in the original paper were translated into a regression formula (Listing 2). Note that we did not—unlike the authors—exclude lower-order terms.

Hide

```
formula <- RESPONSE ~
  # 1: Sex of Participant
  SEX_OF_PARTICIPANT +

  # 2: Interaction between Sex of Participant, Sex of Requester, and
  # Condition
  SEX_OF_PARTICIPANT:SEX_OF_REQUESTER:CONDITION +
  # Lower-order terms
  SEX_OF_PARTICIPANT:SEX_OF_REQUESTER +
  SEX_OF_PARTICIPANT:CONDITION +
  SEX_OF_REQUESTER:CONDITION +
  SEX_OF_REQUESTER +
  CONDITION +

  # 3: Two-way interaction between Sex of Participant and H-Index
  SEX_OF_PARTICIPANT:sHINDEX +
  # Lower-order term
  sHINDEX +

  # By-participant random intercepts
  (1|PARTICIPANT)
```

Listing 2: Regression formula that contains the significant effects reported in the original paper alongside lower-order terms. The numbers in the comments (starting with #) connect the parts to the list of significant effects summarized above.

The formula in Listing 2 was then used to build a mixed-effects logistic regression model. To achieve convergence, we used the `bobyqa` optimizer and increased the number of iterations (Listing 3).

Hide

```
m <- glmer(formula,
  data = data,
  family = binomial(link = "logit"),
  control = glmerControl(
    optimizer = 'bobyqa',
    optCtrl = list(maxfun = 1e6)
  )
)
```

Listing 3: Parameters of the initial regression model.

As can be seen in the model output below (Listing 4), none of the effects described by the authors turn out to be statistically significant. If we then proceed in the fashion outlined by the authors (i.e., backwards model selection) and reduce the model based on AIC scores (using `drop1`, see Listings 5–8), we arrive at model `m4` (Listing 9)

Hide

```
summary(m, correlation = FALSE)
```

```

Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) ['glmerMod']
Family: binomial ( logit )
Formula: RESPONSE ~ SEX_OF_PARTICIPANT + SEX_OF_PARTICIPANT:SEX_OF_REQUESTER:CONDITION +
  SEX_OF_PARTICIPANT:SEX_OF_REQUESTER + SEX_OF_PARTICIPANT:CONDITION +
  SEX_OF_REQUESTER:CONDITION + SEX_OF_REQUESTER + CONDITION +
  SEX_OF_PARTICIPANT:sHINDEX + sHINDEX + (1 | PARTICIPANT)
Data: data
Control: glmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 1e+06))

      AIC      BIC   logLik deviance df.resid
451.5    495.2   -214.8    429.5     383

Scaled residuals:
      Min       1Q   Median       3Q      Max
-4.0563 -0.7698  0.4429  0.5493  1.0750

Random effects:
Groups      Name      Variance Std.Dev.
PARTICIPANT (Intercept) 0.6855   0.8279
Number of obs: 394, groups: PARTICIPANT, 291

Fixed effects:
                                Estimate Std. Error z value Pr(>|z|)
(Intercept)                   1.16437    0.33529   3.473 0.000515 ***
SEX_OF_PARTICIPANTm            0.54438    0.45997   1.184 0.236605
SEX_OF_REQUESTERM            -0.15658    0.42370  -0.370 0.711714
CONDITIONdata                 -0.79151    0.56895  -1.391 0.164172
sHINDEX                      -0.02097    0.21898  -0.096 0.923706
SEX_OF_PARTICIPANTm:SEX_OF_REQUESTERM  0.94997    0.66534   1.428 0.153348
SEX_OF_PARTICIPANTm:CONDITIONdata -0.67276    0.78664  -0.855 0.392419
SEX_OF_REQUESTERM:CONDITIONdata -0.08658    0.80088  -0.108 0.913909
SEX_OF_PARTICIPANTm:sHINDEX      -0.53254    0.31349  -1.699 0.089367 .
SEX_OF_PARTICIPANTm:SEX_OF_REQUESTERM:CONDITIONdata 0.19394    1.17604   0.165 0.869013
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Listing 4: The summary of the regression model with lower order terms leaves all effects non-significant. The only significant term is the intercept, which indicates that people are in general more likely to share than not.

Hide

drop1(m)

Single term deletions

Model:

```

RESPONSE ~ SEX_OF_PARTICIPANT + SEX_OF_PARTICIPANT:SEX_OF_REQUESTER:CONDITION +
  SEX_OF_PARTICIPANT:SEX_OF_REQUESTER + SEX_OF_PARTICIPANT:CONDITION +
  SEX_OF_REQUESTER:CONDITION + SEX_OF_REQUESTER + CONDITION +
  SEX_OF_PARTICIPANT:sHINDEX + sHINDEX + (1 | PARTICIPANT)

```

	Df	AIC
<none>		451.51
SEX_OF_PARTICIPANT:sHINDEX	1	452.53
SEX_OF_PARTICIPANT:SEX_OF_REQUESTER:CONDITION	1	449.53

Listing 5: In the first iteration, the lowest AIC value is associated with dropping the interaction SEX_OF_PARTICIPANT:SEX_OF_REQUESTER:CONDITION.

Hide

```
m2 <- update(m, .~. - SEX_OF_PARTICIPANT:SEX_OF_REQUESTER:CONDITION)
drop1(m2)
```

Single term deletions

Model:

```
RESPONSE ~ SEX_OF_PARTICIPANT + SEX_OF_REQUESTER + CONDITION +
  sHINDEX + (1 | PARTICIPANT) + SEX_OF_PARTICIPANT:SEX_OF_REQUESTER +
  SEX_OF_PARTICIPANT:CONDITION + SEX_OF_REQUESTER:CONDITION +
  SEX_OF_PARTICIPANT:sHINDEX
```

	Df	AIC
<none>		449.53
SEX_OF_PARTICIPANT:SEX_OF_REQUESTER	1	451.18
SEX_OF_PARTICIPANT:CONDITION	1	448.62
SEX_OF_REQUESTER:CONDITION	1	447.53
SEX_OF_PARTICIPANT:sHINDEX	1	450.57

Listing 6: In the second iteration, the lowest AIC value is associated with dropping the interaction SEX_OF_REQUESTER:CONDITION.

Hide

```
m3 <- update(m2, .~. - SEX_OF_REQUESTER:CONDITION)
drop1(m3)
```

Single term deletions

Model:

```
RESPONSE ~ SEX_OF_PARTICIPANT + SEX_OF_REQUESTER + CONDITION +
  sHINDEX + (1 | PARTICIPANT) + SEX_OF_PARTICIPANT:SEX_OF_REQUESTER +
  SEX_OF_PARTICIPANT:CONDITION + SEX_OF_PARTICIPANT:sHINDEX
```

	Df	AIC
<none>		447.53
SEX_OF_PARTICIPANT:SEX_OF_REQUESTER	1	449.21
SEX_OF_PARTICIPANT:CONDITION	1	446.64
SEX_OF_PARTICIPANT:sHINDEX	1	448.58

Listing 7: In the third iteration, the lowest AIC value is associated with dropping the interaction SEX_OF_PARTICIPANT:CONDITION.

Hide

```
m4 <- update(m3, .~. - SEX_OF_PARTICIPANT:CONDITION)
drop1(m4)
```

Single term deletions

Model:

```
RESPONSE ~ SEX_OF_PARTICIPANT + SEX_OF_REQUESTER + CONDITION +  
  sHINDEX + (1 | PARTICIPANT) + SEX_OF_PARTICIPANT:SEX_OF_REQUESTER +  
  SEX_OF_PARTICIPANT:sHINDEX
```

	Df	AIC
<none>		446.64
CONDITION	1	460.47
SEX_OF_PARTICIPANT:SEX_OF_REQUESTER	1	448.34
SEX_OF_PARTICIPANT:sHINDEX	1	447.37

Listing 8: In the fourth iteration, the lowest AIC value is associated with leaving the model as is.

Hide

```
summary(m4, correlation = FALSE)
```

Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) [`'glmerMod'`]

Family: binomial (logit)

Formula: RESPONSE ~ SEX_OF_PARTICIPANT + SEX_OF_REQUESTER + CONDITION +
 sHINDEX + (1 | PARTICIPANT) + SEX_OF_PARTICIPANT:SEX_OF_REQUESTER +
 SEX_OF_PARTICIPANT:sHINDEX

Data: data

Control: glmerControl(optimizer = "bobyqa", optCtrl = list(maxfun = 1e+06))

AIC	BIC	logLik	deviance	df.resid
446.6	478.4	-215.3	430.6	386

Scaled residuals:

Min	1Q	Median	3Q	Max
-3.7883	-0.7707	0.4400	0.5368	1.0959

Random effects:

Groups	Name	Variance	Std.Dev.
PARTICIPANT	(Intercept)	0.6756	0.822

Number of obs: 394, groups: PARTICIPANT, 291

Fixed effects:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	1.24889	0.31554	3.958	7.56e-05 ***
SEX_OF_PARTICIPANTm	0.31873	0.37617	0.847	0.396822
SEX_OF_REQUESTERm	-0.17807	0.35816	-0.497	0.619054
CONDITIONdata	-1.11122	0.30199	-3.680	0.000234 ***
sHINDEX	-0.02642	0.21986	-0.120	0.904357
SEX_OF_PARTICIPANTm:SEX_OF_REQUESTERm	1.01318	0.54180	1.870	0.061480 .
SEX_OF_PARTICIPANTm:sHINDEX	-0.50056	0.30918	-1.619	0.105450

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Listing 9: The summary of the final model `m4` contains only one significant effect: Condition.

The only term that is statistically significant in model `m4` (Listing 9) is `CONDITION`. The interaction between `SEX OF PARTICIPANT` and `SEX OF REQUESTER` fails to meet the customary $p < 0.05$ criterion.

The same results appear if we use non-parametric conditional inference tree modeling (Figure 1).

```
ct <- ctree(formula = RESPONSE ~
  CONDITION +
  SEX_OF_REQUESTER +
  SEX_OF_PARTICIPANT +
  STATUS_REQUESTER +
  SHINDEX +
  HINDEX_DIFF,
  data = data)
plot(ct)
```

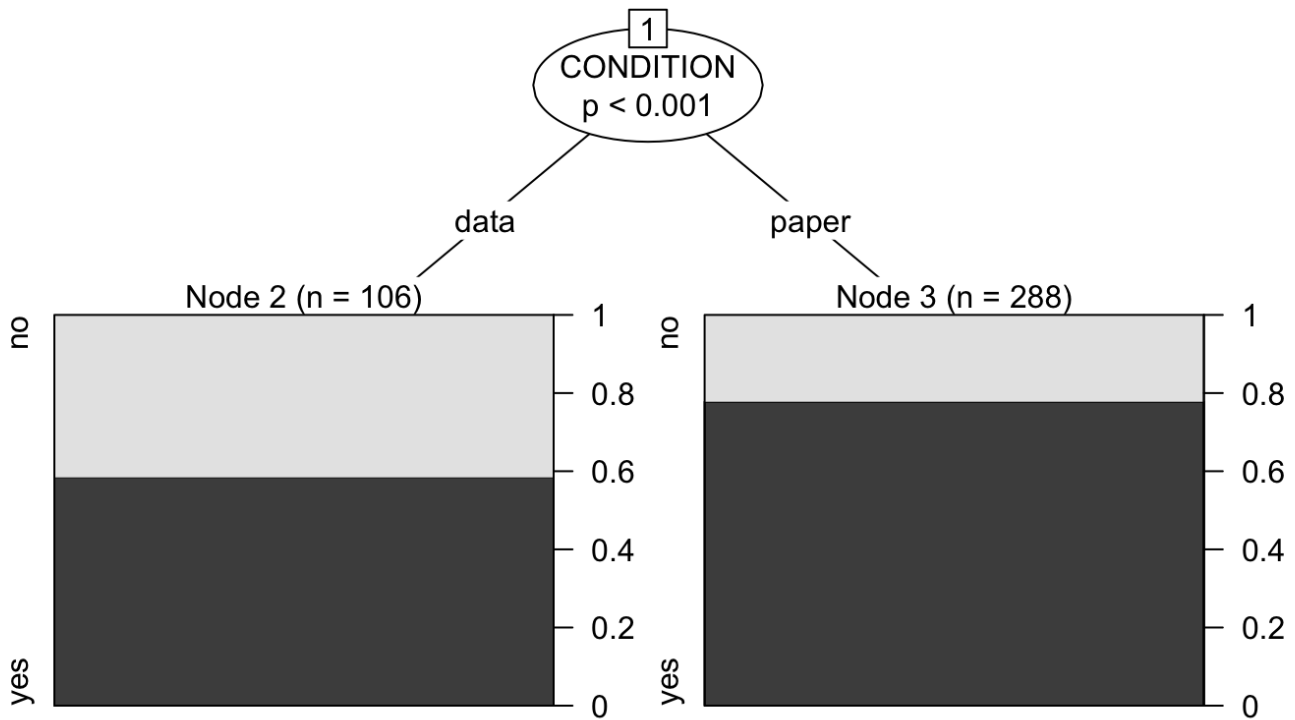


Figure 1: The conditional inference tree shows one significant effect: *CONDITION*.

The true false positive rate in the original paper is close to 30 %

Our simulation replicates the setup of the original study and performs the same backward modeling procedure; it assumes that the response rate is entirely random. In what follows we show that under these circumstances, the crucial interaction between SEX OF PARTICIPANT and SEX OF REQUESTER is retained in 30 % of all cases.

Before we start the simulation, we set up a cluster for efficient computation (Listing 10). We used the same model formula as in the regression model above but removed the by-participant random intercepts because we do not have any repeated measures.

```
n <- 394 # number of observations in original dataset
formula <- update.formula(formula, .~. - (1|PARTICIPANT))
# Setting up cluster
cluster <- makeCluster(detectCores() - 1, outfile = "log.txt")
clusterEvalQ(cluster, {
  library(data.table)
  library(doParallel)
  library(MASS)
})
```

```
[[1]]
[1] "MASS"      "doParallel" "parallel"    "iterators"  "foreach"    "data.table" "stats"
[8] "graphics"  "grDevices"  "utils"      "datasets"   "methods"    "base"

[[2]]
[1] "MASS"      "doParallel" "parallel"    "iterators"  "foreach"    "data.table" "stats"
[8] "graphics"  "grDevices"  "utils"      "datasets"   "methods"    "base"

[[3]]
[1] "MASS"      "doParallel" "parallel"    "iterators"  "foreach"    "data.table" "stats"
[8] "graphics"  "grDevices"  "utils"      "datasets"   "methods"    "base"
```

Hide

```
clusterExport(cluster, c("data", "n", "formula"),
  envir = environment())
```

Listing 10: Setting up a cluster for efficient computation.

Next, we set up a function that simulates the original study (Listing 11). Categorical predictors (SEX OF PARTICIPANT, SEX OF REQUESTER, etc.) are sampled randomly, and the numerical predictor H-INDEX is sampled from the distribution in the original dataset. The function then performs AIC-based backwards model selection and returns the coefficients of the resulting model.

Hide

```

simulate_study <- function(x) {
  random_data <- data.table(
    SEX_OF_PARTICIPANT = factor(sample(c("f", "m"), n, replace = TRUE)),
    SHINDEX            = sample(data$SHINDEX),
    SEX_OF_REQUESTER   = factor(sample(c("f", "m"), n, replace = TRUE)),
    STATUS_REQUESTER   = factor(sample(c("postdoc", "student"), n,
                                      replace = TRUE)),
    RESPONSE           = factor(sample(c("no", "yes"), n,
                                      replace = TRUE)),
    CONDITION          = factor(sample(c("data", "paper"), n,
                                      replace = TRUE))
  )

  m <- stepAIC(glm(formula,
                  family = "binomial",
                  data = random_data,
                  direction = "backward",
                  trace = FALSE)

  summary(m)$coefficients
}

```

Listing 11: Function that simulates a dataset and a model selection procedure similar to the original study.

This function is now called 10,000 times, and its results are recorded (Listing 12).

Hide

```

# Simulating procedure 10,000 times
set.seed(12345)
model_coefficients <- parLapply(cl = cluster, 1:10000, simulate_study)

```

Listing 12: Calling the simulation 10,000 times on the cluster. Note that results vary slightly depending on the random seed.

From these results, all models that contain an interaction between SEX OF PARTICIPANT and SEX OF REQUESTER are extracted and compared to the number of iterations (Listing 13). As can be seen, 2,961 out of 10,000 models (i.e., 29.61 %) contain the interaction.

Hide

```

# Extracting models where interaction of Sex of Participant and Sex of Requester
# shows up as significant
fp_models <- lapply(model_coefficients, function(x) {
  if ("SEX_OF_PARTICIPANTm:SEX_OF_REQUESTERm" %in% rownames(x)) {
    return(x)
  } else {
    return(NA)
  }
})
fp_models <- fp_models[!is.na(fp_models)]
# How many models include the false-positive interaction between sex of the
# participant and sex of the requester?
cat(round(100 * length(fp_models) / length(model_coefficients), 3), "%")

```

29.61 %

Listing 13: Counting the number of cases in which the interaction between SEX OF PARTICIPANT and SEX OF REQUESTER was kept in the model.

References

1. R Core Team. *R: A language and environment for statistical computing*. (R Foundation for Statistical Computing, 2017).
2. Bates, D., Mächler, M., Bolker, B. & Walker, S. Fitting linear mixed-effects models using lme4. *Journal of Statistical Software* **67**, 1–48 (2015).
3. Hothorn, T., Hornik, K. & Zeileis, A. Unbiased recursive partitioning: A conditional inference framework. *Journal of Computational and Graphical Statistics* **15**, 651–674 (2006).
4. Massen, J. J. M., Bauer, L., Spurny, B., Bugnyar, T. & Kret, M. E. Sharing of science is most likely among male scientists. *Nature Scientific Reports* **7**, 1–5 (2017).