AQ2: Changing usefulness beliefs

Nele Albers

10 December, 2024

Contents

This file is meant to reproduce our analysis of the impact of the nine persuasive activities on the nine corresponding usefulness beliefs based on paired Bayesian t-tests. Specifically, we reproduce Table 5.2 from the chapter as well as the Cohen's d values we report for AQ2.

Authored by Nele Albers, Mark A. Neerincx, and Willem-Paul Brinkman.

These files are required: Data/data_rl_samples_binary.csv

Setup

Let's import the packages we need.

```
library(BayesianFirstAid)  # for the t-tests
```

```
## Loading required package: rjags
```

```
## Loading required package: coda
```

```
## Linked to JAGS 4.3.0
```

```
## Loaded modules: basemod,bugs
```

```
library (formatR)  # for formatting
library(pander)  # for creating a table
library(tidyr)  # for removing na values
```

And we load our data.

```
df = read.csv("Data/data_rl_samples_binary.csv")
```

Let's create some vectors to store our data for the nine persuasive activities.

```
num_persuasive_activities = 9
hdi_lower_list = vector( ,num_persuasive_activities)
hdi_upper_list = vector( ,num_persuasive_activities)
mean_list = vector( ,num_persuasive_activities)
sd_list = vector( ,num_persuasive_activities)
prob_list = vector( ,num_persuasive_activities)
```

Conduct Bayesian t-tests

Now we conduct one paired Bayesian t-test for each persuasive activity, comparing the values of the corresponding usefulness beliefs before and after the activity.

```
for (pers_activity_idx in 1:num_persuasive_activities){

  set.seed(1)

  # Get samples where this activity was proposed
  df_pers_activity = df[df$activity == 43 + pers_activity_idx, ]

  col1 = paste0("s0_B", pers_activity_idx -1)
  col2 = paste0("s1_B", pers_activity_idx -1)

  # Drop samples with na for the second belief
  df_pers_activity = df_pers_activity %>% drop_na(col2)

  # Compute Bayesian t-test
  fit <- bayes.t.test(df_pers_activity[, col2], df_pers_activity[, col1], paired=TRUE)
  results <- summary(fit)
  hdi_lower <- results[1, 5]  # lower bound of 95% HDI
  hdi_lower_list[pers_activity_idx] = hdi_lower
  hdi_upper <- results[1, 6]  # upper bound of 95% HDI
  hdi_upper_list[pers_activity_idx] = hdi_upper
  mean_list[pers_activity_idx] = results[1, 1]
  sd_list[pers_activity_idx] = results[1, 2]
  prob_list[pers_activity_idx] = results[1,7]
}
```

```
## Note: Using an external vector in selections is ambiguous.
## i Use `all_of(col2)` instead of `col2` to silence this message.
## i See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This message is displayed once per session.
```

```
##   Data
## df_pers_activity[, col2], n = 120
## df_pers_activity[, col1], n = 120
##
##   Model parameters and generated quantities
## mu_diff: the mean pairwise difference between df_pers_activity[, col2] and df_pers_activity[, col1]
## sigma_diff: the scale of the pairwise difference, a consistent
##   estimate of SD when nu is large.
## nu: the degrees-of-freedom for the t distribution fitted to the pairwise difference
## eff_size: the effect size calculated as (mu_diff - 0) / sigma_diff
## diff_pred: predicted distribution for a new datapoint generated
##   as the pairwise difference between df_pers_activity[, col2] and df_pers_activity[, col1]
##
##   Measures
##              mean      sd  HDIlo HDIup %<comp %>comp
## mu_diff     0.350   0.173  0.012 0.693  0.016  0.984
## sigma_diff  1.425   0.289  0.918 2.019  0.000  1.000
## nu          1.396   0.345  1.000 2.053  0.000  1.000
## eff_size    0.248   0.116  0.016 0.473  0.016  0.984
## diff_pred  -0.417 226.290 -9.452 9.537  0.417  0.583
```

```
##
## 'HDIlo' and 'HDIup' are the limits of a 95% HDI credible interval.
## '%<comp' and '%>comp' are the probabilities of the respective parameter being
## smaller or larger than 0.
##
##   Quantiles
##             q2.5%   q25% median  q75% q97.5%
## mu_diff     0.031  0.233  0.342 0.460  0.715
## sigma_diff  0.955  1.220  1.390 1.597  2.081
## nu          1.016  1.151  1.310 1.547  2.270
## eff_size    0.022  0.170  0.247 0.326  0.479
## diff_pred  -9.162 -0.887  0.346 1.617  9.930
##   Data
## df_pers_activity[, col2], n = 121
## df_pers_activity[, col1], n = 121
##
##   Model parameters and generated quantities
## mu_diff: the mean pairwise difference between df_pers_activity[, col2] and df_pers_activity[, col1]
## sigma_diff: the scale of the pairwise difference, a consistent
##   estimate of SD when nu is large.
## nu: the degrees-of-freedom for the t distribution fitted to the pairwise difference
## eff_size: the effect size calculated as (mu_diff - 0) / sigma_diff
## diff_pred: predicted distribution for a new datapoint generated
##   as the pairwise difference between df_pers_activity[, col2] and df_pers_activity[, col1]
##
##   Measures
##             mean    sd  HDIlo HDIup %<comp %>comp
## mu_diff    0.171 0.177 -0.176 0.519  0.164  0.836
## sigma_diff 1.562 0.273  1.053 2.110  0.000  1.000
## nu         2.101 0.794  1.007 3.429  0.000  1.000
## eff_size   0.112 0.114 -0.109 0.338  0.164  0.836
## diff_pred  0.159 6.739 -6.573 7.268  0.463  0.537
##
## 'HDIlo' and 'HDIup' are the limits of a 95% HDI credible interval.
## '%<comp' and '%>comp' are the probabilities of the respective parameter being
## smaller or larger than 0.
##
##   Quantiles
##             q2.5%   q25% median  q75% q97.5%
## mu_diff    -0.174  0.053  0.169 0.287  0.521
## sigma_diff  1.088  1.371  1.539 1.728  2.162
## nu          1.182  1.609  1.940 2.386  3.903
## eff_size   -0.111  0.034  0.111 0.188  0.337
## diff_pred  -6.819 -1.106  0.167 1.445  7.069
##   Data
## df_pers_activity[, col2], n = 124
## df_pers_activity[, col1], n = 124
##
##   Model parameters and generated quantities
## mu_diff: the mean pairwise difference between df_pers_activity[, col2] and df_pers_activity[, col1]
## sigma_diff: the scale of the pairwise difference, a consistent
##   estimate of SD when nu is large.
## nu: the degrees-of-freedom for the t distribution fitted to the pairwise difference
## eff_size: the effect size calculated as (mu_diff - 0) / sigma_diff
```

```
## diff_pred: predicted distribution for a new datapoint generated
##   as the pairwise difference between df_pers_activity[, col2] and df_pers_activity[, col1]
##
##   Measures
##             mean    sd  HDIlo HDIup %<comp %>comp
## mu_diff    0.307 0.155 -0.003 0.606  0.023  0.977
## sigma_diff 1.374 0.200  0.999 1.772  0.000  1.000
## nu         2.362 0.734  1.204 3.815  0.000  1.000
## eff_size   0.228 0.117 -0.006 0.452  0.023  0.977
## diff_pred  0.276 4.954 -4.823 6.029  0.418  0.582
##
## 'HDIlo' and 'HDIup' are the limits of a 95% HDI credible interval.
## '%<comp' and '%>comp' are the probabilities of the respective parameter being
## smaller or larger than 0.
##
##   Quantiles
##             q2.5%   q25% median  q75% q97.5%
## mu_diff     0.005  0.204  0.305 0.409  0.615
## sigma_diff  1.018  1.234  1.363 1.502  1.798
## nu          1.356  1.850  2.223 2.710  4.189
## eff_size    0.004  0.148  0.225 0.305  0.462
## diff_pred  -5.179 -0.782  0.320 1.394  5.706
##   Data
## df_pers_activity[, col2], n = 120
## df_pers_activity[, col1], n = 120
##
##   Model parameters and generated quantities
## mu_diff: the mean pairwise difference between df_pers_activity[, col2] and df_pers_activity[, col1]
## sigma_diff: the scale of the pairwise difference, a consistent
##   estimate of SD when nu is large.
## nu: the degrees-of-freedom for the t distribution fitted to the pairwise difference
## eff_size: the effect size calculated as (mu_diff - 0) / sigma_diff
## diff_pred: predicted distribution for a new datapoint generated
##   as the pairwise difference between df_pers_activity[, col2] and df_pers_activity[, col1]
##
##   Measures
##             mean    sd  HDIlo  HDIup %<comp %>comp
## mu_diff    1.386 0.515  0.446  2.421  0.000  1.000
## sigma_diff 3.508 0.708  2.234  4.898  0.000  1.000
## nu         5.321 6.393  1.006 16.475  0.000  1.000
## eff_size   0.392 0.113  0.174  0.619  0.000  1.000
## diff_pred  1.420 8.002 -8.965 13.053  0.362  0.638
##
## 'HDIlo' and 'HDIup' are the limits of a 95% HDI credible interval.
## '%<comp' and '%>comp' are the probabilities of the respective parameter being
## smaller or larger than 0.
##
##   Quantiles
##             q2.5%   q25% median  q75% q97.5%
## mu_diff     0.482  1.015  1.343 1.727  2.474
## sigma_diff  2.256  2.988  3.467 4.010  4.926
## nu          1.433  2.289  3.279 5.435 24.831
## eff_size    0.167  0.316  0.392 0.469  0.612
## diff_pred  -9.598 -1.253  1.321 4.041 12.526
```

```
##   Data
## df_pers_activity[, col2], n = 111
## df_pers_activity[, col1], n = 111
##
##   Model parameters and generated quantities
## mu_diff: the mean pairwise difference between df_pers_activity[, col2] and df_pers_activity[, col1]
## sigma_diff: the scale of the pairwise difference, a consistent
##   estimate of SD when nu is large.
## nu: the degrees-of-freedom for the t distribution fitted to the pairwise difference
## eff_size: the effect size calculated as (mu_diff - 0) / sigma_diff
## diff_pred: predicted distribution for a new datapoint generated
##   as the pairwise difference between df_pers_activity[, col2] and df_pers_activity[, col1]
##
##   Measures
##             mean     sd  HDIlo  HDIup %<comp %>comp
## mu_diff    0.583  0.277  0.050  1.133  0.013  0.987
## sigma_diff 2.267  0.386  1.538  3.033  0.000  1.000
## nu         2.301  0.890  1.081  3.914  0.000  1.000
## eff_size   0.260  0.120  0.019  0.487  0.013  0.987
## diff_pred  0.539 18.971 -8.658 10.344  0.412  0.588
##
## 'HDIlo' and 'HDIup' are the limits of a 95% HDI credible interval.
## '%<comp' and '%>comp' are the probabilities of the respective parameter being
## smaller or larger than 0.
##
##   Quantiles
##             q2.5%   q25% median  q75% q97.5%
## mu_diff     0.065  0.394  0.574 0.765  1.152
## sigma_diff  1.597  1.996  2.239 2.505  3.110
## nu          1.252  1.731  2.111 2.635  4.464
## eff_size    0.028  0.179  0.259 0.340  0.497
## diff_pred  -8.984 -1.256  0.559 2.394 10.066
##   Data
## df_pers_activity[, col2], n = 110
## df_pers_activity[, col1], n = 110
##
##   Model parameters and generated quantities
## mu_diff: the mean pairwise difference between df_pers_activity[, col2] and df_pers_activity[, col1]
## sigma_diff: the scale of the pairwise difference, a consistent
##   estimate of SD when nu is large.
## nu: the degrees-of-freedom for the t distribution fitted to the pairwise difference
## eff_size: the effect size calculated as (mu_diff - 0) / sigma_diff
## diff_pred: predicted distribution for a new datapoint generated
##   as the pairwise difference between df_pers_activity[, col2] and df_pers_activity[, col1]
##
##   Measures
##             mean      sd  HDIlo HDIup %<comp %>comp
## mu_diff    0.396   0.183  0.042 0.757  0.013  0.987
## sigma_diff 1.445   0.243  1.011 1.942  0.000  1.000
## nu         1.453   0.315  1.000 2.051  0.000  1.000
## eff_size   0.278   0.126  0.030 0.527  0.013  0.987
## diff_pred  6.949 988.494 -9.706 9.528  0.414  0.586
##
## 'HDIlo' and 'HDIup' are the limits of a 95% HDI credible interval.
```

```
## '%<comp' and '%>comp' are the probabilities of the respective parameter being
## smaller or larger than 0.
##
##   Quantiles
##             q2.5%   q25% median  q75% q97.5%
## mu_diff     0.049  0.272  0.391 0.515  0.766
## sigma_diff  1.032  1.274  1.425 1.594  1.982
## nu          1.033  1.222  1.394 1.614  2.219
## eff_size    0.034  0.192  0.275 0.361  0.532
## diff_pred  -9.346 -0.909  0.382 1.661 10.003
##   Data
## df_pers_activity[, col2], n = 122
## df_pers_activity[, col1], n = 122
##
##   Model parameters and generated quantities
## mu_diff: the mean pairwise difference between df_pers_activity[, col2] and df_pers_activity[, col1]
## sigma_diff: the scale of the pairwise difference, a consistent
##   estimate of SD when nu is large.
## nu: the degrees-of-freedom for the t distribution fitted to the pairwise difference
## eff_size: the effect size calculated as (mu_diff - 0) / sigma_diff
## diff_pred: predicted distribution for a new datapoint generated
##   as the pairwise difference between df_pers_activity[, col2] and df_pers_activity[, col1]
##
##   Measures
##             mean     sd  HDIlo HDIup %<comp %>comp
## mu_diff    0.427  0.198  0.037 0.813  0.015  0.985
## sigma_diff 1.705  0.250  1.235 2.205  0.000  1.000
## nu         1.959  0.506  1.112 2.952  0.000  1.000
## eff_size   0.254  0.118  0.027 0.490  0.015  0.985
## diff_pred  0.469 10.695 -7.243 8.459  0.411  0.589
##
## 'HDIlo' and 'HDIup' are the limits of a 95% HDI credible interval.
## '%<comp' and '%>comp' are the probabilities of the respective parameter being
## smaller or larger than 0.
##
##   Quantiles
##             q2.5%   q25% median  q75% q97.5%
## mu_diff     0.045  0.294  0.424 0.558  0.824
## sigma_diff  1.254  1.531  1.690 1.866  2.234
## nu          1.214  1.599  1.876 2.231  3.173
## eff_size    0.027  0.174  0.252 0.331  0.489
## diff_pred  -7.377 -0.986  0.433 1.850  8.337
##   Data
## df_pers_activity[, col2], n = 124
## df_pers_activity[, col1], n = 124
##
##   Model parameters and generated quantities
## mu_diff: the mean pairwise difference between df_pers_activity[, col2] and df_pers_activity[, col1]
## sigma_diff: the scale of the pairwise difference, a consistent
##   estimate of SD when nu is large.
## nu: the degrees-of-freedom for the t distribution fitted to the pairwise difference
## eff_size: the effect size calculated as (mu_diff - 0) / sigma_diff
## diff_pred: predicted distribution for a new datapoint generated
##   as the pairwise difference between df_pers_activity[, col2] and df_pers_activity[, col1]
```

```
##
##   Measures
##             mean    sd  HDIlo HDIup %<comp %>comp
## mu_diff    1.160 0.271  0.632 1.699  0.000  1.000
## sigma_diff 2.432 0.339  1.789 3.101  0.000  1.000
## nu         2.790 1.071  1.271 4.677  0.000  1.000
## eff_size   0.487 0.133  0.234 0.754  0.000  1.000
## diff_pred  1.176 5.986 -7.234 9.921  0.337  0.663
##
## 'HDIlo' and 'HDIup' are the limits of a 95% HDI credible interval.
## '%<comp' and '%>comp' are the probabilities of the respective parameter being
## smaller or larger than 0.
##
##   Quantiles
##             q2.5%   q25% median  q75% q97.5%
## mu_diff     0.623  0.981  1.160 1.340  1.689
## sigma_diff  1.831  2.197  2.411 2.641  3.160
## nu          1.532  2.119  2.565 3.173  5.381
## eff_size    0.237  0.396  0.483 0.574  0.758
## diff_pred  -7.397 -0.741  1.161 3.038  9.830
##   Data
## df_pers_activity[, col2], n = 120
## df_pers_activity[, col1], n = 120
##
##   Model parameters and generated quantities
## mu_diff: the mean pairwise difference between df_pers_activity[, col2] and df_pers_activity[, col1]
## sigma_diff: the scale of the pairwise difference, a consistent
##   estimate of SD when nu is large.
## nu: the degrees-of-freedom for the t distribution fitted to the pairwise difference
## eff_size: the effect size calculated as (mu_diff - 0) / sigma_diff
## diff_pred: predicted distribution for a new datapoint generated
##   as the pairwise difference between df_pers_activity[, col2] and df_pers_activity[, col1]
##
##   Measures
##             mean     sd  HDIlo HDIup %<comp %>comp
## mu_diff    0.230  0.207 -0.182 0.633  0.126  0.874
## sigma_diff 1.790  0.293  1.233 2.370  0.000  1.000
## nu         2.939  1.928  1.164 5.337  0.000  1.000
## eff_size   0.127  0.112 -0.103 0.340  0.126  0.874
## diff_pred  0.317 11.662 -6.414 6.465  0.452  0.548
##
## 'HDIlo' and 'HDIup' are the limits of a 95% HDI credible interval.
## '%<comp' and '%>comp' are the probabilities of the respective parameter being
## smaller or larger than 0.
##
##   Quantiles
##             q2.5%   q25% median  q75% q97.5%
## mu_diff    -0.162  0.089  0.224 0.361  0.657
## sigma_diff  1.284  1.589  1.767 1.962  2.442
## nu          1.428  2.040  2.529 3.248  6.737
## eff_size   -0.096  0.052  0.127 0.202  0.349
## diff_pred  -6.264 -1.162  0.233 1.651  6.701
```

Below we print a summary of the Bayesian t-test results. This reproduces Table 2 from the paper.

```
competencies = c("Self-efficacy", "Practical knowledge",
    "Awareness of positive outcomes", "Awareness of negative outcomes",
    "Motivation to change", "Knowledge of how to maintain/achieve mental well-being",
    "Mindset that physical activity helps to quit smoking",
    "Awareness of smoking patterns", "Knowledge of how to maintain/achieve well-being")

tab <- rbind(c("Self-efficacy", round(mean_list[1],
    2), round(sd_list[1], 2), paste("[", round(hdi_lower_list[1],
    2), ",", round(hdi_upper_list[1], 2), "]"), round(prob_list[1],
    4)))

for (pers_activity_idx in 2:num_persuasive_activities) {
    tab <- rbind(tab, c(competencies[pers_activity_idx],
        round(mean_list[pers_activity_idx], 2), round(sd_list[pers_activity_idx],
            2), paste("[", round(hdi_lower_list[pers_activity_idx],
            2), ",", round(hdi_upper_list[pers_activity_idx],
            2), "]"), round(prob_list[pers_activity_idx],
            4)))
}

colnames(tab) = c("Competency", "Mean", "SD", "95% HDI",
    "Prob > 0")

pander(tab, caption = "Impact of the persuasive activities on the corresponding usefulness beliefs base
```

Table 1: Impact of the persuasive activities on the corresponding usefulness beliefs based on paired Bayesian t-tests.

Competency	Mean	SD	95% HDI	Prob > 0
Self-efficacy	0.35	0.17	[0.01 , 0.69]	0.9845
Practical knowledge	0.17	0.18	[-0.18 , 0.52]	0.8358
Awareness of positive outcomes	0.31	0.15	[0 , 0.61]	0.9767
Awareness of negative outcomes	1.39	0.52	[0.45 , 2.42]	0.9996
Motivation to change	0.58	0.28	[0.05 , 1.13]	0.9868
Knowledge of how to maintain/achieve mental well-being	0.4	0.18	[0.04 , 0.76]	0.9875
Mindset that physical activity helps to quit smoking	0.43	0.2	[0.04 , 0.81]	0.9854
Awareness of smoking patterns	1.16	0.27	[0.63 , 1.7]	0.9999
Knowledge of how to maintain/achieve well-being	0.23	0.21	[-0.18 , 0.63]	0.8736

Effect size (Cohen's d)

Let's also compute the effect sizes of the differences based on Cohen's d.

```
for (pers_activity_idx in 1:num_persuasive_activities){

  # Get samples where this activity was proposed
  # Activity indices start at 0 and there are 44 preparatory activities, so
  # the persuasive activities start at index 44.
  df_pers_activity = df[df$activity == 43 + pers_activity_idx, ]
```

```
  col1 = paste0("s0_B", pers_activity_idx -1)
  col2 = paste0("s1_B", pers_activity_idx -1)

  # Drop samples with na for the second belief value
  df_pers_activity = df_pers_activity %>% drop_na(col2)

  # Compute the difference
  diff = df_pers_activity[, col2] - df_pers_activity[, col1]

  effect_size = mean(diff)/sd(diff)

  print(paste(competencies[pers_activity_idx], ":", round(effect_size, 2)))
}
```

```
## [1] "Self-efficacy : 0.14"
## [1] "Practical knowledge : 0.01"
## [1] "Awareness of positive outcomes : 0.09"
## [1] "Awareness of negative outcomes : 0.45"
## [1] "Motivation to change : 0.17"
## [1] "Knowledge of how to maintain/achieve mental well-being : 0.32"
## [1] "Mindset that physical activity helps to quit smoking : 0.13"
## [1] "Awareness of smoking patterns : 0.18"
## [1] "Knowledge of how to maintain/achieve well-being : 0.2"
```