Effects of all actions on confidence and perceived usefulness

Andrei Stefan

09 November, 2023

This file contains the code for the analysis of the effects of multiple actions on confidence and perceived usefulness. Note that, due to the nature of the Bayesian t-tests, results will not always be the same for the means and standard deviations. From the output of the `print` of the result, we use the probability of the mean difference being larger and smaller than 0. From the output of the `summary` of the result, we use the mean and standard deviation values, and their 95% CIs.

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

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

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

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

Get the confidence and perceived usefulness values before and after the whole interaction as lists.

```
# get the confidence before as a list
c_before = as.numeric(t(read.csv("~/analysis/data/c_before.csv", header = FALSE)))
# get the confidence after as a list
c_after = as.numeric(t(read.csv("~/analysis/data/c_after.csv", header = FALSE)))
# get the perceived usefulness before as a list
pu_before = as.numeric(t(read.csv("~/analysis/data/pu_before.csv", header = FALSE)))
# get the perceived usefulness after as a list
pu_after = as.numeric(t(read.csv("~/analysis/data/pu_after.csv", header = FALSE)))
```

Run a Bayesian t-test on the confidence values.

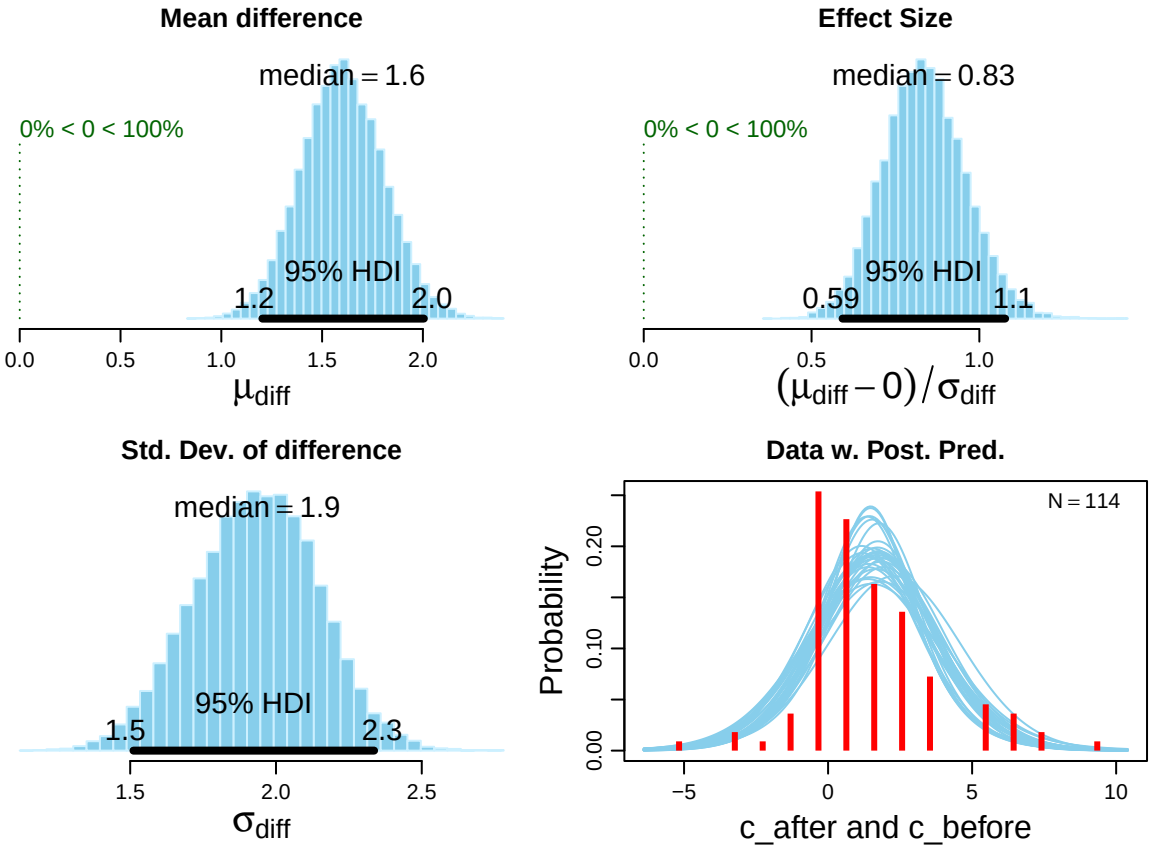
```
# run the Bayesian t-test
fit_1 = bayes.t.test(c_after, c_before, paired=TRUE)

# print the results of the test, the summary, and plot the results
print(fit_1)
```

```
##
##  Bayesian estimation supersedes the t test (BEST) - paired samples
##
## data: c_after and c_before, n = 114
##
##   Estimates [95% credible interval]
## mean paired difference: 1.6 [1.2, 2.0]
## sd of the paired differences: 1.9 [1.5, 2.3]
##
## The mean difference is more than 0 by a probability of >0.999
## and less than 0 by a probability of <0.001
```

```
summary(fit_1)
```

```
##   Data
## c_after, n = 114
## c_before, n = 114
##
##   Model parameters and generated quantities
## mu_diff: the mean pairwise difference between c_after and c_before
## 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 c_after and c_before
##
##   Measures
##              mean     sd  HDIlo  HDIup %<comp %>comp
## mu_diff     1.601  0.205  1.204  2.005  0.000  1.000
## sigma_diff  1.931  0.213  1.511  2.337  0.000  1.000
## nu         18.995 22.758  1.664 61.469  0.000  1.000
## eff_size    0.837  0.124  0.592  1.077  0.000  1.000
## diff_pred   1.596  2.310 -2.817  5.949  0.215  0.785
##
## '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     1.202 1.462  1.599  1.740  2.004
## sigma_diff  1.512 1.786  1.936  2.079  2.338
## nu          3.223 6.418 11.065 22.114 83.510
## eff_size    0.603 0.753  0.834  0.917  1.090
## diff_pred  -2.811 0.244  1.601  2.974  5.957
```

```
plot(fit_1)
```



Run a Bayesian *t*-test on the perceived usefulness values.

```
fit_2 = bayes.t.test(pu_after, pu_before, paired=TRUE)
print(fit_2)
```

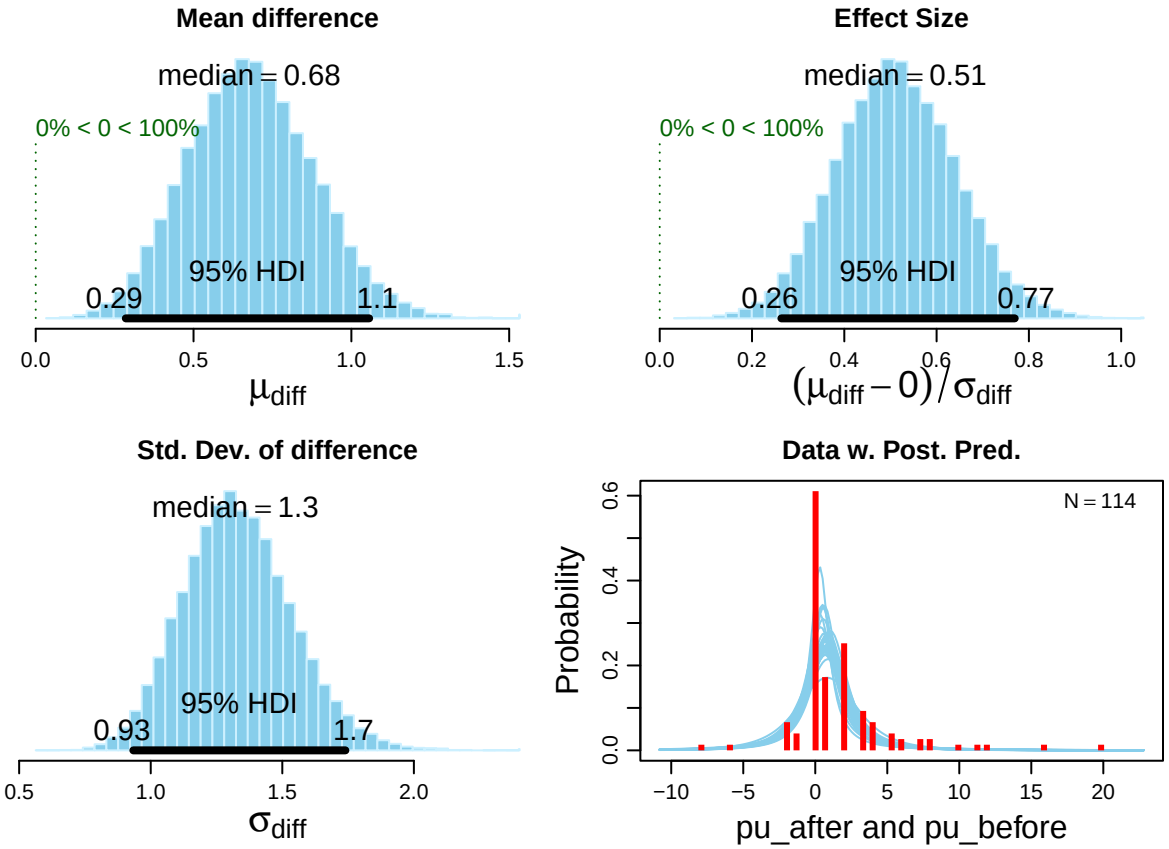
```
##
##  Bayesian estimation supersedes the t test (BEST) - paired samples
##
## data: pu_after and pu_before, n = 114
##
##   Estimates [95% credible interval]
## mean paired difference: 0.68 [0.29, 1.1]
## sd of the paired differences: 1.3 [0.93, 1.7]
##
## The mean difference is more than 0 by a probability of >0.999
## and less than 0 by a probability of <0.001
```

```
summary(fit_2)
```

```
##   Data
## pu_after, n = 114
## pu_before, n = 114
##
##   Model parameters and generated quantities
## mu_diff: the mean pairwise difference between pu_after and pu_before
## 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 pu_after and pu_before
##
##   Measures
##              mean     sd  HDIlo HDIup %<comp %>comp
## mu_diff     0.682  0.199  0.285 1.056  0.000  1.000
## sigma_diff 1.332  0.208  0.934 1.740  0.000  1.000
## nu          1.688  0.400  1.000 2.423  0.000  1.000
## eff_size    0.513  0.130  0.264 0.769  0.000  1.000
## diff_pred   0.645 14.022 -6.376 8.232  0.339  0.661
##
## '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      0.312  0.543  0.677 0.815  1.086
## sigma_diff   0.953  1.188  1.322 1.465  1.763
## nu           1.091  1.400  1.629 1.901  2.635
## eff_size     0.265  0.424  0.510 0.600  0.772
## diff_pred   -6.545 -0.474  0.669 1.851  8.104
```

```
plot(fit_2)
```



Calculate Cohen's d for the values of confidence and perceived usefulness before and after.

```r
# calculate Cohen's d
cohen.d(c_after, c_before)
```

```
##
## Cohen's d
##
## d estimate: 0.7533302 (medium)
## 95 percent confidence interval:
##     lower     upper
## 0.4832301 1.0234304
```

```r
cohen.d(pu_after, pu_before)
```

```
##
## Cohen's d
##
## d estimate: 0.4344207 (small)
## 95 percent confidence interval:
##     lower     upper
## 0.1703589 0.6984825
```