DAS Study - Simulation Analysis

Simulations to validate the statistical model and to understand power and precision for the study design

Thomas Klebel

2024-10-30

First, we load the data from our simulation and clean it.

```
library(tidyverse)
library(tidytext)

df <- read_csv("combined_sim_res.csv")

df_clean <- df %>%
  mutate(divergent_transitions = str_extract(warnings, "There were (\\d+)",
                                             group = TRUE) %>% as.numeric(),
         ESS_issue = str_detect(warnings, "ESS"),
         r_hat_issue = str_detect(warnings, "R-hat")) %>%
  replace_na(list(divergent_transitions = 0, ESS_issue = FALSE,
                  r_hat_issue = FALSE)) %>%
  filter(term == "intervention")

true_eff_small <- .044
true_eff_large <- .089
```

We check the simulations for potential warnings.

```
n_warnings <- df_clean %>%
  filter(!is.na(warnings)) %>%
  nrow()

stopifnot(identical(n_warnings, 0L))
```

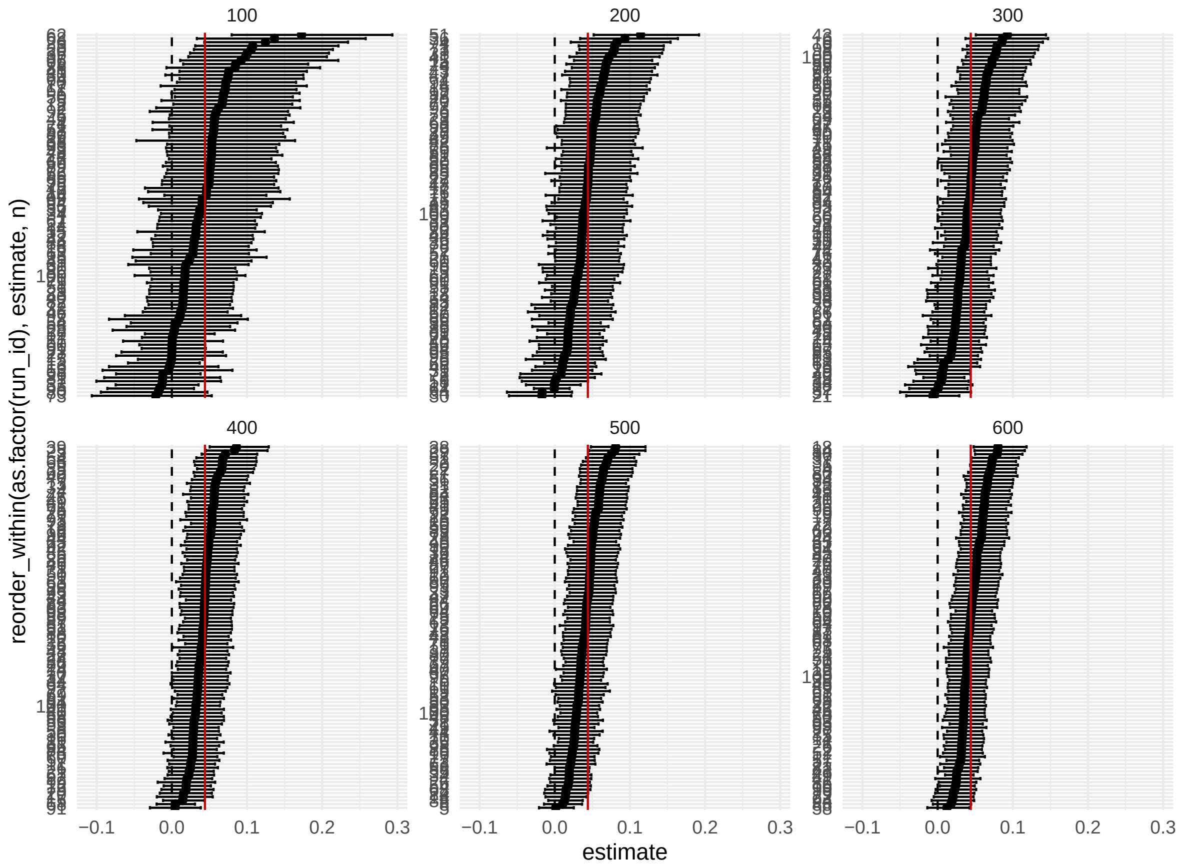
Success, we have no warnings, divergent transitions or other issues that might impair our estimation.

Small effect

We first explore the outcomes for the small effect. We display marginal effects on the probability scale (median estimate of the posterior) with the respective 95% credibility-intervals.

```
pdata <- df_clean %>%
  filter(round(true_effect, 1) == 0, n %in% seq(100, 600, by = 100))

pdata %>%
  ggplot(aes(x = estimate,
             y = reorder_within(as.factor(run_id), estimate, n))) +
  geom_errorbar(aes(xmin = conf.low, xmax = conf.high)) +
  geom_point() +
  geom_vline(xintercept = true_eff_small, colour = "red3") +
  geom_vline(xintercept = 0, linetype = 2) +
  scale_y_reordered() +
  theme_minimal() +
  facet_wrap(vars(n), scales = "free_y")
```



We clearly see that as the sample size increases, our estimates become less variable in terms of credibility intervals.

We further check some numbers.

```
# check for bias and power
df_clean %>%
  filter(round(true_effect, 1) == 0) %>%
  group_by(n) %>%
  summarise(med_eff = median(estimate),
            power = sum(conf.low > 0) / 100,
            median_interval_width = median(conf.high - conf.low)) %>%
  knitr::kable()
```

n	med_eff	power	median_interval_width
100	0.0359320	0.15	0.1492686
200	0.0375386	0.48	0.1001600

n	med_eff	power	median_interval_width
300	0.0390763	0.60	0.0837939
400	0.0415924	0.75	0.0705885
500	0.0410565	0.76	0.0628803
600	0.0427671	0.93	0.0563502
700	0.0400286	0.89	0.0510480
800	0.0443544	0.96	0.0489008
900	0.0438805	0.99	0.0465935
1000	0.0427503	1.00	0.0429024
1100	0.0458990	1.00	0.0425939
1200	0.0442296	1.00	0.0397487
1300	0.0447593	1.00	0.0385025
1400	0.0434075	1.00	0.0369142
1500	0.0445430	0.99	0.0355206

From the table, we can make a couple of conclusions:

1. The median effect size across 100 runs per sample size is slightly lower than the true value for smaller sample sizes. This is expected and desirable, with the prior on the beta coefficients having a larger impact on the final estimates for small effect sizes.
2. We further compute the power, by calculating the proportion of runs where the credibility interval does not include 0. With a sample size of 400-500, we reach a power of about 75%. The power increases to 90% and above for larger sample sizes.
3. The credibility intervals get narrower for higher sample sizes. With a sample size of 500, the credibility interval covers a range of 6.3 percentage points, with a median estimate of a 4.1% higher rate of data sharing in the intervention group.

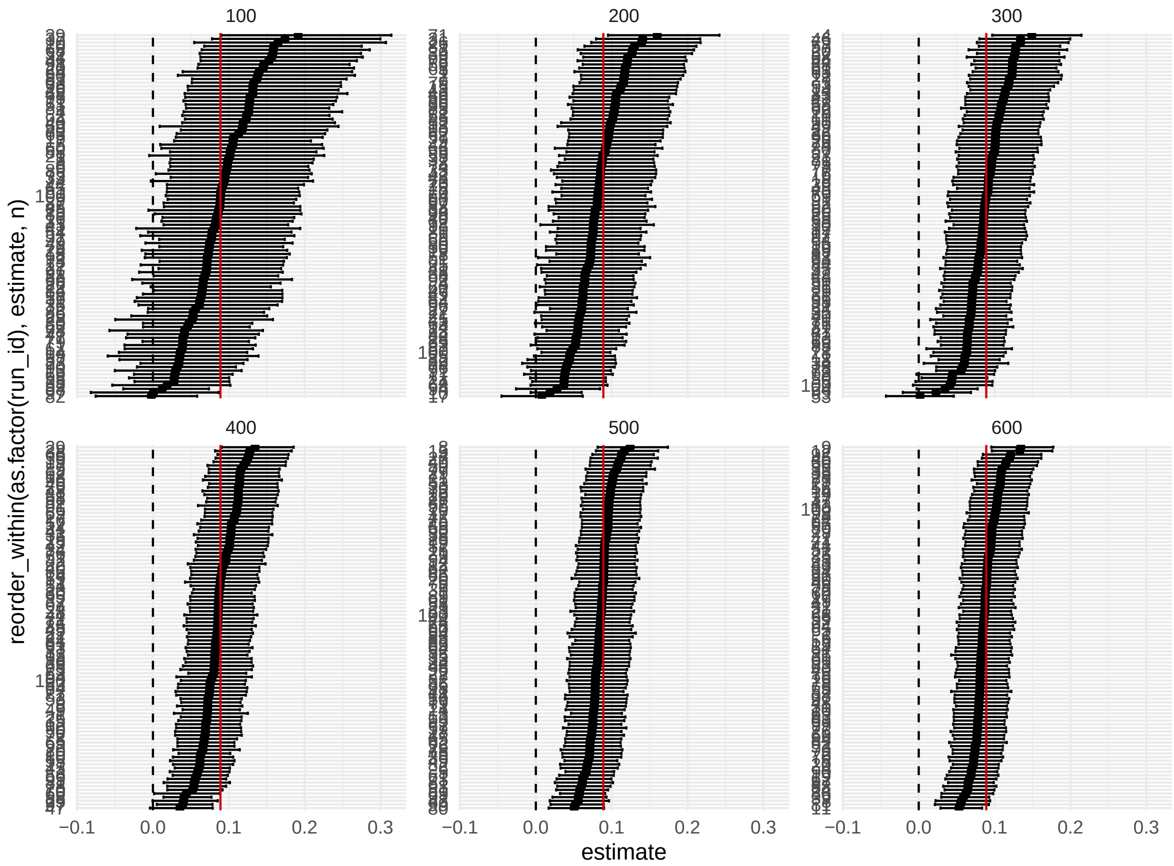
Overall, a sample size of 400 or higher seems to provide reasonable precision and power, even for such a small effect.

Larger effect

```
pdata <- df_clean %>%
  filter(round(true_effect, 1) == 0.1, n %in% seq(100, 600, by = 100))

pdata %>%
  ggplot(aes(x = estimate,
             y = reorder_within(as.factor(run_id), estimate, n))) +
  geom_errorbar(aes(xmin = conf.low, xmax = conf.high)) +
```

```
  geom_point() +
  geom_vline(xintercept = true_eff_large, colour = "red3") +
  geom_vline(xintercept = 0, linetype = 2) +
  scale_y_reordered() +
  theme_minimal() +
  facet_wrap(vars(n), scales = "free_y")
```




```
# check for bias and power
df_clean %>%
  filter(round(true_effect, 1) == .1) %>%
  group_by(n) %>%
  summarise(med_eff = median(estimate),
            power = sum(conf.low > 0) / 100,
            median_interval_width = median(conf.high - conf.low)) %>%
  knitr::kable()
```

n	med_eff	power	median_interval_width
100	0.0847525	0.56	0.1887878
200	0.0775052	0.83	0.1246589
300	0.0857746	0.94	0.1018027
400	0.0850884	0.98	0.0877950
500	0.0847282	1.00	0.0774991
600	0.0849092	1.00	0.0710023
700	0.0862748	1.00	0.0654723
800	0.0874275	1.00	0.0613889
900	0.0842603	1.00	0.0572648
1000	0.0869726	1.00	0.0547581
1100	0.0878136	1.00	0.0519826
1200	0.0886806	1.00	0.0502843
1300	0.0875663	1.00	0.0484477
1400	0.0853358	1.00	0.0461770
1500	0.0863515	1.00	0.0445284

Compared to the case of a small expected effect size, a smaller sample size is necessary to achieve high power in the case of the moderate effect size. With $n = 300$, we achieve 94% power, which increases further to 100% with larger samples. However, the variability of the estimates is higher than in the case with a small effect.