Analysis of Ratings for Interaction Scenarios

Nele Albers

27 June, 2022

Contents

Introduction

This file is to reproduce the means and 95% highest density intervals for the ratings per interaction scenario that we show in Figure 3.

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

Setup

First, we load the rethinking package, which we need to fit and sample from models. We also load the ggplot2 package for creating plots and formatR for formatting.

```
library(formatR)  # for formatting
library(ggplot2)  # For plots
library(rethinking)  # For Bayesian models
```

Also, we set the number of iterations and number of chains used for fitting the models.

```
# For fitting models
NUM_ITERATIONS = 2000  # our value: 2000
NUM_CHAINS = 4  # our value: 4
```

Data files

We load the pre-processed data.

```
df = read.csv(file = "Data/all_coded_data.csv")
```

Highest density interval per interaction scenario

Now we compute the mean and 95% highest density interval for each interaction scenario.

```
resp_types = c(c("gp_relapse"), c("gp_start"), c("discussion_so"),
    c("tell_se"), c("motivational_messages"), c("failing_pa_goals"),
    c("hrs_reflection_pa"), c("hrs_reflection_smoking"),
    c("hrs_help_button_pa"), c("hrs_help_button_smoking"),
    c("hrs_plan_pa"), c("hrs_plan_smoking"), c("pa_program"))

creds_low = numeric(length(resp_types))
creds_high = numeric(length(resp_types))
means = numeric(length(resp_types))
nums = numeric(length(resp_types))

i = 1

# For each interaction scenario
for (r in resp_types) {

    df_temp = df[df$response_type %in% r, ]

    # Create a data list to be used for the
    # model
    dat_list <- list(rating = df_temp$rating)

    set.seed(18)  # For reproducibility

    # Fit model
    ml <- ulam(alist(rating ~ normal(mu, sigma),
        mu <- a_bar, a_bar ~ dnorm(0, 10), sigma ~
            dexp(1)), data = dat_list, chains = NUM_CHAINS,
        log_lik = TRUE, cores = NUM_CHAINS, iter = NUM_ITERATIONS)

    res = precis(ml, prob = 0.95)

    means[i] = res[1, 1]
    nums[i] = length(df_temp$rating)

    # Sample from posterior
    set.seed(18)  # For reproducibility
    samples <- extract.samples(ml, n = 10000)

    # Compute 95% HDI
    hdpi <- HPDI(samples$a_bar, prob = 0.95)
    creds_low[i] = hdpi[1]
    creds_high[i] = hdpi[2]

    i = i + 1

    rm(ml)
}
```

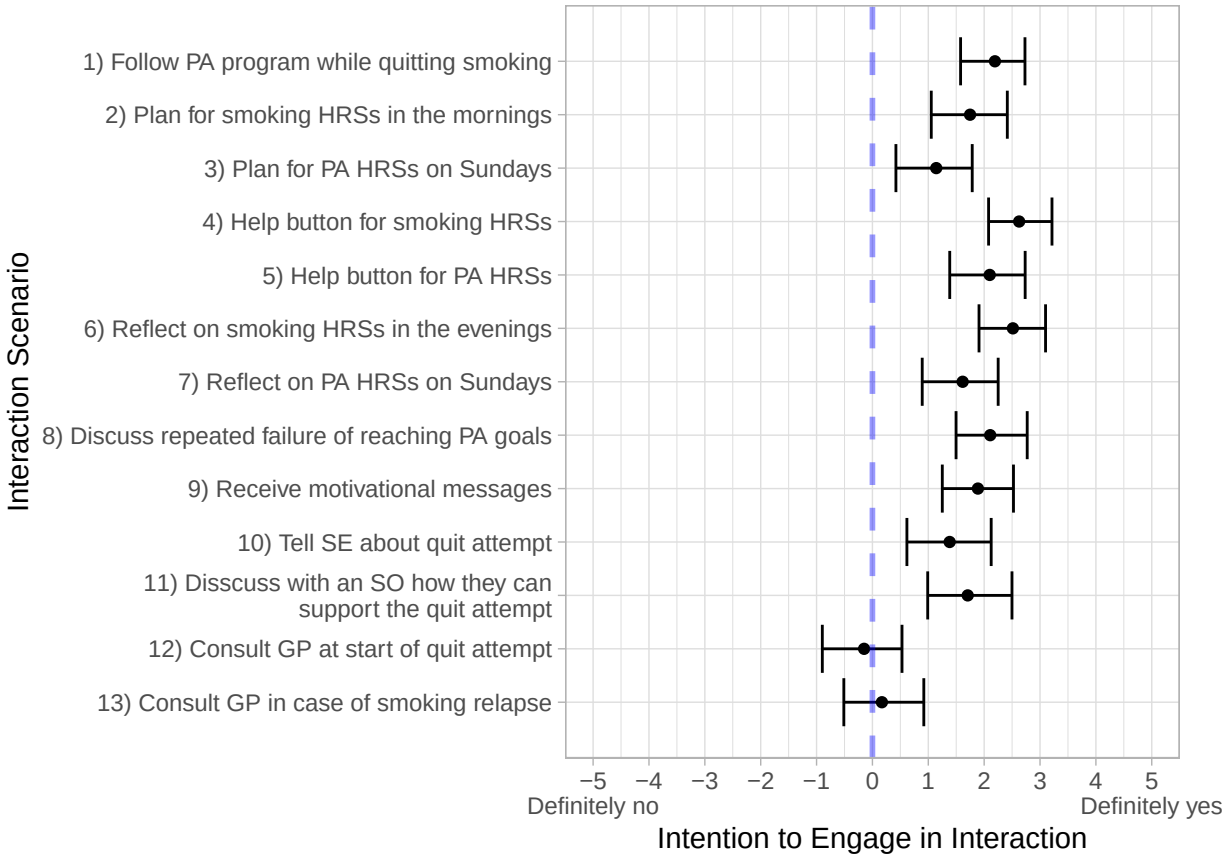
Create plot

Now we plot the mean and 95% highest density interval of the ratings per interaction scenario in a single figure.

```
  data = data.frame(means, nums, creds_low, creds_high)  # create data frame
  data$row_num <- seq.int(nrow(data))  # add row number

  ggplot(data, aes(means, row_num)) +        # ggplot2 plot with confidence intervals
  geom_point() +
  geom_errorbar(aes(xmin = creds_low, xmax = creds_high)) +
  theme_light() +
  ylab("Interaction Scenario") +
  scale_y_discrete(limits = c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13), labels = c("13) Consult GP in
  theme(plot.margin = margin(t = 0.1, r = 0.9, b = 0.1, l = 0.1, "cm"))
```

```
## Warning: Continuous limits supplied to discrete scale.
## Did you mean 'limits = factor(...)' or 'scale_*_continuous()'?
```




```
# Save image
pdf_file <- "Figures/interaction_scenarios_ratings_hdis.pdf"
ggsave(pdf_file, dpi=1500)
```

```
## Saving 6.5 x 4.5 in image
```