F06_mci_style_neu_sensitvity.R

2024-01-23

```
## MCI_STYLE_NEU SENSITIVITY ANALYSIS SCRIPT ##

# Runs a sensitivity analysis for the (nested) effect of interest ("stylenor:semantics2")
# for the verb and picture models. For this sensitivity analysis, the effect size is changed
# linearly from 0.1 to 1.0 µV, and the model is re-fitted many times tom compute the power
# at each effect size (defined as the average number of simulated models for which the
# effect of interest is significant). This is used to determine the smallest effect size
# at which (given the current participant and item sample sizes) the effect can be detected
# with sufficient statistical power (e.g., 80 or 90%).

## SETUP ## ---------------------------------------------------------------------------------------------------

# Load packages
library(MASS)         # Version 7.3-51.6
library(lme4)         # Version 1.1-23
```

```
## Loading required package: Matrix
```

```
library(lmerTest)     # Version 3.1-2
```

```
## Warning: replacing previous import 'ellipsis::check_dots_unnamed' by
## 'rlang::check_dots_unnamed' when loading 'tibble'
```

```
## Warning: replacing previous import 'ellipsis::check_dots_used' by
## 'rlang::check_dots_used' when loading 'tibble'
```

```
## Warning: replacing previous import 'ellipsis::check_dots_empty' by
## 'rlang::check_dots_empty' when loading 'tibble'
## Warning: replacing previous import 'ellipsis::check_dots_unnamed' by
## 'rlang::check_dots_unnamed' when loading 'pillar'
```

```
## Warning: replacing previous import 'ellipsis::check_dots_used' by
## 'rlang::check_dots_used' when loading 'pillar'
```

```
## Warning: replacing previous import 'ellipsis::check_dots_empty' by
## 'rlang::check_dots_empty' when loading 'pillar'
```

```
## Warning: replacing previous import 'vctrs::data_frame' by 'tibble::data_frame' when
## loading 'dplyr'
```

```
## 
## Attaching package: 'lmerTest'
```

```
## The following object is masked from 'package:lme4':
## 
##     lmer
```

```
## The following object is masked from 'package:stats':
## 
##     step
```

```r
library(afex)         # Version 0.27-2
```

```
## Registered S3 methods overwritten by 'car':
##   method                          from
##   influence.merMod                lme4
##   cooks.distance.influence.merMod lme4
##   dfbeta.influence.merMod         lme4
##   dfbetas.influence.merMod        lme4
```

```
## ************
## Welcome to afex. For support visit: http://afex.singmann.science/
## - Functions for ANOVAs: aov_car(), aov_ez(), and aov_4()
## - Methods for calculating p-values with mixed(): ’KR’, ’S’, ’LRT’, and ’PB’
## - ’afex_aov’ and ’mixed’ objects can be passed to emmeans() for follow-up tests
## - NEWS: library(’emmeans’) now needs to be called explicitly!
## - Get and set global package options with: afex_options()
## - Set orthogonal sum-to-zero contrasts globally: set_sum_contrasts()
## - For example analyses see: browseVignettes("afex")
## ************

##
## Attaching package: ’afex’

## The following object is masked from ’package:lme4’:
##
##     lmer
```

```
library(emmeans)      # Version 1.4.8
library(tidyverse)    # Version 1.3.0
```

```
## -- Attaching packages ------------------------------------------------ tidyverse 1.3.0 --

## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.0
## v tidyr   1.1.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0

## -- Conflicts ---------------------------------------------------- tidyverse_conflicts() --
## x tidyr::expand() masks Matrix::expand()
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## x tidyr::pack()   masks Matrix::pack()
## x dplyr::select() masks MASS::select()
## x tidyr::unpack() masks Matrix::unpack()
```

```
library(magrittr)     # Version 1.5
```

```
##
## Attaching package: ’magrittr’
## The following object is masked from 'package:purrr':
##
##     set_names
```

```
## The following object is masked from 'package:tidyr':
##
##     extract
```

```r
library(simr)         # Version 1.0.7
```

```
##
## Attaching package: 'simr'
```

```
## The following object is masked from 'package:stringr':
##
##     fixed
```

```
## The following object is masked from 'package:lme4':
##
##     getData
```

```r
library(furrr)        # Version 0.3.1
```

```
## Loading required package: future
```

```r
library(glue)         # Version 1.4.1
```

```
##
## Attaching package: 'glue'
```

```
## The following object is masked from 'package:dplyr':
##
##     collapse
```

```r
library(testit)       # Version 0.13
```

```
## Warning: package 'testit' was built under R version 4.0.5
```

```r
library(tictoc)       # Version 1.2

# Load preprocessed data
a1 <- readRDS("EEG/export/a1.RDS")

# Remove trials with errors or invalid RTs/ERPs
a1 %<>% filter(!error) %>% na.omit()

# Center behavioral ratings (valence and arousal) around 0
a1 %<>% mutate(rating_1 = Rating1Resp - 2, rating_2 = Rating2Resp - 2)

# Define simple contrast coding for context narrative style (normal - fairytale)
#     H0(Intercept): (mu1+mu2)/2 = 0 <-> mu1+mu2 = 0
#     H0(Slope): -mu1 + mu2 = 0
#     with mu1 = mean of the normal style and mu2 = mean of the fairytale style
t(contrasts.style <- t(cbind(c("nor" = -1, "ftl" = 1))))
```

```
##     [,1]
## nor   -1
## ftl    1
```

```r
contrasts(a1$style) <- ginv(contrasts.style)

# Define simple contrast coding for semantics (violation - intuitive, mci - intuitive)
#     H0(Intercept): (mu1+mu2+mu3)/3 = 0 <-> mu1+mu2+mu3 = 0
#     H0(Slope1): -1*mu1 +1*mu2 + 0*mu3 = 0
#     H0(Slope2): -1*mu1 +0*mu2 + 1*mu3 = 0
#     with mu1 = mean of intuitive concepts, mu2 = mean of violations, mu3 = mean of MCIs
t(contrasts.semantics <- t(cbind(c("int" = -1, "vio" = 1, "mci" = 0),
                                 c("int" = -1, "vio" = 0, "mci" = 1))))
```

```
##     [,1] [,2]
## int   -1   -1
## vio    1    0
## mci    0    1
```

```r
contrasts(a1$semantics) <- ginv(contrasts.semantics)

## LINEAR MIXED-EFFECTS MODELS ## --------------------------------------------------------------------------------------

# Specify settings for optimization in lmer
control_params <- lmerControl(calc.derivs = FALSE, optimizer = "bobyqa", optCtrl = list(maxfun = 2e5))

# LMM for verb-related N400
mod_N400_verb <- lmer_alt(N400_verb ~ style/semantics + (semantics*style||participant) + (semantics*style||item),
                          data = a1, control = control_params)

# LMM for picture-related N400
mod_N400_pict <- lmer_alt(N400_pict ~ style/semantics + (semantics*style||participant) + (semantics*style||item),
                          data = a1, control = control_params)

# Create a list of all four models
models <-  list("N400_VERB" = mod_N400_verb, "N400_PICT" = mod_N400_pict)

# Settings for sensitivity simulation
effect_name <- "stylenor:semantics2"
effect_sizes <- seq(-1.0, 0.1, 0.1)
alpha <- 0.05
n_sim <- 1000
n_cores <- 7 #8

# Set up parallel processing
plan(multisession, workers = n_cores)

# Loop over models (verb/pict)
message("Launching sensitivity simulations")
```

```
## Launching sensitivity simulations
```

```r
tic()
power <- map_dfr(models, function(model) {
```

```
# Extract data from the original model
new_data <- model@frame
dep_var <- all.vars(formula(model))[1]

# Loop over effect sizes of interest
future_map_dfr(effect_sizes, function(effect_size,
                                      model_ = model,
                                      control_params_ = control_params) {

  # Print progress (only gets printed once the worker is done with all effect sizes)
  message(glue("Finished simulating effect size {effect_size} for model {dep_var}."))

  # Generate many simulations (response vectors) with the new effect size
  fixef(model_)[effect_name] <- effect_size
  resp_sims <- simulate(model_, nsim = n_sim)

  # Loop over simulations
  map_dfr(resp_sims, function(resp_sim) {

    # Re-fit the model
    new_data[[dep_var]] <- resp_sim
    is_not_converged <- has_warning(new_model <- update(model_,
                                                        data = new_data,
                                                        control = control_params_))

    # Extract model outputs
    estimate <- round(fixef(new_model)[effect_name], 4)
    p_value <- round(summary(new_model)$coefficients[effect_name, "Pr(>|t|)"], 6)
    is_significant <- p_value < alpha
    is_singular <- any(grepl("singular", new_model@optinfo$conv$lme4$messages))

    # Return model outputs
    data.frame(dep_var,
               effect_name,
               effect_size,
               estimate,
               p_value,
               is_significant,
                  is_singular,
                  is_not_converged)
    })
  },
  .options=furrr_options(seed = 42))
})
```

```
## Finished simulating effect size -1 for model N400_verb.

## Finished simulating effect size -0.9 for model N400_verb.

## Finished simulating effect size -0.8 for model N400_verb.

## Finished simulating effect size -0.7 for model N400_verb.

## Finished simulating effect size -0.6 for model N400_verb.

## Finished simulating effect size -0.5 for model N400_verb.

## Finished simulating effect size -0.4 for model N400_verb.

## Finished simulating effect size -0.3 for model N400_verb.

## Finished simulating effect size -0.2 for model N400_verb.

## Finished simulating effect size -0.1 for model N400_verb.

## Finished simulating effect size 0 for model N400_verb.

## Finished simulating effect size 0.1 for model N400_verb.

## Finished simulating effect size -1 for model N400_pict.

## Finished simulating effect size -0.9 for model N400_pict.
## Finished simulating effect size -0.8 for model N400_pict.
```

```
## Finished simulating effect size -0.7 for model N400_pict.
```

```
## Finished simulating effect size -0.6 for model N400_pict.
```

```
## Finished simulating effect size -0.5 for model N400_pict.
```

```
## Finished simulating effect size -0.4 for model N400_pict.
```

```
## Finished simulating effect size -0.3 for model N400_pict.
```

```
## Finished simulating effect size -0.2 for model N400_pict.
```

```
## Finished simulating effect size -0.1 for model N400_pict.
```

```
## Finished simulating effect size 0 for model N400_pict.
```

```
## Finished simulating effect size 0.1 for model N400_pict.
```

```r
toc()
```

```
## 106954.1 sec elapsed
```

```r
# Stop parallel processing
plan(sequential)

# Summarize across simulations
conf_level <- 1 - alpha
conf_method <- "logit"
power_summary <- power %>%
  group_by(dep_var, effect_name, effect_size) %>%
  summarize(estimate_mean = mean(estimate),
            n_sim = n(),
            n_significant = sum(is_significant),
            perc_singular = mean(is_singular),
            perc_not_converged = mean(is_not_converged),
            .groups = "drop")

# Compute average power incl. binomial confidence interval
power_confint <- with(power_summary,
                      binom::binom.confint(n_significant, n_sim, conf_level, conf_method)) %>%
  select(-c(method, x, n)) %>%
  rename(power_mean = mean, power_lower = lower, power_upper = upper)

# Combine and round
power_summary <- power_summary %>%
  cbind(power_confint) %>%
  mutate(across(.cols = c(estimate_mean, power_mean, power_lower, power_upper,
                          perc_singular, perc_not_converged),
                .fns = ~ round(.x, 4)))
print(power_summary)
```

```
##      dep_var         effect_name effect_size estimate_mean n_sim n_significant
## 1  N400_pict stylenor:semantics2        -1.0       -0.9977  1000           989
## 2  N400_pict stylenor:semantics2        -0.9       -0.9077  1000           961
## 3  N400_pict stylenor:semantics2        -0.8       -0.8115  1000           937
## 4  N400_pict stylenor:semantics2        -0.7       -0.6962  1000           860
## 5  N400_pict stylenor:semantics2        -0.6       -0.5989  1000           710
## 6  N400_pict stylenor:semantics2        -0.5       -0.5015  1000           576
## 7  N400_pict stylenor:semantics2        -0.4       -0.3938  1000           378
## 8  N400_pict stylenor:semantics2        -0.3       -0.2986  1000           233
## 9  N400_pict stylenor:semantics2        -0.2       -0.2070  1000           139
## 10 N400_pict stylenor:semantics2        -0.1       -0.0984  1000            72
## 11 N400_pict stylenor:semantics2         0.0       -0.0111  1000            47
## 12 N400_pict stylenor:semantics2         0.1        0.1000  1000            52
## 13 N400_verb stylenor:semantics2        -1.0       -1.0016  1000          1000
## 14 N400_verb stylenor:semantics2        -0.9       -0.9060  1000          1000
## 15 N400_verb stylenor:semantics2        -0.8       -0.8056  1000           998
## 16 N400_verb stylenor:semantics2        -0.7       -0.7017  1000           995
## 17 N400_verb stylenor:semantics2        -0.6       -0.6046  1000           961
## 18 N400_verb stylenor:semantics2        -0.5       -0.5020  1000           879
## 19 N400_verb stylenor:semantics2        -0.4       -0.3924  1000           679
## 20 N400_verb stylenor:semantics2        -0.3       -0.3022  1000           453
## 21 N400_verb stylenor:semantics2        -0.2       -0.1992  1000           223
## 22 N400_verb stylenor:semantics2        -0.1       -0.0983  1000            89
## 23 N400_verb stylenor:semantics2         0.0       -0.0079  1000            50
## 24 N400_verb stylenor:semantics2         0.1        0.1001  1000            83
##    perc_singular perc_not_converged power_mean power_lower power_upper
## 1              0              0.042      0.989      0.9802      0.9939
## 2              0              0.065      0.961      0.9471      0.9714
## 3              0              0.052      0.937      0.9202      0.9505
## 4              0              0.046      0.860      0.8371      0.8802
## 5              0              0.045      0.710      0.6811      0.7373
## 6              0              0.044      0.576      0.5451      0.6063
## 7              0              0.046      0.378      0.3484      0.4085
## 8              0              0.046      0.233      0.2078      0.2602
## 9              0              0.062      0.139      0.1189      0.1619
## 10             0              0.054      0.072      0.0575      0.0898
## 11             0              0.052      0.047      0.0355      0.0620
## 12             0              0.052      0.052      0.0398      0.0676
## 13             0              0.054      1.000      0.9963      1.0000
## 14             0              0.053      1.000      0.9963      1.0000
## 15             0              0.063      0.998      0.9920      0.9995
## 16             0              0.043      0.995      0.9880      0.9979
## 17             0              0.041      0.961      0.9471      0.9714
## 18             0              0.065      0.879      0.8573      0.8978
## 19             0              0.054      0.679      0.6494      0.7072
## 20             0              0.043      0.453      0.4224      0.4840
## 21             0              0.046      0.223      0.1983      0.2499
## 22             0              0.050      0.089      0.0729      0.1083
## 23             0              0.055      0.050      0.0381      0.0654
## 24             0              0.052      0.083      0.0674      0.1018
```

```
# Save to files
write_csv(power, "EEG/tables/power.csv")
write_csv(power_summary, "EEG/tables/power_summary.csv")
```