F02_mci_style_neu_mixed_models.R

Aristei et al.

2020

```
## MCI_STYLE_NEU MIXED MODELS SCRIPT ##

# Computes linear mixed-effects regression models with simple contrast coding for the fixed effects of semantics and
# narrative context (the context being emotionally neutral). Thus, in each model, the estimate of the intercept is the
# grand mean, while the estimates of the slopes contrast "treatment" levels to their respective reference levels
# (semantics: violation - intuitive, mci - intuitive; narrative context style: fairytale - normal). The maximal random
# effects structure is used with all by-participant and by-item random slopes and random intercepts. Correlations
# between random effects are removed if the model fails to converge with two different numerical optimizers. Planned
# follow-up contrasts are computed for the main effects and the effects of semantics separately within each type of
# narrative context style.

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

# Load packages
library(MASS)          # Version 7.3-51.6
library(lme4)          # Version 1.1-23
library(lmerTest)      # Version 3.1-2
library(afex)          # Version 0.27-2
library(emmeans)       # Version 1.4.8
library(tidyverse)     # Version 1.3.0
library(magrittr)      # Version 1.5

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

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

```r
# 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 ## -----------------------------------------------------------------------------------

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

# LMM for rating 1
mod_valence <- lmer_alt(rating_1 ~ semantics*style + (semantics*style||participant) + (semantics*style||item),
```

```
                        data = a1, control = control_params)

# LMM for rating 2
mod_arousal <- lmer_alt(rating_2 ~ semantics*style + (semantics*style||participant) + (semantics*style||item),
                        data = a1, control = control_params)

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

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

# Create a list of all four models
models <- list("RATING_1" = mod_valence, "RATING_2" = mod_arousal,
               "N400_VERB" = mod_N400_verb, "N400_PICT" = mod_N400_pict)

# F-tests (type III tests)
(tests <- map(models, anova))
```

```
## $RATING_1
## Type III Analysis of Variance Table with Satterthwaite's method
##                   Sum Sq  Mean Sq NumDF   DenDF F value Pr(>F)
## semantics       0.015486 0.007743     2  90.117  0.0562 0.9453
## style           0.246118 0.246118     1  22.151  1.7877 0.1948
## semantics:style 0.061438 0.030719     2 161.930  0.2231 0.8003
##
## $RATING_2
## Type III Analysis of Variance Table with Satterthwaite's method
##                  Sum Sq Mean Sq NumDF  DenDF F value  Pr(>F)
## semantics       0.02041 0.01020     2 78.896  0.0519 0.94945
## style           1.15011 1.15011     1 23.074  5.8503 0.02386 *
## semantics:style 0.08628 0.04314     2 69.605  0.2194 0.80352
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## $N400_VERB
```

```
## Type III Analysis of Variance Table with Satterthwaite's method
##                 Sum Sq Mean Sq NumDF   DenDF F value Pr(>F)
## semantics       49.615  24.808     2 167.039  2.1461 0.1202
## style           12.252  12.252     1  49.762  1.0599 0.3082
## semantics:style 36.095  18.047     2  58.826  1.5613 0.2184
##
## $N400_PICT
## Type III Analysis of Variance Table with Satterthwaite's method
##                  Sum Sq Mean Sq NumDF  DenDF F value   Pr(>F)
## semantics        73.377  36.688     2   75.8  1.8929 0.157685
## style           134.130 134.130     1 6692.5  6.9202 0.008542 **
## semantics:style 185.370  92.685     2   82.2  4.7819 0.010851 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
## PLANNED FOLLOW-UP CONTRASTS ## ------------------------------------------------------------------------------------

# Allow emmeans to compute Satterthwaites p-values
emm_options(lmer.df = "Satterthwaite", lmerTest.limit = Inf)

# Follow-up contrasts for the main effect of semantics
(means.semantics <- map(models,function(x){
  emmeans(x, trt.vs.ctrl ~ semantics, infer = TRUE, adjust = "bonferroni")$contrasts
  }))
```

```
## NOTE: Results may be misleading due to involvement in interactions
## NOTE: Results may be misleading due to involvement in interactions
## NOTE: Results may be misleading due to involvement in interactions
## NOTE: Results may be misleading due to involvement in interactions

## $RATING_1
##  contrast  estimate     SE   df lower.CL upper.CL t.ratio p.value
##  vio - int  0.00694 0.0218 46.5  -0.0436   0.0574  0.318  1.0000
##  mci - int -0.00136 0.0216 46.7  -0.0515   0.0488 -0.063  1.0000
##
## Results are averaged over the levels of: style
## Degrees-of-freedom method: satterthwaite
## Confidence level used: 0.95
```

```
## Conf-level adjustment: bonferroni method for 2 estimates
## P value adjustment: bonferroni method for 2 tests
##
## $RATING_2
##  contrast  estimate     SE   df lower.CL upper.CL t.ratio p.value
##  vio - int  0.00611 0.0227 50.9  -0.0464   0.0586  0.269  1.0000
##  mci - int -0.00323 0.0243 50.9  -0.0594   0.0530 -0.133  1.0000
##
## Results are averaged over the levels of: style
## Degrees-of-freedom method: satterthwaite
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 2 estimates
## P value adjustment: bonferroni method for 2 tests
##
## $N400_VERB
##  contrast  estimate    SE     df lower.CL upper.CL t.ratio p.value
##  vio - int    0.020 0.100 6757.9   -0.205   0.2449  0.199  1.0000
##  mci - int   -0.181 0.105   81.3   -0.421   0.0593 -1.719  0.1787
##
## Results are averaged over the levels of: style
## Degrees-of-freedom method: satterthwaite
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 2 estimates
## P value adjustment: bonferroni method for 2 tests
##
## $N400_PICT
##  contrast  estimate    SE   df lower.CL upper.CL t.ratio p.value
##  vio - int   0.0229 0.192 57.2   -0.420   0.4655  0.119  1.0000
##  mci - int  -0.3524 0.189 40.6   -0.793   0.0884 -1.861  0.1400
##
## Results are averaged over the levels of: style
## Degrees-of-freedom method: satterthwaite
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 2 estimates
## P value adjustment: bonferroni method for 2 tests
```

```
# Follow-up contrasts for the main effect of context style
(means.style <- map(models, function(x){
  emmeans(x, trt.vs.ctrl ~ style, infer = TRUE, adjust = "bonferroni")$contrasts
```

```
  }))
```

```
## NOTE: Results may be misleading due to involvement in interactions
## NOTE: Results may be misleading due to involvement in interactions
## NOTE: Results may be misleading due to involvement in interactions
## NOTE: Results may be misleading due to involvement in interactions
```

```
## $RATING_1
##  contrast  estimate     SE   df lower.CL upper.CL t.ratio p.value
##  ftl - nor  -0.0231 0.0173 22.1  -0.0589   0.0127 -1.337  0.1948
##
## Results are averaged over the levels of: semantics
## Degrees-of-freedom method: satterthwaite
## Confidence level used: 0.95
##
## $RATING_2
##  contrast  estimate    SE   df lower.CL upper.CL t.ratio p.value
##  ftl - nor  -0.0508 0.021 23.1  -0.0942 -0.00736 -2.419  0.0239
##
## Results are averaged over the levels of: semantics
## Degrees-of-freedom method: satterthwaite
## Confidence level used: 0.95
##
## $N400_VERB
##  contrast  estimate     SE   df lower.CL upper.CL t.ratio p.value
##  ftl - nor  -0.0857 0.0832 49.8   -0.253   0.0815 -1.030  0.3082
##
## Results are averaged over the levels of: semantics
## Degrees-of-freedom method: satterthwaite
## Confidence level used: 0.95
##
## $N400_PICT
##  contrast  estimate    SE   df lower.CL upper.CL t.ratio p.value
##  ftl - nor    0.279 0.106 6692    0.071    0.486 2.631   0.0085
##
## Results are averaged over the levels of: semantics
## Degrees-of-freedom method: satterthwaite
## Confidence level used: 0.95
```

```r
# Follow-up contrasts for semantics within each context style
(means.nested <- map(models, function(x){
  emmeans(x, trt.vs.ctrl ~ semantics|style, infer = TRUE, adjust = "bonferroni")$contrasts
  }))
```

```
## $RATING_1
## style = nor:
##  contrast  estimate     SE   df lower.CL upper.CL t.ratio p.value
##  vio - int  0.00222 0.0244 72.8  -0.0536   0.0580  0.091  1.0000
##  mci - int  0.00131 0.0244 72.5  -0.0546   0.0572  0.054  1.0000
##
## style = ftl:
##  contrast  estimate     SE   df lower.CL upper.CL t.ratio p.value
##  vio - int  0.01167 0.0244 72.9  -0.0442   0.0675  0.478  1.0000
##  mci - int -0.00403 0.0244 72.4  -0.0599   0.0518 -0.165  1.0000
##
## Degrees-of-freedom method: satterthwaite
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 2 estimates
## P value adjustment: bonferroni method for 2 tests
##
## $RATING_2
## style = nor:
##  contrast   estimate     SE   df lower.CL upper.CL t.ratio p.value
##  vio - int  0.011525 0.0262 90.0  -0.0482   0.0713  0.440  1.0000
##  mci - int  0.006008 0.0282 77.0  -0.0585   0.0705  0.213  1.0000
##
## style = ftl:
##  contrast   estimate     SE   df lower.CL upper.CL t.ratio p.value
##  vio - int  0.000692 0.0262 90.2  -0.0591   0.0605  0.026  1.0000
##  mci - int -0.012475 0.0282 76.8  -0.0769   0.0520 -0.443  1.0000
##
## Degrees-of-freedom method: satterthwaite
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 2 estimates
## P value adjustment: bonferroni method for 2 tests
##
## $N400_VERB
```

```
## style = nor:
##  contrast  estimate    SE    df lower.CL upper.CL t.ratio p.value
##  vio - int  -0.0241 0.146 294.2   -0.352   0.3041 -0.165  1.0000
##  mci - int  -0.3805 0.156  90.7   -0.736  -0.0251 -2.440  0.0332
##
## style = ftl:
##  contrast  estimate    SE    df lower.CL upper.CL t.ratio p.value
##  vio - int   0.0640 0.146 295.6   -0.265   0.3925  0.439  1.0000
##  mci - int   0.0192 0.156  90.3   -0.336   0.3743  0.123  1.0000
##
## Degrees-of-freedom method: satterthwaite
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 2 estimates
## P value adjustment: bonferroni method for 2 tests
##
## $N400_PICT
## style = nor:
##  contrast  estimate    SE   df lower.CL upper.CL t.ratio p.value
##  vio - int   -0.221 0.240 91.6   -0.769    0.327 -0.920  0.7198
##  mci - int   -0.751 0.230 87.8   -1.275   -0.227 -3.270  0.0031
##
## style = ftl:
##  contrast  estimate    SE   df lower.CL upper.CL t.ratio p.value
##  vio - int    0.267 0.241 91.7   -0.281    0.815  1.110  0.5399
##  mci - int    0.046 0.229 87.5   -0.477    0.569  0.201  1.0000
##
## Degrees-of-freedom method: satterthwaite
## Confidence level used: 0.95
## Conf-level adjustment: bonferroni method for 2 estimates
## P value adjustment: bonferroni method for 2 tests
```

```r
# Backup results
save(models, tests, means.semantics, means.style, means.nested, file = "EEG/export/stats.RData")

# System specs and package versions
sessionInfo()
```

```
## R version 4.0.2 (2020-06-22)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
```

```
## Running under: Windows 10 x64 (build 18362)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=German_Germany.1252  LC_CTYPE=German_Germany.1252    LC_MONETARY=German_Germany.1252
## [4] LC_NUMERIC=C                    LC_TIME=German_Germany.1252
##
## attached base packages:
## [1] stats     graphics  grDevices datasets  utils     methods   base
##
## other attached packages:
##  [1] magrittr_1.5    forcats_0.5.0   stringr_1.4.0   dplyr_1.0.0     purrr_0.3.4     readr_1.3.1
##  [7] tidyr_1.1.0     tibble_3.0.3    ggplot2_3.3.2   tidyverse_1.3.0 emmeans_1.4.8   afex_0.27-2
## [13] lmerTest_3.1-2  lme4_1.1-23     Matrix_1.2-18   MASS_7.3-51.6
##
## loaded via a namespace (and not attached):
##   [1] minqa_1.2.4         colorspace_1.4-1    ellipsis_0.3.1      rio_0.5.16          estimability_1.3
##   [6] fs_1.5.0            rstudioapi_0.11     listenv_0.8.0       R.matlab_3.6.2      fansi_0.4.1
##  [11] mvtnorm_1.1-1       lubridate_1.7.9     xml2_1.3.2          codetools_0.2-16    splines_4.0.2
##  [16] R.methodsS3_1.8.0   knitr_1.29          eegUtils_0.5.0.9000 jsonlite_1.7.0      nloptr_1.2.2.2
##  [21] broom_0.7.0         dbplyr_1.4.4        R.oo_1.23.0         shiny_1.5.0         compiler_4.0.2
##  [26] httr_1.4.2          backports_1.1.8     lazyeval_0.2.2      assertthat_0.2.1    fastmap_1.0.1
##  [31] cli_2.0.2           later_1.1.0.1       htmltools_0.5.0     tools_4.0.2         gtable_0.3.0
##  [36] glue_1.4.1          reshape2_1.4.4      Rcpp_1.0.5          carData_3.0-4       cellranger_1.1.0
##  [41] vctrs_0.3.2         nlme_3.1-148        xfun_0.16           globals_0.12.5      Rmisc_1.5
##  [46] openxlsx_4.1.5      rvest_0.3.6         mime_0.9            miniUI_0.1.1.1      lifecycle_0.2.0
##  [51] renv_0.12.0         statmod_1.4.34      future_1.18.0       scales_1.1.1        hms_0.5.3
##  [56] promises_1.1.1      parallel_4.0.2      yaml_2.2.1          curl_4.3            stringi_1.4.6
##  [61] highr_0.8           boot_1.3-25         zip_2.1.1           matrixStats_0.56.0  rlang_0.4.7
##  [66] pkgconfig_2.0.3     pracma_2.2.9        evaluate_0.14       lattice_0.20-41     htmlwidgets_1.5.1
##  [71] tidyselect_1.1.0    plyr_1.8.6          R6_2.4.1            generics_0.0.2      ini_0.3.1
##  [76] DBI_1.1.0           mgcv_1.8-31         pillar_1.4.6        haven_2.3.1         foreign_0.8-80
##  [81] withr_2.2.0         abind_1.4-5         future.apply_1.6.0  modelr_0.1.8        crayon_1.3.4
##  [86] car_3.0-8           plotly_4.9.2.1      rmarkdown_2.3       grid_4.0.2          readxl_1.3.1
##  [91] data.table_1.13.0   blob_1.2.1          reprex_0.3.0        digest_0.6.25       xtable_1.8-4
##  [96] httpuv_1.5.4        numDeriv_2016.8-1.1 R.utils_2.9.2       signal_0.7-6        munsell_0.5.0
## [101] viridisLite_0.3.0
```