Familiarity Facilitates Feature-based Face Processing Models for Reaction Times

Matteo Visconti di Oleggio Castello, Kelsey G. Wheeler, Carlo Cipolli, M. Ida Gobbini

Contents

Setup

In this document we'll analyze the reaction times creating Linear Mixed-Effect Models separately for Target Present and Target Absent trials.

```
# return version information
version
```

```
##                _
## platform       x86_64-apple-darwin13.4.0
## arch           x86_64
## os             darwin13.4.0
## system         x86_64, darwin13.4.0
## status
## major          3
## minor          2.3
## year           2015
## month          12
## day            10
## svn rev        69752
## language       R
## version.string R version 3.2.3 (2015-12-10)
## nickname       Wooden Christmas-Tree
```

```
packages <- c('lme4',
              'car',
              'dplyr')

for (package in packages) {
  require(package, character.only=T)
  cat(paste(package, packageVersion(package), '\n'))
}
```

```
## lme4 1.1.11
## car 2.1.1
## dplyr 0.4.3
```

```
data <- read.csv('data.csv')
# set order of levels for plotting
data$orientation <- factor(data$orientation,
                           levels=c('Upright', 'Inverted'))
data$target_presence <- factor(data$target_presence,
                               levels=c('Target Present', 'Target Absent'))
# get correct trials
data_correct <- data %>% filter(correct == 1)

# set set_size as a factor
data_correct$set_size <- as.factor(data_correct$set_size)
```

Set up zero-sum contrasts for factors.

```
contrasts(data_correct$set_size) <- contr.poly(3)
contrasts(data_correct$orientation) <- c(-1, 1)
contrasts(data_correct$familiarity) <- c(-1,1)
contrasts(data_correct$target_sex) <- c(-1,1)
```

Model on Target Present Trials

```
# get target present trials
data_correct_tp <- data_correct %>%
    filter(target_presence == 'Target Present')
```

Try to fit a very general model:

```
m1 <- lmer(log(RT) ~ set_size*familiarity*orientation + target_sex +
             (1 + target_sex | subid) +
             (1 + target_sex | stimuli_combination),
           REML=F, data=data_correct_tp)
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, : Model is nearly unide
##  - Rescale variables?
```

Model `m1` fails to converge, being very complex. We remove the random slope for `target_sex` first from the random effects for `stimuli_combination`.

```
m2 <- lmer(log(RT) ~ set_size*familiarity*orientation + target_sex +
               (1 + target_sex | subid) +
               (1 | stimuli_combination),
           REML=F, data=data_correct_tp)
```

Now reduce the complexity of the random effects structure and test it.

```
m3 <- lmer(log(RT) ~ set_size*familiarity*orientation + target_sex +
               (1 + target_sex | subid),
           REML=F, data=data_correct_tp)
anova(m3, m2)
```

```
## Data: data_correct_tp
## Models:
## m3: log(RT) ~ set_size * familiarity * orientation + target_sex +
## m3:     (1 + target_sex | subid)
## m2: log(RT) ~ set_size * familiarity * orientation + target_sex +
## m2:     (1 + target_sex | subid) + (1 | stimuli_combination)
##    Df    AIC    BIC  logLik deviance  Chisq Chi Df Pr(>Chisq)
## m3 17 4159.8 4279.6 -2062.9   4125.8
## m2 18 4007.0 4133.9 -1985.5   3971.0 154.72      1  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Model `m2` is better, thus we'll keep the random effect for `stimuli_combination`. Let's remove the random slope for `subid` now.

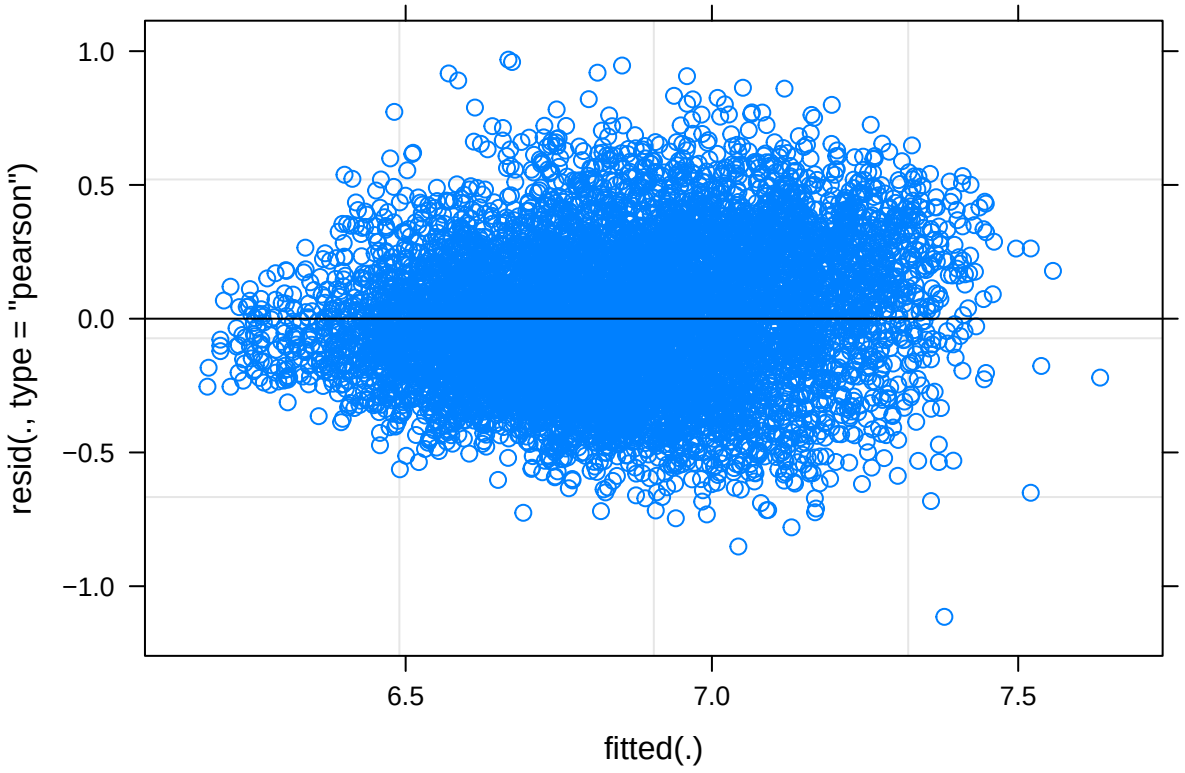
```
m4 <- lmer(log(RT) ~ set_size*familiarity*orientation + target_sex +
              (1 | subid) + (1 | stimuli_combination),
           REML=F, data=data_correct_tp)
anova(m4, m2)
```

```
## Data: data_correct_tp
## Models:
## m4: log(RT) ~ set_size * familiarity * orientation + target_sex +
## m4:     (1 | subid) + (1 | stimuli_combination)
## m2: log(RT) ~ set_size * familiarity * orientation + target_sex +
## m2:     (1 + target_sex | subid) + (1 | stimuli_combination)
##    Df    AIC    BIC  logLik deviance  Chisq Chi Df Pr(>Chisq)
## m4 16 4370.7 4483.5 -2169.4   4338.7
## m2 18 4007.0 4133.9 -1985.5   3971.0 367.71      2  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

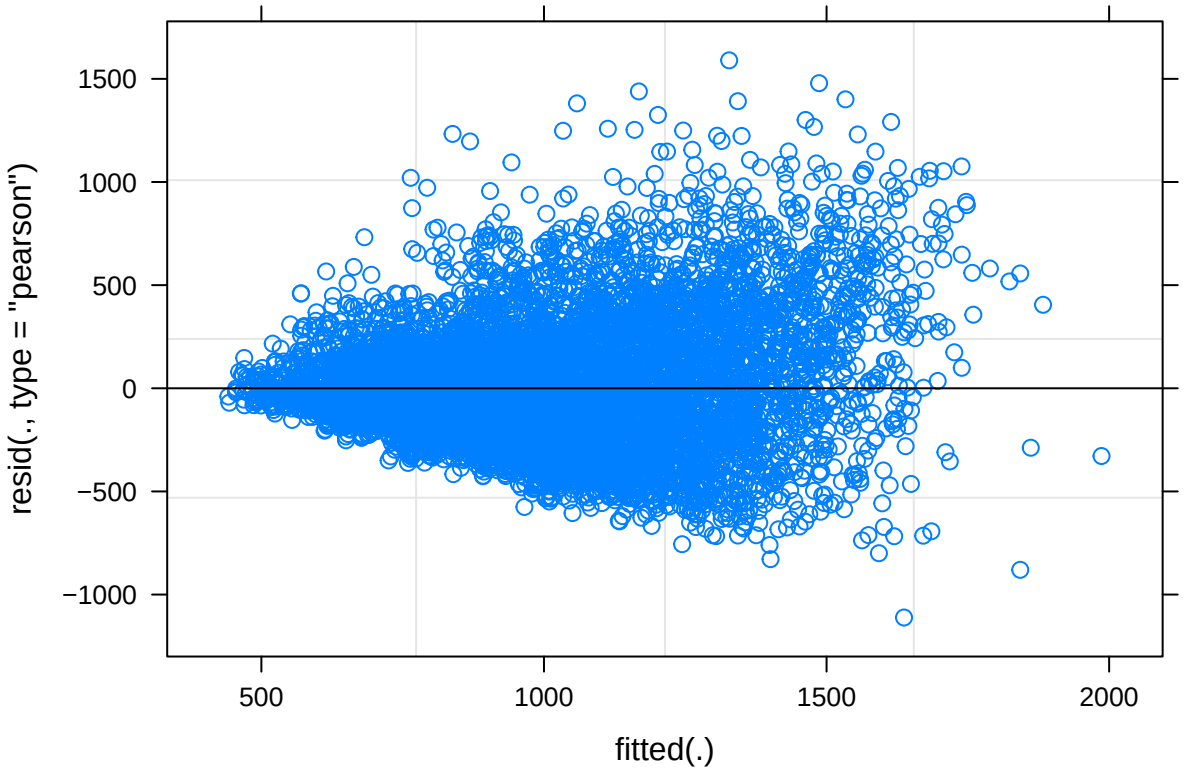
Model `m2` is still better, thus we'll keep its random effects structure. We'll refit the model with Restricted Maximum Likelihood (REML), and check the fitted vs. residual plot with and without log-transformation.

```
m2_reml <- update(m2, REML=T)
m2_reml_nolog <- update(m2_reml, RT ~ .)

plot(m2_reml)
```




```
plot(m2_reml_nolog)
```



The log-transformation seems to work better than no transformation. Now we'll test significance of the factors using Type 3 Analysis of Deviance with Wald's χ^2 test.

```
Anova(m2_reml, type=3)
```

```
## Analysis of Deviance Table (Type III Wald chisquare tests)
##
## Response: log(RT)
##                                      Chisq Df Pr(>Chisq)
## (Intercept)                       83358.3987  1  < 2.2e-16 ***
## set_size                           1318.9264  2  < 2.2e-16 ***
## familiarity                         169.6050  1  < 2.2e-16 ***
## orientation                         400.4934  1  < 2.2e-16 ***
## target_sex                            7.9642  1   0.004771 **
## set_size:familiarity                  8.5886  2   0.013646 *
## set_size:orientation                  2.7510  2   0.252709
## familiarity:orientation               9.1612  1   0.002472 **
## set_size:familiarity:orientation     11.1657  2   0.003762 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Model on Target Absent Trials

```
# get target absent trials
data_correct_ta <- data_correct %>%
    filter(target_presence == 'Target Absent')
```

Try to fit a very general model:

```
m1 <- lmer(log(RT) ~ set_size*familiarity*orientation + target_sex +
             (1 + target_sex | subid) +
             (1 + target_sex | stimuli_combination),
           REML=F, data=data_correct_ta)
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control
## $checkConv, : unable to evaluate scaled gradient
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control
## $checkConv, : Model failed to converge: degenerate Hessian with 1 negative
## eigenvalues
```

Also in this case the general model `m1` fails. We'll thus follow the same steps as with target present trials. We remove the random slope for `target_sex` first from the random effects for `stimuli_combination`.

```
m2 <- lmer(log(RT) ~ set_size*familiarity*orientation + target_sex +
               (1 + target_sex | subid) +
               (1 | stimuli_combination),
           REML=F, data=data_correct_ta)
```

Now we reduce the complexity of the random effects structure and test it.

```
m3 <- lmer(log(RT) ~ set_size*familiarity*orientation + target_sex +
               (1 + target_sex | subid),
           REML=F, data=data_correct_ta)
anova(m3, m2)
```

```
## Data: data_correct_ta
## Models:
## m3: log(RT) ~ set_size * familiarity * orientation + target_sex +
## m3:     (1 + target_sex | subid)
## m2: log(RT) ~ set_size * familiarity * orientation + target_sex +
## m2:     (1 + target_sex | subid) + (1 | stimuli_combination)
##    Df     AIC     BIC logLik deviance  Chisq Chi Df Pr(>Chisq)
## m3 17 -1345.0 -1224.5 689.50  -1379.0
## m2 18 -1402.7 -1275.1 719.37  -1438.7 59.739      1  1.083e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Model `m2` is better, thus we'll keep the random effect for `stimuli_combination`. Let's remove the random slope for `subid` now.

```
m4 <- lmer(log(RT) ~ set_size*familiarity*orientation + target_sex +
               (1 | subid) + (1 | stimuli_combination),
           REML=F, data=data_correct_ta)
anova(m4, m2)
```

```
## Data: data_correct_ta
## Models:
## m4: log(RT) ~ set_size * familiarity * orientation + target_sex +
## m4:     (1 | subid) + (1 | stimuli_combination)
## m2: log(RT) ~ set_size * familiarity * orientation + target_sex +
## m2:     (1 + target_sex | subid) + (1 | stimuli_combination)
##    Df      AIC      BIC logLik deviance  Chisq Chi Df Pr(>Chisq)
## m4 16  -404.11  -290.69 218.05  -436.11
## m2 18 -1402.73 -1275.14 719.37 -1438.73 1002.6      2  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Model `m2` is still better, thus we'll keep its random effects structure. The model thus has the same random effects structure as in the target present trials. We'll refit the model with Restricted Maximum Likelihood (REML), and test significance of the factors using Type 3 Analysis of Deviance with Wald's χ^2 test.

```
m2_reml <- update(m2, REML=T)

Anova(m2_reml, type=3)
```

```
## Analysis of Deviance Table (Type III Wald chisquare tests)
##
## Response: log(RT)
##                                    Chisq Df Pr(>Chisq)
## (Intercept)                    95809.3559  1  < 2.2e-16 ***
## set_size                        8131.3895  2  < 2.2e-16 ***
```

```
## familiarity                         414.3051  1  < 2.2e-16 ***
## orientation                         792.6386  1  < 2.2e-16 ***
## target_sex                            4.9866  1   0.025545 *
## set_size:familiarity                  6.5868  2   0.037127 *
## set_size:orientation                  6.2015  2   0.045016 *
## familiarity:orientation               6.7476  1   0.009387 **
## set_size:familiarity:orientation      0.5648  2   0.753976
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```