m21 LDT ERP analysis N250

Joanna Morris

2025-06-12

Contents

1 Set parameters

Set chunk parameters

Load libraries

```
library(performance)
library(effectsize)
library(afex)
library(emmeans)
library(gridExtra)
library(kableExtra)
library(pander)
library(tidyverse)
```

Set ggplot parameters

```
theme_set(theme_classic() +
            theme(legend.position = "bottom",
                  axis.text=element_text(size=10),
                  axis.title=element_text(size=9)))

# Define a custom color palette
my_palette <- c("#A6CEE3",  "#FB9A99")
my_palette_2 <- c( "#1F78B4","#E31A1C" )
my_palette_3 <- c("#A6CEE3","#1F78B4","#FB9A99","#E31A1C")


# Create a function to apply this palette
scale_color_custom <- function() {
  scale_color_manual(values = my_palette_2)
```

```
}

scale_fill_custom <- function() {
  scale_fill_manual(values = my_palette_2)
}
```

Define standard error of the mean function

```
sem <- function(x) sd(x)/sqrt(length(x))
```

2 Load and format data files

Now we extract `SubjID` from the `ERPset` column

We then join the ERP data, and language into a single data frame

Divide into word, non-word and difference wave dataframes

```
n250_1_words <- n250_1 |> filter(bini %in% c(1:2))
n250_1_nonwords <- n250_1 |> filter(bini %in% c(3:6))
n250_1_diff <- n250_1 |> filter(bini %in% c(9:11))

n250_2_words <- n250_2 |> filter(bini %in% c(1:2))
n250_2_nonwords <- n250_2 |> filter(bini %in% c(3:6))
n250_2_diff <- n250_2 |> filter(bini %in% c(9:11))
```

Then we do some more formatting and cleanup of the dataframes.We create separate columns, one for each independent variable (anteriority, laterality, morphological family size). To do this we have to use `seperate` function from the `stringr` package. Run `vignette("programming", package = "dplyr")` to see more about `tidy-selection` and `tidy-evaluation`.

Now we need to extract just the bins and channels that we intend to analyse. For this analysis we will use 9 channels: F3, Fz, F4, C3, Cz, C4, P3, Pz, P4 . We will use the`mutate` function from the `dplyr` package along with the `case_when` function. The `case_when` function is a sequence of two-sided formulas. The left hand side determines which values match this case. The right hand side provides the replacement value.

3 Compute the ANOVA

A linear mixed-effects modeling approach was used to analyze ERP amplitudes in the N250 time window. This method offers several advantages over traditional repeated-measures ANOVA. First, it allows for accurate estimation of fixed effects by explicitly modeling nuisance variables such as Laterality and Anteriority, which account for a substantial proportion of the variance in ERP amplitude. Including these topographic covariates in the model improves the precision of estimates for the linguistic variables of interest. Second, mixed-effects models provide a more appropriate treatment of within-subject dependencies by estimating random effects for participants, thereby avoiding the sphericity assumptions required by repeated-measures ANOVA and allowing for unbalanced data without biasing results. Third, the use of planned contrasts and post hoc comparisons with Bonferroni correction helps control the family-wise error rate, reducing the risk of Type I error across multiple tests. Finally, by treating subjects as random effects, the model supports broader generalization of the findings beyond the specific sample tested. Together, these features make linear mixed-effects models well suited for analyzing ERP data with complex factorial designs and repeated observations.

3.1 Group 1

3.1.1 ANOVA Model

```
# Fit the ANOVA/mixed model
anova_model_1 <- mixed(
  value ~ lang_type_ortho * family_size * complexity +
    laterality * anteriority  +  # Nuisance variables
    (1 | SubjID),
  data = n250_1_nonwords,
  method = "KR"  # Kenward-Roger approximation for accurate F-tests
)

# Print ANOVA results
anova_model_1
```

```
|| Mixed Model Anova Table (Type 3 tests, KR-method)
||
|| Model: value ~ lang_type_ortho * family_size * complexity + laterality *
|| Model:     anteriority + (1 | SubjID)
|| Data: n250_1_nonwords
||                               Effect      df         F p.value
|| 1                    lang_type_ortho   1, 59      0.13    .722
|| 2                        family_size 1, 2121      0.49    .482
|| 3                         complexity 1, 2121      0.08    .775
|| 4                         laterality 2, 2121      0.50    .606
|| 5                        anteriority 2, 2121 36.58 ***   <.001
|| 6        lang_type_ortho:family_size 1, 2121    4.52 *    .034
|| 7         lang_type_ortho:complexity 1, 2121    6.14 *    .013
|| 8             family_size:complexity 1, 2121    2.99 +    .084
```

```
|| 9                 laterality:anteriority 4, 2121      0.78    .540
|| 10 lang_type_ortho:family_size:complexity 1, 2121      0.14    .708
|| ---
|| Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '+' 0.1 ' ' 1
```

3.1.2 Partial Eta Squared

Compute Partial Eta Squared (η_p^2) for F-tests. This gives η_p^2 values for each effect. Then, compute R^2 for the Mixed Model. This provides marginal R^2 (fixed effects only) and conditional R^2 (fixed + random effects).

```
# Extract effect sizes from your ANOVA model
eta_squared(anova_model_1, partial = TRUE)
```

```
|| # Effect Size for ANOVA (Type III)
||
|| Parameter                              | Eta2 (partial) |       95% CI
|| ----------------------------------------------------------------------
|| lang_type_ortho                        |       2.16e-03 | [0.00, 1.00]
|| family_size                            |       2.33e-04 | [0.00, 1.00]
|| complexity                             |       3.86e-05 | [0.00, 1.00]
|| laterality                             |       4.72e-04 | [0.00, 1.00]
|| anteriority                            |           0.03 | [0.02, 1.00]
|| lang_type_ortho:family_size            |       2.13e-03 | [0.00, 1.00]
|| lang_type_ortho:complexity             |       2.89e-03 | [0.00, 1.00]
|| family_size:complexity                 |       1.41e-03 | [0.00, 1.00]
|| laterality:anteriority                 |       1.46e-03 | [0.00, 1.00]
|| lang_type_ortho:family_size:complexity |       6.62e-05 | [0.00, 1.00]
||
|| - One-sided CIs: upper bound fixed at [1.00].
```

```
# Compute Marginal and Conditional R²
r2(anova_model_1)
```

```
|| # R2 for Mixed Models
||
||   Conditional R2: 0.421
||      Marginal R2: 0.025
```

3.1.3 Main Findings

Effect	df	F	p	eta-sqrd
lang_type_ortho x family_size	(1, 2121)	4.52 *	.034	2.13e-03
lang_type_ortho x complexity	(1, 2121)	6.14 *	.013	2.89e-03

3.1.3.1 lang_type_ortho × family_size (Interaction)

```
pairs <- emmeans(anova_model_1, pairwise ~ lang_type_ortho * family_size, adjust = "bonferroni", pbkrtest.limit = 6480)
(pairs_df <- as.data.frame(pairs$contrasts))
```

3.1.3.1.1 Custom contrasts for lang_type_ortho × family_size (Interaction)

```
||  contrast                  estimate        SE      df t.ratio p.value
||  High large - Low large   0.3737003 0.4899145   63.69   0.763  1.0000
||  High large - High small  0.2687520 0.1396884 2121.00   1.924  0.3270
||  High large - Low small   0.2384747 0.4899145   63.69   0.487  1.0000
||  Low large - High small  -0.1049483 0.4899145   63.69  -0.214  1.0000
||  Low large - Low small   -0.1352256 0.1286715 2121.00  -1.051  1.0000
||  High small - Low small  -0.0302773 0.4899145   63.69  -0.062  1.0000
||
|| Results are averaged over the levels of: complexity, laterality, anteriority
|| Degrees-of-freedom method: kenward-roger
|| P value adjustment: bonferroni method for 6 tests
```

```
selected_contrasts_famsize <- pairs$contrasts[pairs_df$contrast %in% c("High large - High small",
                                                                "Low large - Low small"),]
selected_contrasts_orthtyp <- pairs$contrasts[pairs_df$contrast %in% c("High small - Low small",
                                                                "High large - Low large"), ]
selected_contrasts_df_famsize <- as.data.frame(selected_contrasts_famsize)  # Convert the emmGrid object to a dataframe
selected_contrasts_df_orthtyp <- as.data.frame(selected_contrasts_orthtyp)  # Convert the emmGrid object to a dataframe
cohensd_small <- as.data.frame(cohens_d(value ~ lang_type_ortho,
         data = subset(n250_1_nonwords, family_size == "small")))
cohensd_large <- as.data.frame(cohens_d(value ~ lang_type_ortho,
         data = subset(n250_1_nonwords, family_size == "large")))
cohensd_hi_ortho <- as.data.frame(cohens_d(value ~ family_size,
         data = subset(n250_1_nonwords, lang_type_ortho == "High")))
cohensd_lo_ortho <- as.data.frame(cohens_d(value ~ family_size,
         data = subset(n250_1_nonwords, lang_type_ortho == "Low")))
```

```r
cohensd_orthtyp <- bind_rows(large = cohensd_large,
                             small = cohensd_small,
                             .id = "famsize")

cohensd_famsize <- bind_rows(hi_ortho = cohensd_hi_ortho,
                             lo_ortho = cohensd_lo_ortho,
                             .id = "orthtyp")

(orthtyp_contrasts_df <- bind_cols(selected_contrasts_df_orthtyp,cohensd_orthtyp))
```

```
||  contrast                 estimate        SE    df t.ratio p.value famsize
||  High large - Low large  0.3737003 0.4899145 63.69   0.763  0.8968 large
||  High small - Low small -0.0302773 0.4899145 63.69  -0.062  1.0000 small
||     Cohens_d   CI      CI_low   CI_high
||   0.13036450 0.95  0.01151218 0.2491575
||  -0.01040422 0.95 -0.12909993 0.1082962
||
|| Results are averaged over the levels of: complexity, laterality, anteriority
|| Degrees-of-freedom method: kenward-roger
|| P value adjustment: bonferroni method for 2 tests
```

```r
(famsize_contrasts_df <- bind_cols(selected_contrasts_df_famsize,cohensd_famsize))
```

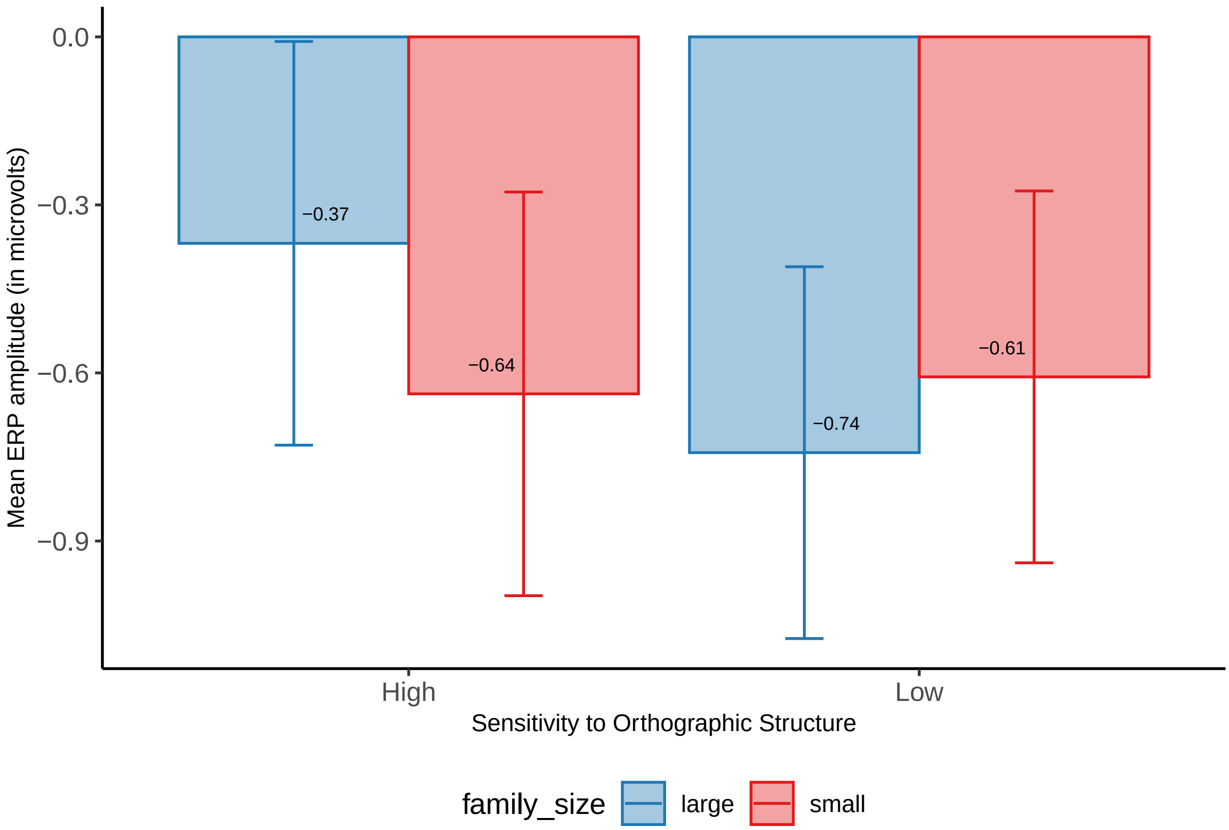
```
||  contrast                  estimate        SE   df t.ratio p.value orthtyp
||  High large - High small  0.2687520 0.1396884 2121   1.924  0.1090 hi_ortho
||  Low large - Low small   -0.1352256 0.1286715 2121  -1.051  0.5868 lo_ortho
||     Cohens_d   CI      CI_low    CI_high
||   0.08927024 0.95 -0.03427963 0.21277580
||  -0.04863125 0.95 -0.16236648 0.06512448
||
|| Results are averaged over the levels of: complexity, laterality, anteriority
|| Degrees-of-freedom method: kenward-roger
|| P value adjustment: bonferroni method for 2 tests
```

```r
(langtyp.famsize_means <- as.data.frame(pairs$emmeans))
```

```
||  lang_type_ortho family_size     emmean        SE    df  lower.CL   upper.CL
||  High            large       -0.3685893 0.3603399 63.69 -1.088518  0.3513391
||  Low             large       -0.7422896 0.3319207 63.69 -1.405439 -0.0791402
||  High            small       -0.6373413 0.3603399 63.69 -1.357270  0.0825871
||  Low             small       -0.6070640 0.3319207 63.69 -1.270213  0.0560854
||
|| Results are averaged over the levels of: complexity, laterality, anteriority
|| Degrees-of-freedom method: kenward-roger
|| Confidence level used: 0.95
```

3.1.3.2 Plots for lang_type_ortho × family_size (Interaction))

Plots for lang_type_ortho × family_size (Interaction)



APA-style Write-up

A significant interaction between orthographic sensitivity and morphological family size was observed, $F(1, 2121) = 4.52$, $p = .034$, despite no individual pairwise contrast reaching significance after Bonferroni correction. To better understand this effect, estimated marginal means were examined. The pattern of means revealed that in the High orthographic sensitivity condition, ERP amplitudes were numerically lower for small compared to large family size, whereas in the Low orthographic sensitivity condition, the opposite pattern was observed. Although neither of these differences was statistically significant on their own ($ps > .10$), the interaction reflects a crossover pattern in which the effect of family size differs depending on orthographic sensitivity This crossover interaction is visualized in Figure X, where the slope of ERP amplitude across family size conditions differs between high and low orthographic sensitivity

3.1.3.3 lang_type_ortho × complexity (Interaction)

```
pairs <- emmeans(anova_model_1, pairwise ~ lang_type_ortho * complexity, adjust = "bonferroni", pbkrtest.limit = 6480)
(pairs_df <- as.data.frame(pairs$contrasts))
```

3.1.3.3.1 Custom contrasts for lang_type_ortho × complexity (Interaction)

```
||  contrast                      estimate        SE      df t.ratio p.value
||  High complex - Low complex -0.0636313 0.4899145   63.69  -0.130  1.0000
||  High complex - High simple -0.2081806 0.1396884 2121.00  -1.490  0.8177
||  High complex - Low simple   0.1988737 0.4899145   63.69   0.406  1.0000
||  Low complex - High simple  -0.1445492 0.4899145   63.69  -0.295  1.0000
||  Low complex - Low simple    0.2625051 0.1286715 2121.00   2.040  0.2488
||  High simple - Low simple    0.4070543 0.4899145   63.69   0.831  1.0000
||
|| Results are averaged over the levels of: family_size, laterality, anteriority
|| Degrees-of-freedom method: kenward-roger
|| P value adjustment: bonferroni method for 6 tests
```

```
selected_contrasts_cmplxty <- pairs$contrasts[pairs_df$contrast %in% c("High complex - High simple",
                                                               "Low complex - Low simple"),]
selected_contrasts_orthtyp <- pairs$contrasts[pairs_df$contrast %in% c("High complex - Low complex",
                                                                       "High simple - Low simple"), ]

selected_contrasts_df_cmplxty <- as.data.frame(selected_contrasts_cmplxty)  # Convert the emmGrid object to a dataframe
selected_contrasts_df_orthtyp <- as.data.frame(selected_contrasts_orthtyp)

cohensd_complex <- as.data.frame(cohens_d(value ~ lang_type_ortho,
         data = subset(n250_1_nonwords, complexity == "complex")))
cohensd_simple <- as.data.frame(cohens_d(value ~ lang_type_ortho,
         data = subset(n250_1_nonwords, complexity == "simple")))
```

```r
cohensd_hi_ortho <- as.data.frame(cohens_d(value ~ complexity,
         data = subset(n250_1_nonwords, lang_type_ortho == "High")))
cohensd_lo_ortho <- as.data.frame(cohens_d(value ~ complexity,
         data = subset(n250_1_nonwords, lang_type_ortho == "Low")))

cohensd_orthtyp <- bind_rows(complex = cohensd_complex,
                             simple = cohensd_simple,
                             .id = "cmplxty")

cohensd_cmplxty <- bind_rows(hi_ortho = cohensd_hi_ortho,
                             lo_ortho = cohensd_lo_ortho,
                             .id = "orthtyp")

(orthtyp_contrasts_df <- bind_cols(selected_contrasts_df_orthtyp,cohensd_orthtyp))
```

```
||  contrast                     estimate        SE    df t.ratio p.value cmplxty
||  High complex - Low complex -0.0636313 0.4899145 63.69  -0.130  1.0000 complex
||  High simple - Low simple    0.4070543 0.4899145 63.69   0.831  0.8183 simple
||    Cohens_d   CI      CI_low    CI_high
||  -0.0221697 0.95 -0.14086556 0.09653626
||   0.1401066 0.95  0.02123269 0.25891684
||
|| Results are averaged over the levels of: family_size, laterality, anteriority
|| Degrees-of-freedom method: kenward-roger
|| P value adjustment: bonferroni method for 2 tests
```

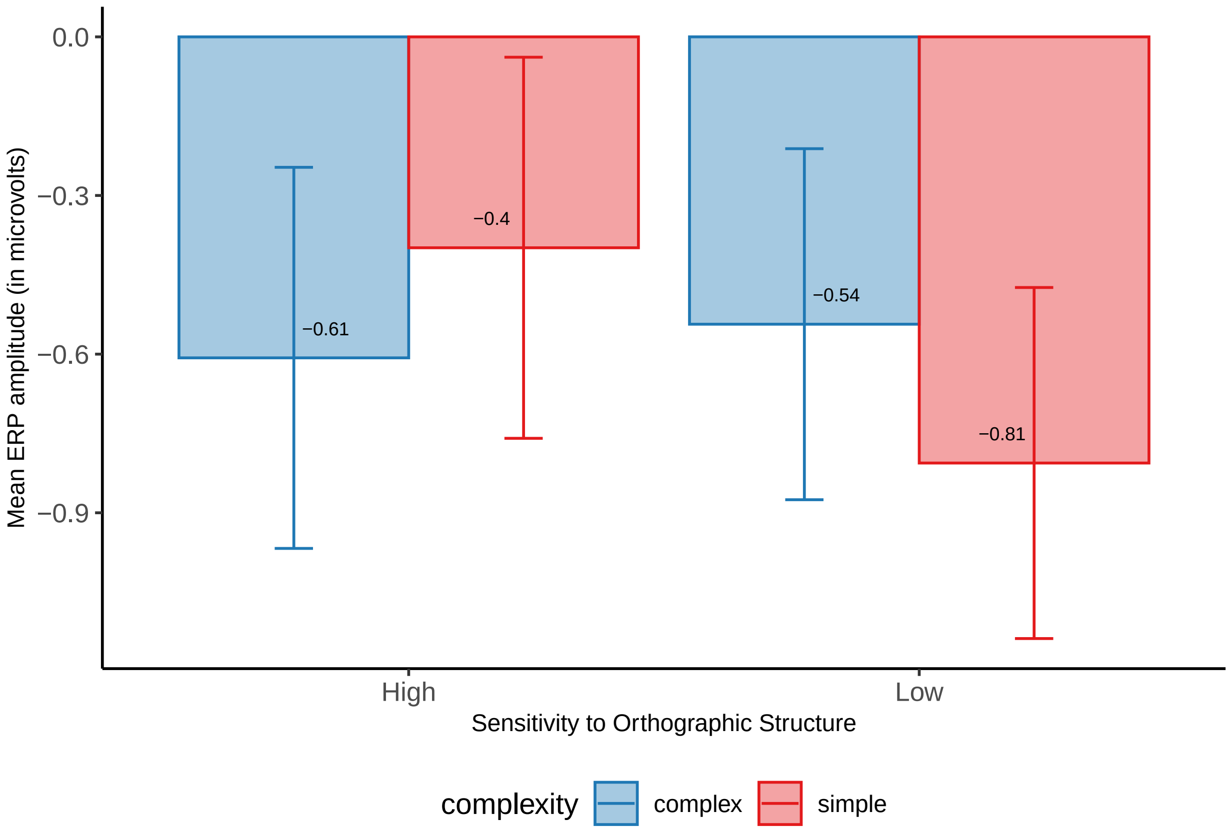
```r
(cmplxty_contrasts_df <- bind_cols(selected_contrasts_df_cmplxty,cohensd_cmplxty))
```

```
||  contrast                     estimate        SE   df t.ratio p.value orthtyp
||  High complex - High simple -0.2081806 0.1396884 2121  -1.490  0.2726 hi_ortho
||  Low complex - Low simple    0.2625051 0.1286715 2121   2.040  0.0829 lo_ortho
||     Cohens_d   CI      CI_low    CI_high
||  -0.06912289 0.95 -0.19260878 0.05439733
||   0.09448232 0.95 -0.01932975 0.20825460
||
|| Results are averaged over the levels of: family_size, laterality, anteriority
|| Degrees-of-freedom method: kenward-roger
|| P value adjustment: bonferroni method for 2 tests
```

```r
(langtyp.cmplxty_means <- as.data.frame(pairs$emmeans))
```

```
||  lang_type_ortho complexity     emmean        SE    df  lower.CL   upper.CL
||  High            complex    -0.6070556 0.3603399 63.69 -1.326984  0.1128729
||  Low             complex    -0.5434242 0.3319207 63.69 -1.206574  0.1197251
||  High            simple     -0.3988750 0.3603399 63.69 -1.118803  0.3210534
||  Low             simple     -0.8059293 0.3319207 63.69 -1.469079 -0.1427799
||
|| Results are averaged over the levels of: family_size, laterality, anteriority
|| Degrees-of-freedom method: kenward-roger
|| Confidence level used: 0.95
```

Plots for lang_type_ortho $\times$ complexity (Interaction)



APA Write-up

A significant interaction between orthographic transparency and morphological complexity was observed, $F(1, 2121) = 6.14$, $p = .013$. Although no individual pairwise comparisons between levels of orthographic sensitivity and complexity were statistically significant after Bonferroni correction ($ps > .08$), the pattern of estimated marginal means revealed a crossover interaction. Specifically, in the high orthographic sensitivity condition, ERP amplitudes were slightly more negative for complex than for simple items, whereas in the Low orthographic sensitivity condition, this pattern was reversed. This non-parallel pattern of effects accounts for the significant interaction and suggests that the influence of morphological complexity on ERP responses depends on participants sensitivity to orthographic structure.

3.2 Cohort 2

3.2.1 ANOVA Model

```
# Fit the ANOVA/mixed model
anova_model_2 <- mixed(
  value ~ lang_type_ortho * family_size * complexity +
    laterality * anteriority  +  # Nuisance variables
    (1 | SubjID),
  data = n250_2_nonwords,
  method = "KR"  # Kenward-Roger approximation for accurate F-tests
)

# Print ANOVA results
anova_model_2
```

```
|| Mixed Model Anova Table (Type 3 tests, KR-method)
||
|| Model: value ~ lang_type_ortho * family_size * complexity + laterality *
|| Model:     anteriority + (1 | SubjID)
|| Data: n250_2_nonwords
||                                    Effect      df         F p.value
|| 1                         lang_type_ortho   1, 39      0.49    .489
|| 2                             family_size 1, 1421      0.01    .907
|| 3                              complexity 1, 1421 12.14 ***   <.001
|| 4                              laterality 2, 1421    3.20 *    .041
|| 5                             anteriority 2, 1421      2.20    .111
|| 6             lang_type_ortho:family_size 1, 1421      2.53    .112
|| 7              lang_type_ortho:complexity 1, 1421    4.03 *    .045
|| 8                  family_size:complexity 1, 1421 17.60 ***   <.001
|| 9                  laterality:anteriority 4, 1421      1.39    .236
```

```
|| 10 lang_type_ortho:family_size:complexity 1, 1421      2.70    .101
|| ---
|| Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '+' 0.1 ' ' 1
```

3.2.2 Compute Partial Eta Squared

Compute Partial Eta Squared (η_p^2) for F-tests. This gives η_p^2 values for each effect. Then, compute R^2 for the Mixed Model. This provides marginal R^2 (fixed effects only) and conditional R^2 (fixed + random effects).

```
# Extract effect sizes from your ANOVA model
eta_squared(anova_model_2, partial = TRUE)
```

```
|| # Effect Size for ANOVA (Type III)
||
|| Parameter                                | Eta2 (partial) |       95% CI
|| -------------------------------------------------------------------------
|| lang_type_ortho                          |           0.01 | [0.00, 1.00]
|| family_size                              |       9.63e-06 | [0.00, 1.00]
|| complexity                               |       8.47e-03 | [0.00, 1.00]
|| laterality                               |       4.49e-03 | [0.00, 1.00]
|| anteriority                              |       3.09e-03 | [0.00, 1.00]
|| lang_type_ortho:family_size              |       1.78e-03 | [0.00, 1.00]
|| lang_type_ortho:complexity               |       2.83e-03 | [0.00, 1.00]
|| family_size:complexity                   |           0.01 | [0.00, 1.00]
|| laterality:anteriority                   |       3.89e-03 | [0.00, 1.00]
|| lang_type_ortho:family_size:complexity   |       1.90e-03 | [0.00, 1.00]
||
|| - One-sided CIs: upper bound fixed at [1.00].
```

```
# Compute Marginal and Conditional R²
r2(anova_model_2)
```

```
|| # R2 for Mixed Models
||
||   Conditional R2: 0.547
||      Marginal R2: 0.024
```

3.2.3 Main Findings

Effect	df	F	p	eta-sqrd
complexity	(1, 1421)	12.14 ***	<.001	8.47e-03
lang_type_ortho x complexity	(1, 1421)	4.03 *	.045	2.83e-03
family_size x complexity	(1, 1421)	17.60 ***	<.001	0.01

Main effect of Complexity; Simple words are more negative than complex words from 200-300 ms.

3.2.3.1 Complexity (Main Effect)

```
pairs_2 <- emmeans(anova_model_2, pairwise ~ complexity, adjust = "bonferroni", pbkrtest.limit = 6480)
(pairs_df_2 <- as.data.frame(pairs_2$contrasts))
```

3.2.3.1.1 Cohen's d for Complexity (Main Effect)

```
||  contrast          estimate        SE   df t.ratio p.value
||  complex - simple 0.4365086 0.1252735 1421   3.484  0.0005
||
|| Results are averaged over the levels of: lang_type_ortho, family_size, laterality, anteriority
|| Degrees-of-freedom method: kenward-roger
```

```
cohensd_2 <- as.data.frame(cohens_d(value ~ complexity, data = n250_2_nonwords))
(complexity_contrasts_df_2 <- bind_cols(pairs_df_2,cohensd_2))
```

```
||  contrast          estimate        SE   df t.ratio p.value  Cohens_d   CI
||  complex - simple 0.4365086 0.1252735 1421   3.484  0.0005 0.1060036 0.95
||       CI_low   CI_high
||  0.005105554 0.2068666
||
|| Results are averaged over the levels of: lang_type_ortho, family_size, laterality, anteriority
|| Degrees-of-freedom method: kenward-roger
```

```
(complexity_means_2 <- as.data.frame(pairs_2$emmeans))
```

```
||  complexity   emmean        SE    df lower.CL upper.CL
||  complex    3.789251 0.4138776 40.85 2.953314 4.625187
||  simple     3.352742 0.4138776 40.85 2.516806 4.188679
||
|| Results are averaged over the levels of: lang_type_ortho, family_size, laterality, anteriority
```

```
|| Degrees-of-freedom method: kenward-roger
|| Confidence level used: 0.95
```

3.2.3.1.2 Mixed Model Comparison: Complexity

To test the effect of `Complexity`, we compare:

- Full model (anova_model_2) → includes `Complexity` and all interactions.
- Reduced model (reduced_model) → removes `Complexity` and all interactions involving `Complexity`

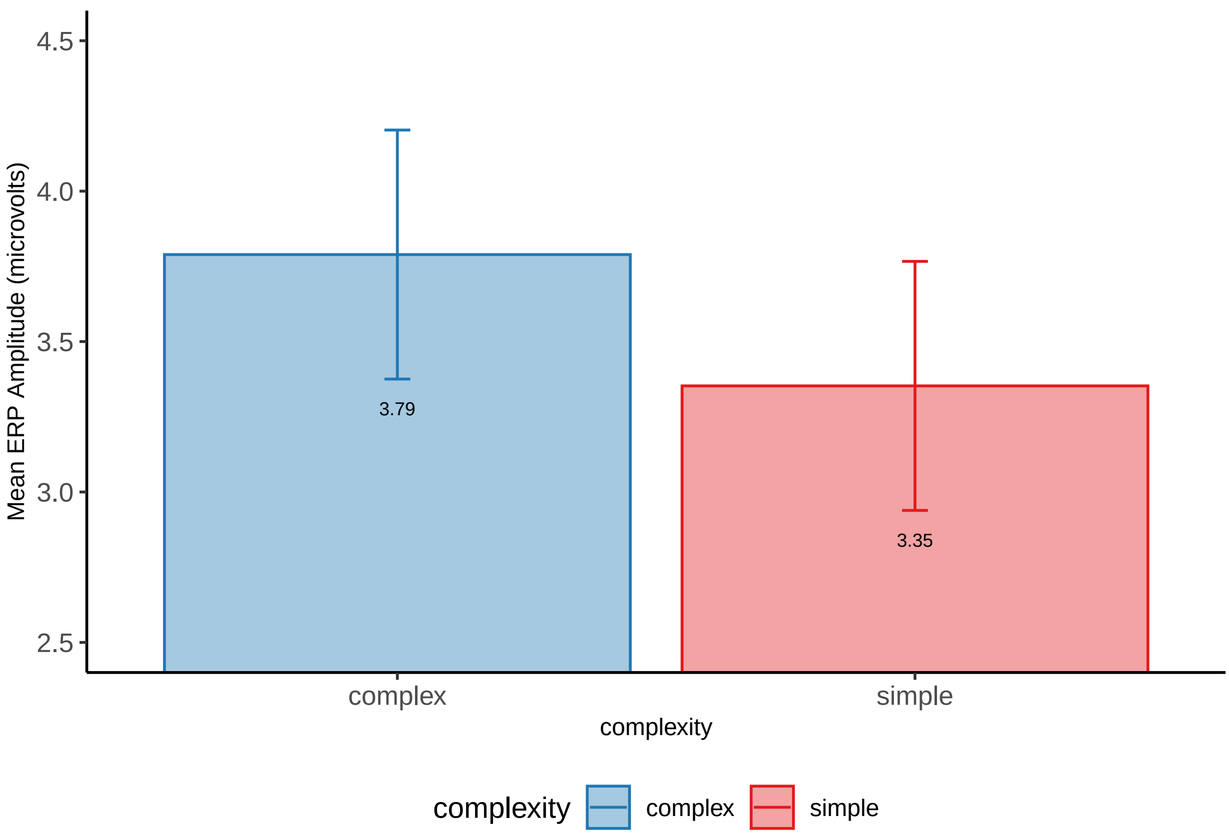
```
reduced_model <- update(anova_model_2,
                        . ~ . - complexity - complexity:family_size - complexity:lang_type_ortho - complexity:family_size:lang_type_ortho)
anova(anova_model_2, reduced_model)
```

```
|| Data: data
|| Models:
|| reduced_model: value ~ lang_type_ortho + family_size + laterality + anteriority + lang_type_ortho:family_size + laterality:anteriority + (1 | S
|| anova_model_2: value ~ lang_type_ortho * family_size * complexity + laterality * anteriority + (1 | SubjID)
||               npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
|| reduced_model   14 6991.9 7066.0 -3481.9   6963.9
|| anova_model_2   18 6978.1 7073.4 -3471.0   6942.1 21.779  4  0.0002218 ***
|| ---
|| Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

A model including Complexity and its interactions provided a significantly better fit than a model without it, $\chi^2(4) = 21.779$, $p < .001$, indicating that the complexity of words modulates the ERP responses.

3.2.3.1.3 Plot for Complexity

Complexity



3.2.3.2 lang_type_ortho × complexity (Interaction)

```
pairs_2 <- emmeans(anova_model_2, pairwise ~ lang_type_ortho * complexity, adjust = "bonferroni", pbkrtest.limit = 6480)
(pairs_df_2 <- as.data.frame(pairs_2$contrasts))
```

3.2.3.2.1 Custom contrasts for lang_type_ortho × complexity (Interaction)

```
||  contrast                      estimate        SE      df t.ratio p.value
||  High complex - Low complex -0.8234243 0.8277551   40.85  -0.995  1.0000
||  High complex - High simple  0.1848937 0.1660097 1421.00   1.114  1.0000
||  High complex - Low simple  -0.1353009 0.8277551   40.85  -0.163  1.0000
||  Low complex - High simple   1.0083180 0.8277551   40.85   1.218  1.0000
```

```
||  Low complex - Low simple    0.6881235 0.1876554 1421.00   3.667  0.0015
||  High simple - Low simple   -0.3201946 0.8277551   40.85  -0.387  1.0000
||
|| Results are averaged over the levels of: family_size, laterality, anteriority
|| Degrees-of-freedom method: kenward-roger
|| P value adjustment: bonferroni method for 6 tests
```

```
selected_contrasts_cmplxty_2 <- pairs_2$contrasts[pairs_df_2$contrast %in% c("High complex - High simple",
                                                                "Low complex - Low simple"),]
selected_contrasts_orthtyp_2 <- pairs_2$contrasts[pairs_df_2$contrast %in% c("High complex - Low complex",
                                                                        "High simple - Low simple"), ]

selected_contrasts_df_cmplxty_2 <- as.data.frame(selected_contrasts_cmplxty_2)  # Convert the emmGrid object to a dataframe
selected_contrasts_df_orthtyp_2 <- as.data.frame(selected_contrasts_orthtyp_2)

cohensd_complex_2 <- as.data.frame(cohens_d(value ~ lang_type_ortho,
         data = subset(n250_2_nonwords, complexity == "complex")))
cohensd_simple_2 <- as.data.frame(cohens_d(value ~ lang_type_ortho,
         data = subset(n250_2_nonwords, complexity == "simple")))
cohensd_hi_ortho_2 <- as.data.frame(cohens_d(value ~ complexity,
         data = subset(n250_2_nonwords, lang_type_ortho == "High")))
cohensd_lo_ortho_2 <- as.data.frame(cohens_d(value ~ complexity,
         data = subset(n250_2_nonwords, lang_type_ortho == "Low")))

cohensd_orthtyp_2 <- bind_rows(complex = cohensd_complex_2,
                              simple = cohensd_simple_2,
                              .id = "cmplxty")

cohensd_cmplxty_2 <- bind_rows(hi_ortho = cohensd_hi_ortho_2,
                              lo_ortho = cohensd_lo_ortho_2,
                              .id = "orthtyp")

(orthtyp_contrasts_df_2 <- bind_cols(selected_contrasts_df_orthtyp_2,cohensd_orthtyp))
```

```
||  contrast                     estimate        SE    df t.ratio p.value cmplxty
||  High complex - Low complex -0.8234243 0.8277551 40.85  -0.995  0.6514 complex
||  High simple - Low simple   -0.3201946 0.8277551 40.85  -0.387  1.0000 simple
||    Cohens_d   CI      CI_low    CI_high
||  -0.0221697 0.95 -0.14086556 0.09653626
||   0.1401066 0.95  0.02123269 0.25891684
||
|| Results are averaged over the levels of: family_size, laterality, anteriority
|| Degrees-of-freedom method: kenward-roger
|| P value adjustment: bonferroni method for 2 tests
```

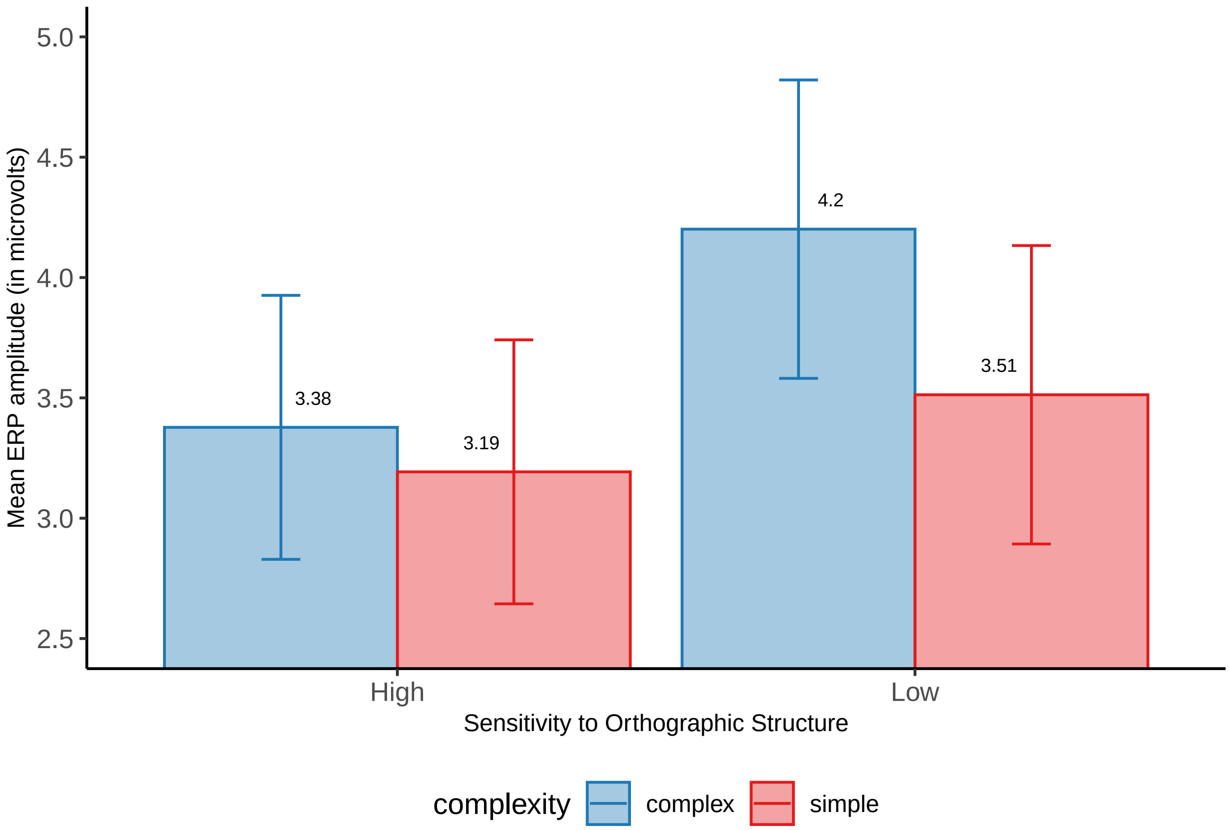
```
(cmplxty_contrasts_df_2 <- bind_cols(selected_contrasts_df_cmplxty_2,cohensd_cmplxty))
```

```
||  contrast                    estimate        SE   df t.ratio p.value orthtyp
||  High complex - High simple 0.1848937 0.1660097 1421   1.114  0.5311 hi_ortho
||  Low complex - Low simple   0.6881235 0.1876554 1421   3.667  0.0005 lo_ortho
||     Cohens_d   CI      CI_low    CI_high
||  -0.06912289 0.95 -0.19260878 0.05439733
||   0.09448232 0.95 -0.01932975 0.20825460
||
|| Results are averaged over the levels of: family_size, laterality, anteriority
|| Degrees-of-freedom method: kenward-roger
|| P value adjustment: bonferroni method for 2 tests
```

```
(langtyp.cmplxty_means_2 <- as.data.frame(pairs_2$emmeans))
```

```
||  lang_type_ortho complexity   emmean        SE    df lower.CL upper.CL
||  High            complex    3.377539 0.5484616 40.85 2.269774 4.485303
||  Low             complex    4.200963 0.6199746 40.85 2.948759 5.453167
||  High            simple     3.192645 0.5484616 40.85 2.084880 4.300410
||  Low             simple     3.512840 0.6199746 40.85 2.260635 4.765044
||
|| Results are averaged over the levels of: family_size, laterality, anteriority
|| Degrees-of-freedom method: kenward-roger
|| Confidence level used: 0.95
```

Plots for lang_type_ortho × complexity (Interaction)



3.2.3.3 family_size x complexity (Interaction)

```
pairs_2 <- emmeans(anova_model_2, pairwise ~ complexity * family_size, adjust = "bonferroni", pbkrtest.limit = 6480)
(pairs_df_2 <- as.data.frame(pairs_2$contrasts))
```

3.2.3.3.1 Custom contrasts for family_size x complexity (Interaction)

```
||  contrast                        estimate        SE   df t.ratio p.value
||  complex large - simple large  -0.0889980 0.1771634 1421  -0.502  1.0000
||  complex large - complex small -0.5108543 0.1771634 1421  -2.884  0.0240
||  complex large - simple small   0.4511609 0.1771634 1421   2.547  0.0659
||  simple large - complex small  -0.4218563 0.1771634 1421  -2.381  0.1043
||  simple large - simple small    0.5401589 0.1771634 1421   3.049  0.0140
||  complex small - simple small   0.9620152 0.1771634 1421   5.430  <.0001
||
|| Results are averaged over the levels of: lang_type_ortho, laterality, anteriority
|| Degrees-of-freedom method: kenward-roger
|| P value adjustment: bonferroni method for 6 tests
```

```
selected_contrasts_famsize_2 <- pairs_2$contrasts[pairs_df_2$contrast %in% c("complex large - complex small",
                                                                              "simple large - simple small"),]
selected_contrasts_cmplxty_2 <- pairs_2$contrasts[pairs_df_2$contrast %in% c("complex small - simple small",
                                                                              "complex large - simple large"), ]
selected_contrasts_df_famsize_2 <- as.data.frame(selected_contrasts_famsize_2)
# Convert the emmGrid object to a dataframe
selected_contrasts_df_cmplxty_2 <- as.data.frame(selected_contrasts_cmplxty_2)
# Convert the emmGrid object to a dataframe
cohensd_small_2 <- as.data.frame(cohens_d(value ~ complexity,
         data = subset(n250_2_nonwords, family_size == "small")))
cohensd_large_2 <- as.data.frame(cohens_d(value ~ complexity,
         data = subset(n250_2_nonwords, family_size == "large")))
cohensd_complex_2 <- as.data.frame(cohens_d(value ~ family_size,
         data = subset(n250_2_nonwords, complexity == "complex")))
cohensd_simple_2 <- as.data.frame(cohens_d(value ~ family_size,
         data = subset(n250_2_nonwords, complexity == "simple")))

cohensd_famsize_2 <- bind_rows(complex = cohensd_complex_2,
                               simple = cohensd_simple_2,
                               .id = "cmplxty")
```

```r
cohensd_cmplxty_2 <- bind_rows(large = cohensd_large_2,
                               small = cohensd_small_2,
                               .id = "famsize")



(cmplxty_contrasts_df_2 <- bind_cols(selected_contrasts_df_cmplxty_2,cohensd_cmplxty_2))
```

```
||  contrast                       estimate        SE   df t.ratio p.value famsize
||  complex large - simple large -0.0889980 0.1771634 1421  -0.502  1.0000 large
||  complex small - simple small  0.9620152 0.1771634 1421   5.430  <.0001 small
||     Cohens_d   CI     CI_low   CI_high
||  -0.04748088 0.95 -0.1900517 0.0951214
||   0.26827491 0.95  0.1249790 0.4113945
||
|| Results are averaged over the levels of: lang_type_ortho, laterality, anteriority
|| Degrees-of-freedom method: kenward-roger
|| P value adjustment: bonferroni method for 2 tests
```

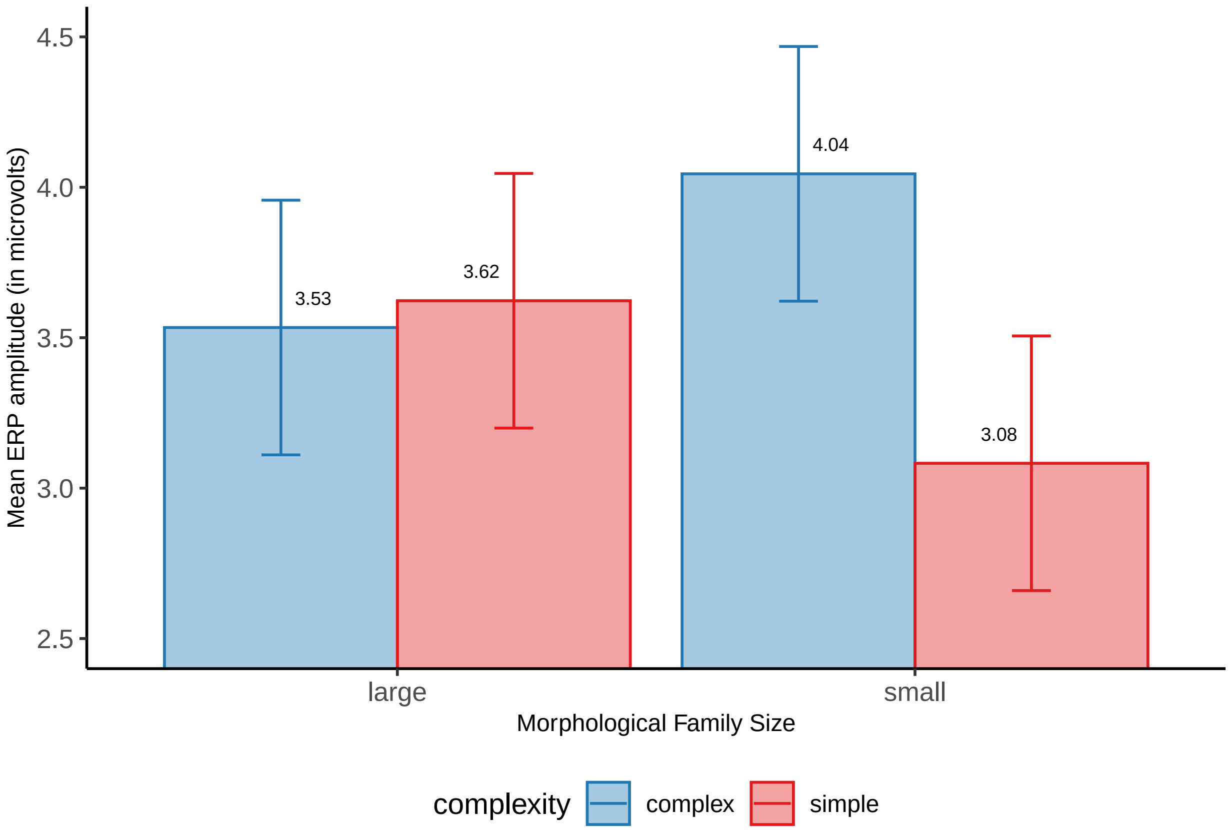
```r
(famsize_contrasts_df_2 <- bind_cols(selected_contrasts_df_famsize_2,cohensd_famsize_2))
```

```
||  contrast                        estimate        SE   df t.ratio p.value
||  complex large - complex small -0.5108543 0.1771634 1421  -2.884  0.0080
||  simple large - simple small    0.5401589 0.1771634 1421   3.049  0.0047
||  cmplxty   Cohens_d   CI      CI_low     CI_high
||  complex -0.1650151 0.95 -0.30776992 -0.02215114
||  simple   0.1446617 0.95  0.00186067  0.28736703
||
|| Results are averaged over the levels of: lang_type_ortho, laterality, anteriority
|| Degrees-of-freedom method: kenward-roger
|| P value adjustment: bonferroni method for 2 tests
```

```r
(famsize_cmplxty_means_2 <- as.data.frame(pairs_2$emmeans))
```

```
||  complexity family_size   emmean        SE    df lower.CL upper.CL
||  complex    large       3.533824 0.4232509 44.67 2.681180 4.386468
||  simple     large       3.622822 0.4232509 44.67 2.770178 4.475466
||  complex    small       4.044678 0.4232509 44.67 3.192034 4.897322
||  simple     small       3.082663 0.4232509 44.67 2.230019 3.935307
||
|| Results are averaged over the levels of: lang_type_ortho, laterality, anteriority
|| Degrees-of-freedom method: kenward-roger
|| Confidence level used: 0.95
```

Plots for family_size x complexity (Interaction)


Mean ERP amplitude (in microvolts)
4.5
4.0
3.5
3.0
2.5
3.53
3.62
4.04
3.08
large
small
Morphological Family Size
complexity
complex
simple