EPL 2022 - Assignment 2

Yanming Li 2033070
Xinhao Lan 1082620
Sili Wang 4621514
Koert Möllers 5942446
Bram Pramono 7978359

Q1: analyze the experiment

Find answers to at least these questions:

1. What is the design of the experiment? That is, how many factors are there? How many conditions? How was the data presented - in Latin square design? Were they randomized?

In this experiment, they applied within-subject experimental design. There were two binary factors, namely the plausibility and interference. The conditions should be the combination of these two factors. To control the experimental flow, they applied pseudo-randomization in a Latin-square design.

2. Are the stimuli available? If so, where?

The texts (items) they used in the main experiment are available in Appendix A of the paper.

3. How many participants took part and how many items were used?

Initially, there were 40 items and 40 fillers. After the pre-tests were conducted and analyzed, they reduced the items to 32 and increased the number of fillers to 96 in which 16 fillers originated from another study.

In total there were 48 participants who were native English speakers from the University of Reading. For the pre-tests, 24 participants took part in both pre-tests. As for the main experiment, even though it was not mentioned explicitly, we believe that the participants in the pre-tests were not involved in the main experiment to keep the integrity of the experiment. This left the researchers with 24 participants for the main experiment.

4. What experimental method was used in this study (self-paced reading, acceptability study, fMRI. . .)? What kind of data (reaction times, ratings, EEG. . .) were gathered?

The main experiment used the eye-movement monitoring method. There are three kinds of reading time measurements collected, namely the *first pass times*, *regression path times,* and

total viewing times. The report used the *total viewing times* as the global index of processing. Aside from these measurements, they also collected the eye fixation durations. Some outliers in the eye fixation were removed.

5. What statistical analyses were conducted? (Do not go into details here, name the statistical technique or tests used.)

For the statistical analysis, they used linear mixed-effects models with crossed random effects for subjects and items. They also mentioned a specific p-value calculation using the Satterthwaite approximation. In the analysis, the reading times were transformed using log-transformation to reduce skewness and they used models with sum coded (-1, 1).

Q2: t-test and linear regression

Use the dataset to address the following question: is it so that early on (in first-pass duration) we can observe that readers are sensitive to the effect of plausibility? Check this for two regions: the verb and the spillover. You will check it by two methods: using t-test and using linear models.

For the t-test: Make sure to develop the correct t-test. Keep in mind you should do some aggregations on the data and decide whether the test is paired or unpaired (it might also help to think about how many degrees of freedom the test should have).

We should use paired t-test.

For verb region:

```
aggregated_plausible_verb <- summarise(group_by(filter(exp1, plausibility == "plausible",
  measure == "fp", region == "verb"), subject), avg_rt = mean(rt))

aggregated_implausible_verb <- summarise(group_by(filter(exp1, plausibility == "implausible",
  measure == "fp", region == "verb"), subject), avg_rt = mean(rt))

t.test(aggregated_plausible_verb$avg_rt, aggregated_implausible_verb$avg_rt, paired = T)
```

The result is:

```
data:  aggregated_plausible_verb$avg_rt and aggregated_implausible_verb$avg_rt
t = -1.4401, df = 47, p-value = 0.1565
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -18.108417   2.998869
sample estimates:
mean of the differences
```

-7.554774

For spillover region:

```
aggregated_plausible_spillover <- summarise(group_by(filter(exp1, plausibility == "plausible",
  measure == "fp", region == "spillover"), subject), avg_rt = mean(rt))

Aggregated_implausible_spillover <- summarise(group_by(filter(exp1, plausibility ==
  "implausible", measure == "fp", region == "spillover"), subject), avg_rt = mean(rt))

t.test(aggregated_plausible_spillover$avg_rt, aggregated_implausible_spillover$avg_rt, paired =
  T)
```

The result is:

```
data:  aggregated_plausible_spillover$avg_rt and aggregated_implausible_spillover$avg_rt
t = -3.695, df = 47, p-value = 0.000573
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -63.38427 -18.69559
sample estimates:
mean of the differences
        -41.03993
```

For the linear model: Develop a linear model (Note: not a mixed-effects model yet). Try two versions of the linear model:

- ***In version 1, the dependent variable is reading times (so, no aggregation, no averaging, just plain reading times).***

For verb region:

```
m_version_1_verb <- lm(rt ~ plausibility, filter(exp1, measure == "fp", region == "verb"))

summary(m_version_1_verb)
```

The result is:

```
Call:
lm(formula = rt ~ plausibility, data = filter(exp1, measure ==
   "fp", region == "verb"))

Residuals:
   Min     1Q  Median     3Q    Max
-244.90  -66.04  -14.90   53.10  695.83

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
```

```
(Intercept)          244.901     4.333  56.521   <2e-16 ***
plausibilityplausible   -7.729     6.126  -1.262    0.207
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 120 on 1533 degrees of freedom
Multiple R-squared:  0.001037,  Adjusted R-squared:  0.0003858
F-statistic: 1.592 on 1 and 1533 DF,  p-value: 0.2072
```

For spillover region:

```
m_version_1_spillover <- lm(rt ~ plausibility, filter(exp1, measure == "fp", region == "spillover"))
summary(m_version_1_spillover)
```

The result is:

```
Call:
lm(formula = rt ~ plausibility, data = filter(exp1, measure ==
   "fp", region == "spillover"))

Residuals:
   Min     1Q  Median     3Q    Max
-457.39 -210.91  -58.39  124.85 1697.61

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)          457.39      10.08   45.36  < 2e-16 ***
plausibilityplausible   -41.48      14.26   -2.91  0.00367 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 279.3 on 1533 degrees of freedom
Multiple R-squared:  0.005493,  Adjusted R-squared:  0.004844
F-statistic: 8.467 on 1 and 1533 DF,  p-value: 0.003668
```

- ***In version 2, the dependent variable is subject-aggregated reading times.***

For verb region:

```
aggregated_subject_verb <- summarise(group_by(filter(exp1, measure == "fp", region == "verb"),
  subject, plausibility), avg_rt = mean(rt))

m_version_2_subject_verb <- lm(avg_rt ~ plausibility, aggregated_subject_verb)

summary(m_version_2_subject_verb)
```

The result is:

```
Call:
lm(formula = avg_rt ~ plausibility, data = aggregated_subject_verb)

Residuals:
    Min      1Q  Median      3Q     Max
-133.660  -33.062   -3.012   28.564  111.711

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)          244.727      7.106  34.439   <2e-16 ***
plausibilityplausible   -7.555     10.049  -0.752    0.454
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 49.23 on 94 degrees of freedom
Multiple R-squared:  0.005976,  Adjusted R-squared:  -0.004598
F-statistic: 0.5651 on 1 and 94 DF,  p-value: 0.4541
```

We get an adjusted R-squared which is smaller than 0 and we think it is because of the adjusted R^2 formula. It can happen when the actual R-squared is very close to 0, which is the case for this model. Basically means that this model can explain next to no variance, is probably a bad model as result.

For spillover region:

```
aggregated_subject_spillover <- summarise(group_by(filter(exp1, measure == "fp", region ==
  "spillover"), subject, plausibility), avg_rt = mean(rt))

m_version_2_subject_spillover <- lm(avg_rt ~ plausibility, aggregated_subject_spillover)

summary(m_version_2_subject_spillover)
```

The result is:

```
Call:
lm(formula = avg_rt ~ plausibility, data = aggregated_subject_spillover)

Residuals:
   Min     1Q  Median     3Q    Max
-341.28  -78.72  -19.68   71.08  448.93

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)          456.95      20.01  22.841   <2e-16 ***
```

```
plausibilityplausible   -41.04      28.29  -1.451     0.15
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 138.6 on 94 degrees of freedom
Multiple R-squared:  0.02189,   Adjusted R-squared:  0.01149
F-statistic: 2.104 on 1 and 94 DF,  p-value: 0.1502
```

Is the linear model justified for version 1 or version 2 or both? Why? Why not?

Version 2 is more justified, because linear models require the data to be independent. Due to the design of the experiment, which is the within-subject, there are dependencies between the collected data. To remove this dependency, aggregating the data per subject will remove the dependency..

Finally, is there any worry about the fact that you test the same question twice: on the verb region and the spillover region? To answer this, it helps to check what Type 1 and Type 2 errors are and what Bonferroni correction is (you can check Wikipedia or other sources on these topics).

To answer this question, we will start by explaining briefly what Type 1 error and Bonferroni correction are. Type 1 error is also known as false positive in which the null hypothesis is rejected, while it should have been correct. As for Bonferroni correction, it is a method to recalculate the alpha level to use in the experiment when experiments are done multiple times. The recalculation uses the value k that indicates how often the experiment has been conducted. The k-value is used to determine the acceptability of p-value. The formula to determine the p-value would then be α / k. If the number of tests is two, then the p-value should be 0.05/2 = 0.025.

When the same experiment design is conducted multiple times, the probability of the Type 1 error occurring will increase (Winter, 2019, p. 175). Therefore, using the same alpha level to indicate significance will be invalid. To improve the acceptability of the result after multiple tests, Bonferroni correction should be applied.

Q3: linear models and mixed-effects models

Create this mixed-effects model on (i) first pass reading times on the region Verb and the region Spillover, (ii) for total viewing times on the same two regions (so you should have 4 mixed-effects model in total). Make sure you include correct fixed effects and your model also includes the random effect structures for subjects and items (it might help to check slides from Monday). What results did you get?

(i) first passing times:

(1) For verb region:

For plausibility:

```
m_log_rt_fp_verb_plausibility = lmer(log(rt) ~ 1 + plausibility + (1|item) + (1|subject),
  filter(log_data, measure == "fp",region == "verb"))
summary(m_log_rt_fp_verb_plausibility)
```

The result is:

```
Linear mixed model fit by REML ['lmerMod']
Formula: log(rt) ~ 1 + plausibility + (1 | item) + (1 | subject)
  Data: filter(log_data, measure == "fp", region == "verb")

REML criterion at convergence: 5230.5

Scaled residuals:
   Min     1Q  Median     3Q    Max
-4.2405 -0.0452  0.1735  0.4305  1.8004

Random effects:
 Groups   Name        Variance Std.Dev.
 subject  (Intercept) 0.1589   0.3987
 item     (Intercept) 0.0446   0.2112
 Residual             1.6580   1.2876
Number of obs: 1535, groups:  subject, 48; item, 32

Fixed effects:
                    Estimate Std. Error t value
(Intercept)          5.18257    0.08287  62.540
plausibilityplausible -0.08778    0.06575  -1.335

Correlation of Fixed Effects:
            (Intr)
plsbltyplsb -0.397
```

For plausibility * interference:

```
m_log_rt_fp_verb_plausibility_interference = lmer(log(rt) ~ 1 + plausibility * interference +
  (1|item) + (1|subject), filter(log_data, measure == "fp",region == "verb"))
summary(m_log_rt_fp_verb_plausibility_interference)
```

The result is:

```
Linear mixed model fit by REML ['lmerMod']
Formula: log(rt) ~ 1 + plausibility * interference + (1 | item) + (1 |      subject)
  Data: filter(log_data, measure == "fp", region == "verb")

REML criterion at convergence: 5232.4
```

Scaled residuals:
```
    Min      1Q  Median      3Q     Max
-4.2396 -0.0440  0.1746  0.4293  1.8005
```

Random effects:
```
 Groups   Name        Variance Std.Dev.
 subject  (Intercept) 0.15876  0.3984
 item     (Intercept) 0.04433  0.2106
 Residual             1.65608  1.2869
Number of obs: 1535, groups:  subject, 48; item, 32
```

Fixed effects:
```
                                    Estimate Std. Error t value
(Intercept)                          5.27390    0.09497  55.531
plausibilityplausible               -0.18092    0.09296  -1.946
interferencenointerf                -0.18242    0.09297  -1.962
plausibilityplausible:interferencenointerf  0.18604    0.13142   1.416
```

Correlation of Fixed Effects:
```
            (Intr) plsblt intrfr
plsbltyplsb -0.490
intrfrncnnt -0.490  0.501
plsbltypls:  0.347 -0.707 -0.707
```

(2) For spillover region:

For plausibility:

```
m_log_rt_fp_spillover_plausibility = lmer(log(rt) ~ 1 + plausibility + (1|item) + (1|subject),
  filter(log_data, measure == "fp",region == "spillover"))
summary(m_log_rt_fp_spillover_plausibility)
```

The result is:

```
Linear mixed model fit by REML ['lmerMod']
Formula: log(rt) ~ 1 + plausibility + (1 | item) + (1 | subject)
  Data: filter(log_data, measure == "fp", region == "spillover")

REML criterion at convergence: 4452.1

Scaled residuals:
    Min      1Q  Median      3Q     Max
-6.0208 -0.2685  0.0969  0.4430  3.1777

Random effects:
 Groups   Name        Variance Std.Dev.
 subject  (Intercept) 0.28072  0.5298
```

```
 item     (Intercept) 0.03472  0.1863
 Residual             0.96645  0.9831
Number of obs: 1535, groups:  subject, 48; item, 32

Fixed effects:
                    Estimate Std. Error t value
(Intercept)          5.85221    0.09052  64.651
plausibilityplausible -0.16122    0.05020  -3.212

Correlation of Fixed Effects:
            (Intr)
plsbltyplsb -0.277
```

For plausibility * interference:

```
m_log_rt_fp_spillover_plausibility_interference = lmer(log(rt) ~ 1 + plausibility * interference +
  (1|item) + (1|subject), filter(log_data, measure == "fp",region == "spillover"))
summary(m_log_rt_fp_spillover_plausibility_interference)
```

The result is:

```
Linear mixed model fit by REML ['lmerMod']
Formula: log(rt) ~ 1 + plausibility * interference + (1 | item) + (1 |      subject)
   Data: filter(log_data, measure == "fp", region == "spillover")

REML criterion at convergence: 4457.3

Scaled residuals:
    Min      1Q  Median      3Q     Max
-5.9858 -0.2778  0.0997  0.4496  3.2124

Random effects:
 Groups   Name        Variance Std.Dev.
 subject  (Intercept) 0.28059  0.5297
 item     (Intercept) 0.03487  0.1867
 Residual             0.96662  0.9832
Number of obs: 1535, groups:  subject, 48; item, 32

Fixed effects:
                                    Estimate Std. Error t value
(Intercept)                         5.882258   0.097267  60.475
plausibilityplausible               -0.156490   0.071023  -2.203
interferencenointerf                -0.060005   0.071033  -0.845
plausibilityplausible:interferencenointerf -0.009544   0.100406  -0.095

Correlation of Fixed Effects:
```

(Intr) plsblt intrfr
plsbltyplsb -0.366
intrfrncnnt -0.366 0.501
plsbltypls: 0.259 -0.707 -0.707

(ii) total viewing times:

(3) For verb region:

For plausibility:

```
m_log_rt_tt_verb_plausibility = lmer(log(rt) ~ 1 + plausibility + (1|item), filter(log_data, measure
  == "tt", region == "verb"))
summary(m_log_rt_tt_verb_plausibility)
```

The result is:

```
Linear mixed model fit by REML ['lmerMod']
Formula: log(rt) ~ 1 + plausibility + (1 | item)
  Data: filter(log_data, measure == "tt", region == "verb")

REML criterion at convergence: 6362.4

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.06792 -0.01641  0.27141  0.54000  1.51558

Random effects:
 Groups   Name        Variance Std.Dev.
 item     (Intercept) 0.2504   0.5004
 Residual             3.5750   1.8908
Number of obs: 1535, groups:  item, 32

Fixed effects:
                    Estimate Std. Error t value
(Intercept)          5.26426    0.11175  47.106
plausibilityplausible -0.36776    0.09655  -3.809

Correlation of Fixed Effects:
            (Intr)
plsbltyplsb -0.432
```

For plausibility * interference:

```
m_log_rt_tt_verb_plausibility_interference = lmer(log(rt) ~ 1 + plausibility * interference +
  (1|item), filter(log_data, measure == "tt", region == "verb"))
summary(m_log_rt_tt_verb_plausibility_interference)
```

The result is:

```
Linear mixed model fit by REML ['lmerMod']
Formula: log(rt) ~ 1 + plausibility * interference + (1 | item)
```

Data: filter(log_data, measure == "tt", region == "verb")

REML criterion at convergence: 6366

Scaled residuals:

Min	1Q	Median	3Q	Max
-3.07966	-0.01836	0.26658	0.53787	1.54183

Random effects:

Groups	Name	Variance	Std.Dev.
item	(Intercept)	0.2505	0.5005
Residual		3.5782	1.8916

Number of obs: 1535, groups: item, 32

Fixed effects:

	Estimate	Std. Error	t value
(Intercept)	5.24012	0.13107	39.981
plausibilityplausible	-0.39345	0.13665	-2.879
interferencenointerf	0.04821	0.13668	0.353
plausibilityplausible:interferencenointerf	0.05145	0.19319	0.266

Correlation of Fixed Effects:

	(Intr)	plsblt	intrfr
plsbltyplsb	-0.522		
intrfrncnnt	-0.522	0.501	
plsbltypls:	0.369	-0.707	-0.707

(4) For spillover region:

For plausibility:

m_log_rt_tt_spillover_plausibility = lmer(log(rt) ~ 1 + plausibility + (1|item), filter(log_data, measure == "tt", region == "spillover"))

summary(m_log_rt_tt_spillover_plausibility)

The result is:

Linear mixed model fit by REML ['lmerMod']

Formula: log(rt) ~ 1 + plausibility + (1 | item)

Data: filter(log_data, measure == "tt", region == "spillover")

REML criterion at convergence: 4314.8

Scaled residuals:

Min	1Q	Median	3Q	Max
-6.6903	-0.3219	0.1083	0.5249	1.8187

Random effects:
 Groups Name Variance Std.Dev.
 item (Intercept) 0.06451 0.2540
 Residual 0.94045 0.9698
Number of obs: 1535, groups: item, 32

Fixed effects:
 Estimate Std. Error t value
(Intercept) 6.41973 0.05694 112.736
plausibilityplausible -0.28193 0.04952 -5.693

Correlation of Fixed Effects:
 (Intr)
plsbltyplsb -0.435

For plausibility * interference:

```
m_log_rt_tt_spillover_plausibility_interference = lmer(log(rt) ~ 1 + plausibility*interference +
  (1|item), filter(log_data, measure == "tt", region == "spillover"))
summary(m_log_rt_tt_spillover_plausibility_interference)
```

The result is:

Linear mixed model fit by REML ['lmerMod']
Formula: log(rt) ~ 1 + plausibility * interference + (1 | item)
 Data: filter(log_data, measure == "tt", region == "spillover")

REML criterion at convergence: 4319.3

Scaled residuals:
 Min 1Q Median 3Q Max
-6.7490 -0.3195 0.1046 0.5162 1.8500

Random effects:
 Groups Name Variance Std.Dev.
 item (Intercept) 0.06429 0.2536
 Residual 0.94020 0.9696
Number of obs: 1535, groups: item, 32

Fixed effects:
 Estimate Std. Error t value
(Intercept) 6.36522 0.06683 95.250
plausibilityplausible -0.23634 0.07005 -3.374
interferencenointerf 0.10887 0.07006 1.554
plausibilityplausible:interferencenointerf -0.09104 0.09903 -0.919

```
Correlation of Fixed Effects:
            (Intr) plsblt intrfr
plsbltyplsb -0.525
intrfrncnnt -0.525  0.501
plsbltypls:  0.371 -0.707 -0.707
```

Do you find differences for the effect of plausibility when you compare the now-created mixed-effects models and the linear model from Q2?

The difference should be that the coefficients per subject using lm and lmer will be different. By using the fixed-effect's mean of lmer as the Grand mean, we can compare the distance of coefficients for the different models.

Q4: null responses

Address this question: is it correct to code such a case as zero? Or should we rather treat it as missing data? Which option is more adequate? Why?

Treating such situations as missing data would be more appropriate than replacing these values as zeros. When the values are replaced by zeros, the numbers will be taken as the part of the computed data. This will give us a miscalculated result. If these data points are treated as missing data, there are still possibilities to exclude these data points from the computation or even use them in the analysis of the result.

After answering this, try the following: recode all zeroes as missing data (you can use the special element NA in R for this). Then, re-run linear models and mixed-effects models on total viewing times, checking whether the effect of plausibility changes. To answer this question, you should use the linear model from Q2 that did not aggregate any data (so, reading times were the dependent variable).

We use the code below to get the result but results are too many so we don't put it into this docs:
Results of Q4 can find though this link: https://rpubs.com/WVI7/873246

```
m_version_1_verb_missing <- lm(rt ~ plausibility, filter(missing_data, measure == "fp", region ==
  "verb"))
summary(m_version_1_verb_missing)
m_version_1_spillover_missing <- lm(rt ~ plausibility, filter(missing_data, measure == "fp", region
  == "spillover"))
summary(m_version_1_spillover_missing)
aggregated_subject_verb_missing <- summarise(group_by(filter(missing_data, measure == "fp",
  region == "verb"), subject, plausibility), avg_rt = mean(rt))
m_version_2_subject_verb_missing <- lm(avg_rt ~ plausibility, aggregated_subject_verb_missing)
summary(m_version_2_subject_verb_missing)
```

```
aggregated_subject_spillover_missing <- summarise(group_by(filter(missing_data, measure ==
  "fp", region == "spillover"), subject, plausibility), avg_rt = mean(rt))
m_version_2_subject_spillover_missing <- lm(avg_rt ~ plausibility,
  aggregated_subject_spillover_missing)
summary(m_version_2_subject_spillover_missing)
m_log_rt_fp_verb_plausibility_missing = lmer(log(rt) ~ 1 + plausibility + (1|item) + (1|subject),
  filter(missing_data, measure == "fp",region == "verb"))
summary(m_log_rt_fp_verb_plausibility_missing)
m_log_rt_fp_verb_plausibility_interference_missing = lmer(log(rt) ~ 1 + plausibility * interference
  + (1|item) + (1|subject), filter(missing_data, measure == "fp",region == "verb"))
summary(m_log_rt_fp_verb_plausibility_interference_missing)
m_log_rt_fp_spillover_plausibility_missing = lmer(log(rt) ~ 1 + plausibility + (1|item) + (1|subject),
  filter(missing_data, measure == "fp",region == "spillover"))
summary(m_log_rt_fp_spillover_plausibility_missing)
m_log_rt_fp_spillover_plausibility_interference_missing = lmer(log(rt) ~ 1 + plausibility *
  interference + (1|item) + (1|subject), filter(missing_data, measure == "fp",region == "spillover"))
summary(m_log_rt_fp_spillover_plausibility_interference_missing)
m_log_rt_tt_verb_plausibility_missing = lmer(log(rt) ~ 1 + plausibility + (1|item),
  filter(missing_data, measure == "tt", region == "verb"))
summary(m_log_rt_tt_verb_plausibility_missing)
m_log_rt_tt_verb_plausibility_interference_missing = lmer(log(rt) ~ 1 + plausibility * interference
  + (1|item), filter(missing_data, measure == "tt", region == "verb"))
summary(m_log_rt_tt_verb_plausibility_interference_missing)
m_log_rt_tt_spillover_plausibility_missing = lmer(log(rt) ~ 1 + plausibility + (1|item),
  filter(missing_data, measure == "tt", region == "spillover"))
summary(m_log_rt_tt_spillover_plausibility_missing)
m_log_rt_tt_spillover_plausibility_interference_missing = lmer(log(rt) ~ 1 +
  plausibility*interference + (1|item), filter(missing_data, measure == "tt", region == "spillover"))
summary(m_log_rt_tt_spillover_plausibility_interference_missing)
```

Did the estimates of plausibility interaction change when you compare the models in this question to the models in Q2 and Q3? Was the change bigger for mixed-effects models or for linear models?

Yes, the estimates indeed changed. This is due to the difference in the treatment of computations between missing data and zero. If the values would be replaced by zeros the intercepts of these linear models will be lower. In the case of aggregated data, the results are even more concerning. The differences between the two treatments become bigger.

Q5: Does the gender of the selfie-taker predict boringness responses?

We will work with Boring responses. Prepare your dataset and test using t-test whether male selfies are seen as less/more boring than female selfies. Make sure to develop the correct t-test (keep in mind you might have to do some transformations on the data and decide whether the test is paired/unpaired; it might also help to think about how many degrees of freedom the test should have).

Because the participants were evaluating the same set of photos, we decided this test should be a paired t-test.

```
summarise(group_by(selfies, StimGender), n())
```

```
# A tibble: 2 × 2
  StimGender `n()`
  <chr>      <int>
1 Female      1079
2 Male        1075
```

We found these two vectors are not in the same length, so we cut the longer one to 1075.

```
t.test(subset(selfiesStimg,StimGender == 'Male')$Boring,
    subset(selfiesStimg,StimGender == 'Female')$Boring, paired = TRUE)
```

```
        Paired t-test

data:  subset(selfiesStimg, StimGender == "Male")$Boring and subset(selfiesStimg, StimGender == "Female")$Boring
t = -7.5398, df = 1071, p-value = 1.001e-13
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.4866979 -0.2856901
sample estimates:
mean of the differences
              -0.386194
```

Then, develop a mixed-effects model with logit link (i.e., a logistic mixed-effects model) to address the same question.

```
comparegm1 <- glmer(BoringYesNo ~ 1 + (1+StimGender|ResponseId), selfies,
  family=binomial(link="logit"))
summary(comparegm1)
```

```
> summary(compareqm1)
Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) ['glmerMod']
 Family: binomial  ( logit )
Formula: BoringYesNo ~ 1 + (1 + StimGender | ResponseId)
   Data: selfies

     AIC      BIC   logLik deviance df.resid
  2072.4   2094.3  -1032.2   2064.4     1775

Scaled residuals:
    Min      1Q  Median      3Q     Max
-2.7191 -0.6505  0.2036  0.6047  2.3549

Random effects:
 Groups     Name           Variance Std.Dev. Corr
 ResponseId (Intercept)    2.435    1.560
            StimGenderMale 1.644    1.282    0.03
Number of obs: 1779, groups:  ResponseId, 134

Fixed effects:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   0.4660     0.1927   2.418   0.0156 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Then, build a model that tests whether StimGender is a significant predictor for BoringYesNo. Check also whether this model provides a better fit to data than the model without the StimGender predictor. Keep in mind that this should be a mixed-effects model. Try to use the maximal random-effects structure that converges but use only subjects as random factors.

```
comparegm2 <- glmer(BoringYesNo ~ 1 + (1+StimGender|ResponseId) + StimGender, selfies,
  family=binomial(link="logit"))
summary(gm2)
```

```
> summary(gm2)
Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) ['glmerMod']
 Family: binomial  ( logit )
Formula: BoringYesNo ~ 1 + (1 + StimGender | ResponseId) + StimGender
   Data: selfies

     AIC      BIC   logLik deviance df.resid
  2036.3   2063.7  -1013.1   2026.3     1774

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.0691 -0.6422  0.1969  0.6058  2.7051

Random effects:
 Groups     Name           Variance Std.Dev. Corr
 ResponseId (Intercept)    2.8332   1.6832
            StimGenderMale 0.5733   0.7571   -0.20
Number of obs: 1779, groups:  ResponseId, 134

Fixed effects:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)     0.7188     0.1731   4.153 3.28e-05 ***
StimGenderMale -0.9608     0.1399  -6.867 6.56e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Correlation of Fixed Effects:
            (Intr)
StimGendrMl -0.422
```

Do you find similarities and differences in the model? Do both approaches give the same answer wrt the significance of StimGender?

There are indeed differences between the model when you eyeball the summaries of both. Model 2 (with fixed effect) generally has a better fit (AIC is lower than model 1).
When we look at the fixed effect of model 2, stimgender does indeed appear to be a significant predictor by itself.

Additionally if we perform an ANOVA between the two models, we can see that differ significantly.

```
> anova(comparegm1, comparegm2)
Data: selfies
Models:
comparegm1: BoringYesNo ~ 1 + (1 + StimGender | ResponseId)
comparegm2: BoringYesNo ~ 1 + (1 + StimGender | ResponseId) + StimGender
           npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
comparegm1    4 2072.4 2094.3 -1032.2   2064.4
comparegm2    5 2036.3 2063.7 -1013.1   2026.3 38.114  1  6.672e-10 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

To try to maximize the random effects and guarantee convergence, we added StimGender and Distance as random effects.

```
lm1 <- glmer(BoringYesNo ~ 1+StimGender+(1+StimGender+Distance|ResponseId),
        data=selfiesStimg, family=binomial(link='logit'))
summary(lm1)
lm2 <- glmer(BoringYesNo ~ 1+(1+StimGender+Distance|ResponseId),
        data=selfiesStimg, family=binomial(link='logit'))
summary(lm2)
anova(lm1,lm2)
```

<u>The result of lm1:</u>

```
Fixed effects:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)    0.7491     0.1758   4.261 2.03e-05 ***
StimGenderMale -0.9827     0.1437  -6.840 7.93e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

<u>The result of lm2:</u>

```
Fixed effects:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   0.4557     0.1956    2.33   0.0198 *
```

The result of comparison of two models:

```
Data: selfiesStimg
Models:
lm2: BoringYesNo ~ 1 + (1 + StimGender + Distance | ResponseId)
lm1: BoringYesNo ~ 1 + StimGender + (1 + StimGender + Distance | ResponseId)
    npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
lm2    7 2068.4 2106.8 -1027.2   2054.4
lm1    8 2032.0 2075.8 -1008.0   2016.0 38.421  1  5.701e-10 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

We can clearly see that the StimGender is a significant predictor for BoringYesNo. And through the comparison of two models, we can see the model using StimGender as a fixed effect performs better than the other.

Q6: Reasoning about the model

Criticize the t-test and logistic mixed-effects models that you just created. While they are (hopefully) right from the statistical perspective, would you conclude from them that general population finds male selfies significantly more/less boring than female selfies? Think about the following issues: data aggregation/transformation, confounds, balancing in the data. All these issues might make it dubious that one can conclude that male selfies are generally found significantly more/less boring than female selfies.

Reasoning for t-test:

The applicable condition of the t-test is that the sample distribution conforms to the normal distribution. We think that we should use some transformations to transform the dataset into a normal distribution and then the result can be more convincing. And the same as the logistic mixed-effect model we just check one individual variable which is StimGender, we are not sure whether other variables can influence the Boring index.

Reasoning for logistic mixed-effect models:

There are a couple things going on in the mixed models that are worth mentioning that can affect the meaningfulness of the above models.

First let’s talk about transformations in the mixed models. For the boring parameter (1-5 scale), we transformed it into a binary variable, either boring or not. However we discarded all ‘3’s from the original variable and only took into account the 1-2 for not boring and 4-5 for boring. While you indeed circumvent the problem of people putting 3’s down as default (which is not very informative) you create a new problem of eliminating legitimate 3’s. As a result the model might be not very representative when we exclude this information.

Furthermore we really only look at the StimGender variable within a dataset that has quite a lot of other potentially interesting parameters. By omitting all of these variables we cannot necessarily conclude whether this relation between StimGender -> Boring predicts as much or is as significant as these models would show.

For example confounding variables that correlate with both the independent variable and dependent variable could confound the relationship between the independent and dependent variables. As seen below, in a correlation matrix e.g. 'Serious' correlates significantly with BoringYesNo and StimGender (Here a binary variable called 'STIMmaleornot').

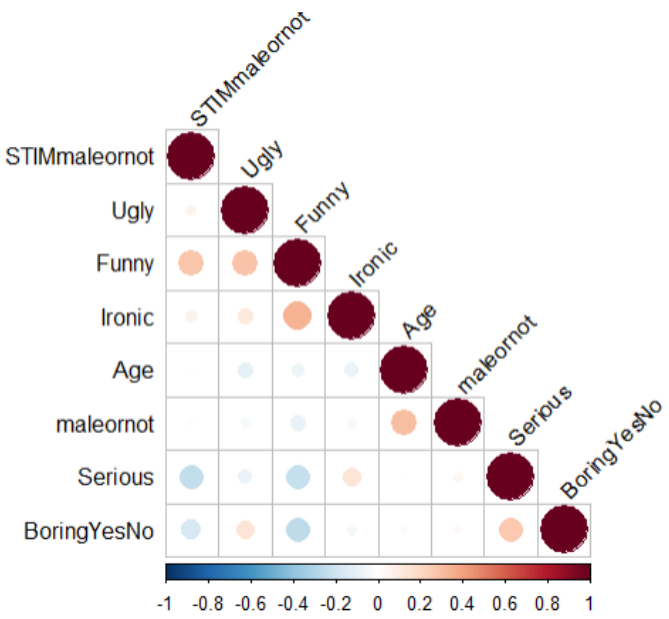


Fig 6.1 correlation matrix of different variables

As a result we cannot say based purely on the models above whether StimGender is actually a significant predictor of boringness.

Q7: Predictions of the model

Suppose you are still interested in boringness of selfies based on the sex of the selfie-taker but you add one confound into the picture: the sex of the person that judges the selfie. You want to see whether females judge male selfies as more boring than female selfies and whether male judgements differ. You furthermore add subjects random effects and have the maximal random effect structure that converges. So, you are thinking of this model:

 # BoringYesNo ~ 1 + StimGender*Gender + (1 + StimGender * Gender |
 # ResponseId)

```
m_Boring <- glmer(as.factor(BoringYesNo) ~ 1 + StimGender * Gender + (1 + StimGender * Gender
  | ResponseId), selfies_new, family = binomial(link = "logit"))
print(summary(m_Boring),corr=FALSE)
```

```
Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) ['glmerMod']
 Family: binomial  ( logit )
Formula: as.factor(BoringYesNo) ~ 1 + StimGender * Gender + (1 + StimGender *      Gender |
   ResponseId)
   Data: selfies_new

     AIC      BIC   logLik deviance df.resid
  2457.4   2536.9  -1214.7   2429.4     2137

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.6261 -0.7182  0.2994  0.6273  2.4474

Random effects:
 Groups     Name                 Variance Std.Dev. Corr
 ResponseId (Intercept)            1.4978   1.2239
          StimGenderMale           0.8796   0.9379    0.36
          GenderMale               1.3195   1.1487    0.01 -0.25
          StimGenderMale:GenderMale 0.5728   0.7568   -0.54 -0.58 -0.21
Number of obs: 2151, groups:  ResponseId, 135

Fixed effects:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)              0.8488     0.1928   4.402 1.07e-05 ***
StimGenderMale          -0.6532     0.2010  -3.249  0.00116 **
GenderMale               0.5111     0.3083   1.658  0.09737 .
StimGenderMale:GenderMale  -0.3863     0.2777  -1.391  0.16417
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
optimizer (Nelder_Mead) convergence code: 0 (OK)
Model is nearly unidentifiable: large eigenvalue ratio
 - Rescale variables?
```

Or some simpler version in the random structure, if this one does not converge. Create the mixed-effects model.

```
m_Boring_simple <- glmer(as.factor(BoringYesNo) ~ 1 + StimGender * Gender + (1 | ResponseId),
   selfies_new, family = binomial(link = "logit"))
print(summary(m_Boring_simple),corr=FALSE)

Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) ['glmerMod']
```

```
 Family: binomial  ( logit )
Formula: as.factor(BoringYesNo) ~ 1 + StimGender * Gender + (1 | ResponseId)
   Data: selfies_new

     AIC      BIC   logLik deviance df.resid
  2454.1   2482.5  -1222.0   2444.1     2146

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.8969 -0.7292  0.3059  0.6261  2.3015

Random effects:
 Groups     Name        Variance Std.Dev.
 ResponseId (Intercept) 2.145    1.464
Number of obs: 2151, groups:  ResponseId, 135

Fixed effects:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)           0.9200     0.2169   4.241 2.23e-05 ***
StimGenderMale       -0.7502     0.1502  -4.995 5.88e-07 ***
GenderMale            0.3555     0.2986   1.191    0.234
StimGenderMale:GenderMale  -0.2152     0.2091  -1.029    0.303
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Second, check predictions of your model. What does the model predict on unseen data? Suppose you got the dataset novel data. You want to see what the model that you just created predicts for those data. With what probability will be each selfie (each row) considered as ugly by a median subject? (Saying that you make predictions for a median subject is just another way to say that you can assume random effects are zero, i.e., you can ignore them.)

```
novel_selfies <- read.csv("C:\\Users\\LYM\\Desktop\\assignment2\\novel_data.csv")
novel_selfies_new <- novel_selfies
novel_selfies_new$StimGender <- ifelse(novel_selfies_new$StimGender== "Male", 1,0)
  #transforming StimGender into binary variable
novel_selfies_new$Gender <- ifelse(novel_selfies_new$Gender== "Male", 1,0) #transforming
  Gender into binary variable
b_1 = fixef(m_Boring)[1]
b_2 = fixef(m_Boring)[2]
b_3 = fixef(m_Boring)[3]
b_4 = fixef(m_Boring)[4]
```

```
y=b_1+b_2*novel_selfies_new$StimGender +b_3*novel_selfies_new$Gender
  +b_4*novel_selfies_new$StimGender*novel_selfies_new$Gender

p =exp(y)/(1+exp(y))
```

First, calculate for each combination of the values in novel data what probability your model estimates. Once you are confident you can do this, calculate probabilities for all the rows (400 data points). As a final check, load the following function:

```
drawprobabilities <- function(probs) {
 if (length(probs) != 400) {
  print("Wrong length of the vector of calculated probabilities. Should be 400 data points.") }
 else {
  matrixprobs <- matrix(ifelse(probs > 0.5, "X", ""), nrow = 20)
  x <- rep(NA, 400)
  y <- rep(NA, 400)
  k <- 1
  for (i in 1:20) {
   for (j in 1:20) {
    if (matrixprobs[i, j] == "X") {
     y[k] <- i
     x[k] <- j
     k <- k + 1
    }
   }
  }
  plot(x, y, xlim = c(0, 40), ylim = c(0, 40), pch = 15)
 }
}
```

Now, run the function drawprobabilities with the argument the vector of predicted probabilities (e.g, if you stored your vector of predicted probabilites for novel data as mypredictions, you would call it as drawprobabilities(mypredictions)). If everything was correct, you should see a picture as a result. What picture do you see?

```
drawprobabilities(p)
```

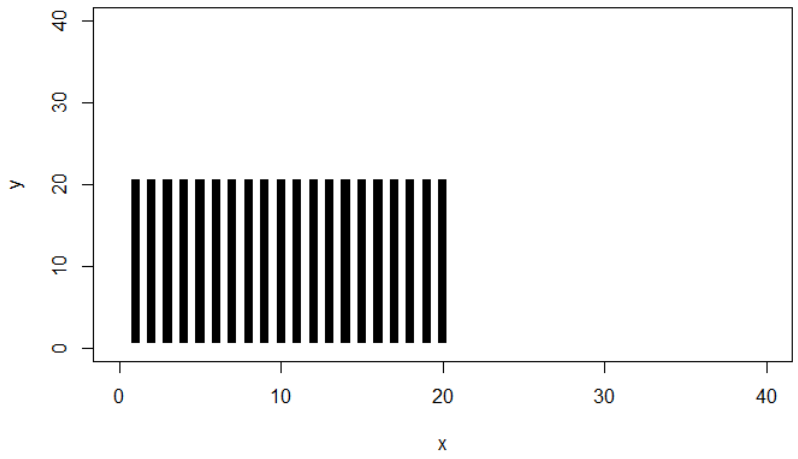


Fig 7.1 Result of complex model

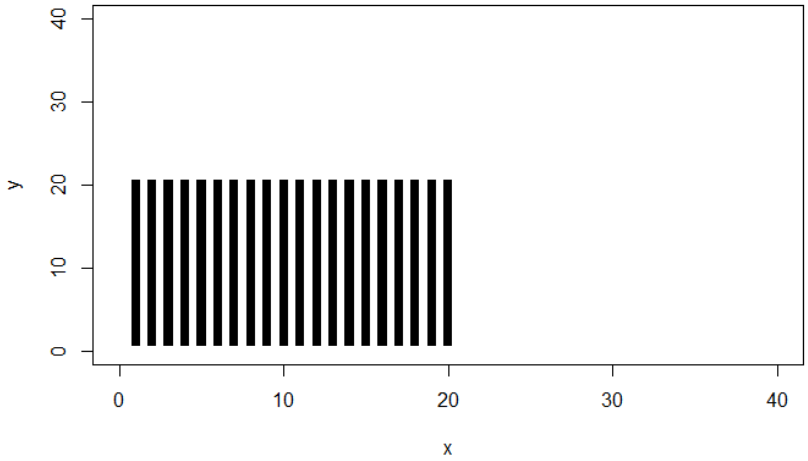
Simpler model:

```
b_1_simple = fixef(m_Boring_simple)[1]
b_2_simple = fixef(m_Boring_simple)[2]
b_3_simple = fixef(m_Boring_simple)[3]
b_4_simple = fixef(m_Boring_simple)[4]

y=b_1_simple+b_2_simple*novel_selfies_new$StimGender +b_3_simple*novel_selfies_new$Gender
+b_4_simple*novel_selfies_new$StimGender*novel_selfies_new$Gender

p =exp(y)/(1+exp(y))

drawprobabilities(p)
```



Result 7.2 Result of Simple model

Q8: Ordinal model

```
library(ordinal)
```

```
# model investigation here
m1 <- clmm(as.factor(Boring) ~ StimGender + (1|ResponseId), data=selfies)
print(summary(m1))
m2 <- clmm(as.factor(Boring) ~ StimGender * Tilt + (1|ResponseId), data=selfies)
print(summary(m2))
m3 <- clmm(as.factor(Boring) ~ StimGender * Distance + (1|ResponseId), data=selfies)
print(summary(m3))
m4 <- clmm(as.factor(Boring) ~ StimGender * Funny + (1|ResponseId), data=selfies)
print(summary(m4))
m5 <- clmm(as.factor(Boring) ~ StimGender * Ironic + (1|ResponseId), data=selfies)
print(summary(m5))
m6 <- clmm(as.factor(Boring) ~ StimGender * Serious + (1|ResponseId), data=selfies)
print(summary(m6))
m7 <- clmm(as.factor(Boring) ~ StimGender * Ugly + (1|ResponseId), data=selfies)
print(summary(m7))
```

By looking at the coefficient of the StimGender = Male, we can see that the male selfies are less boring. This value indicates that the mean value of boringness for the male selfies is - 0.67133 compared to the female selfies.

To investigate for the possible confounds, we tried to find any interactions with other variables that would lead to a reduction of the boringness for male selfies. As we can see in the table below, the male selfies are less tilted, less near, more serious, and slightly more ironic. All of these aspects could contribute to the less boringness judgment of the male selfies.

```
|2nd Var \ Conditions|StimGender = Male|Coeff. of 2nd Var|Interaction|Interaction p-value|
|---|---|---|---|---|
|Tilt = Tilted|-0.4624720|-0.1797142|-0.4340604|0.0073 **|
|Distance = Near|-0.3213502|0.3531456|-0.7080898|1.25e-05 ***|
|Funny|-0.372139361|-0.544575655|0.001992284|0.9776|
|Ironic|-1.0937134|-0.1500431|0.1615640|<2e-16 ***|
|Serious|-1.4201659|0.2075733|0.3472521|1.94e-06 ***|
|Ugly|-0.57899335|0.18044131|-0.04443218|0.516239|
```

2nd Var \ Conditions	StimGender = Male	Coeff. of 2nd Var	Interaction	Interaction p-value
Tilt = Tilted	-0.4624720	-0.1797142	-0.4340604	0.0073 **
Distance = Near	-0.3213502	0.3531456	-0.7080898	1.25e-05 ***
Funny	-0.372139361	-0.544575655	0.001992284	0.9776

Ironic	-1.0937134	-0.1500431	0.1615640	<2e-16 ***
Serious	-1.4201659	0.2075733	0.3472521	1.94e-06 ***
Ugly	-0.57899335	0.18044131	-0.04443218	0.516239

Q9: Inspecting ordinal model (individual component)

References

Winter, B. (2019). Statistics for linguists: An introduction using R. Routledge.