emodash_workbook_stats_R

November 26, 2019

1 Emodash Statistical Analysis

We use R kernel for jupyter for this notebook

#Experiment Design

We conducted an analysis of variance on the feedback report content. The study was a 2x4 repeated measures design (within-subject), with the following factors and levels:

- *Condition*: With, Without (Emodash)
- *Category* (of the feedback): Affective, Motivational, Summative, Formative

The measured variable was the number of *utterances*.

The analysis was carried out using Generalized Linear Mixed Model (GLMM) with Anova and Bonferroni adjustments for *post hoc* comparisons.

We are going to use each feedback report for each participant as a trial in the model. This gives us a quite good amount of data points to fit the the model.

GLMM deals with winthin-subject experiment, and does not require the three assumptions of the ANOVA (normality, independence, and homogeneity), which is best suited for our case. GLMM deals with missing data as we don't have a full balancing.

The data on which we conduct this analysis, can be found inside the folder *data* associated with this notebook.

R packages

[1]:
```
## Imports
## -------

# Data Manipulation Packages
library(readxl)
library(dplyr)
library(tidyr)
library(tidyverse)
library(reshape2)

# For plots
library(ggplot2)

# Fit distribution
library(fitdistrplus)
# For GLMM
```

```r
library(lme4)

# Post Hoc Pairwise Comparision Test,
# emmeans replaces lsmeans (deprecated)
library(emmeans)
# CI for effect size
library(psych)
# For Wald Chi-square test and levene's test
library(car)
```

```
Attaching package: dplyr

The following objects are masked from package:stats:

    filter, lag

The following objects are masked from package:base:

    intersect, setdiff, setequal, union

Registered S3 methods overwritten by 'ggplot2':
  method         from
  [.quosures     rlang
  c.quosures     rlang
  print.quosures rlang
 Attaching packages  tidyverse
1.2.1
 ggplot2 3.1.1      purrr   0.3.2
 tibble  2.1.3      stringr 1.4.0
 readr   1.3.1      forcats 0.4.0
 Conflicts
tidyverse_conflicts()
 dplyr::filter() masks stats::filter()
 dplyr::lag()    masks stats::lag()

Attaching package: reshape2

The following object is masked from package:tidyr:

    smiths

Loading required package: MASS

Attaching package: MASS

The following object is masked from package:dplyr:
```

```
    select

Loading required package: survival
Loading required package: npsurv
Loading required package: lsei
Loading required package: Matrix

Attaching package: Matrix

The following object is masked from package:tidyr:

    expand

Welcome to emmeans.
NOTE -- Important change from versions <= 1.41:
    Indicator predictors are now treated as 2-level factors by default.
    To revert to old behavior, use emm_options(cov.keep = character(0))

Attaching package: psych

The following objects are masked from package:ggplot2:

    %+%, alpha

Loading required package: carData
Registered S3 methods overwritten by 'car':
  method                          from
  influence.merMod                lme4
  cooks.distance.influence.merMod lme4
  dfbeta.influence.merMod         lme4
  dfbetas.influence.merMod        lme4

Attaching package: car

The following object is masked from package:psych:

    logit

The following object is masked from package:purrr:

    some

The following object is masked from package:dplyr:

    recode
```

Loading Data

[0]:
```
# Check path
#getwd()

# Back to data folder
setwd('./data')
```

[0]:
```
# load data

feedbackWith = read_excel('emodash-workbook.v2.xlsx',␣
 ↪sheet="feedback-with-emodash")
feedbackWithout = read_excel('emodash-workbook.v2.xlsx',␣
 ↪sheet="feedback-without-emodash")
```

Check Feedback Data

[4]:
```
# Check With Emodash head

head(feedbackWith)
```

A tibble: 6 Œ 7

bloc <chr>	session_id <chr>	pair <chr>	pair_session <chr>	unit_id <dbl>	MA <chr>	SF <chr>
bloc1	5a0bfe9bb2ee7900015fcd38	P1	P1S1	1	Affective	None
bloc1	NA	P1	P1S1	2	Motivational	Summ
bloc1	NA	P1	P1S1	3	None	Summ
bloc1	NA	P1	P1S1	4	Affective	None
bloc1	NA	P1	P1S1	5	None	Forma
bloc1	NA	P1	P1S1	6	None	Forma

[5]:
```
# Check Without Emodash head

head(feedbackWithout)
```

A tibble: 6 Œ 7

bloc <dbl>	session_id <chr>	pair <chr>	pair_session <chr>	unit_id <dbl>	MA <chr>	SF <chr>
1	59eef14ed7ca3d0001ea9f4c	P5	P5S1	1	Affective	None
1	NA	P5	P5S1	2	Motivational	Summ
1	NA	P5	P5S1	3	Motivational	Summ
1	NA	P5	P5S1	4	Motivational	Forma
1	NA	P5	P5S1	5	Motivational	Forma
1	NA	P5	P5S1	6	None	Forma

Data Overview

- bloc: bloc index of learning sessions
- session_id: id of the learning session
- pair: id of the participant (also id of the pair as we have one-to-one relationship b/ tutor and learner)
- pair_session: id of the session of pair
- unit_id: utterance id per pair, per session
- MA: Motivational and Affective feedback content coding
- MA: Summative and Formative feedback content coding

Stacking Feedback with/without emodash data frames

As each line the data frames feedbackWith and feedbackWithout is a participant ('pair' column) utterance.

- first, we stack both frames together
- second, we count the number of utterances by (pair, pair_session, category, condition)

[0]:
```
# Reshape data
with = melt(feedbackWith, id.vars = c('pair','pair_session'), measure.vars =␣
↪c('SF', 'MA'))
without = melt(feedbackWithout, id.vars = c('pair','pair_session'), measure.
↪vars = c('SF', 'MA'))
```

[0]:
```
# Rename 'value' col to 'category'
names(with)[names(with) == 'value'] = 'category'
names(without)[names(without) == 'value'] = 'category'
```

[0]:
```
# Remove None utterances that fall in any category
with = filter(with, with$category != 'None')
without = filter(without, without$category != 'None')
```

[0]:
```
# Add condition col
without$condition = factor('without')
with$condition = factor('with')
```

[0]:
```
# Stack with and without data frames
contentAnalysis = rbind(with, without)
```

Convert pair_session into long

Here we convert pair_session into a long factor, by keeping index of the session (1, 2, 3, 4, 5). For instance: P1S1 => 1

pair_session will be considered as the 'trial' in GLMM.

[0]:
```
contentAnalysis$pair_session = substr(contentAnalysis$pair_session, start=4,␣
↪stop=5)
```

Making sure that columns are factors

[0]:
```
contentAnalysis$pair = factor(contentAnalysis$pair)
contentAnalysis$pair_session = factor(contentAnalysis$pair_session)
contentAnalysis$condition = factor(contentAnalysis$condition)
contentAnalysis$category = factor(contentAnalysis$category)
```

Count number of utterances by pair, pair_session, condition, category

[0]:
```
# count by our factors
contentAnalysis = dplyr::count(contentAnalysis, pair, pair_session, condition,␣
↪category)
```

Check data

[14]:
```
head(contentAnalysis)
```

A tibble: 6 Œ 5

pair <fct>	pair_session <fct>	condition <fct>	category <fct>	n <int>
P1	1	with	Affective	3
P1	1	with	Formative	4
P1	1	with	Motivational	2
P1	1	with	Summative	4
P1	1	without	Affective	1
P1	1	without	Formative	6

Rename *n* column to utterance

[15]:
```
names(contentAnalysis)[names(contentAnalysis) == 'n'] = 'utterance'

# recheck what we have so far
head(contentAnalysis)

print('Nbr of data points:')
print(count(contentAnalysis))
```

A tibble: 6 Œ 5

pair <fct>	pair_session <fct>	condition <fct>	category <fct>	utterance <int>
P1	1	with	Affective	3
P1	1	with	Formative	4
P1	1	with	Motivational	2
P1	1	with	Summative	4
P1	1	without	Affective	1
P1	1	without	Formative	6

```
[1] "Nbr of data points:"
# A tibble: 1 x 1
      n
  <int>
1   157
```

Summarising data

For instance we can see that on average, we have 4.261 utterances per participant per trial (feedback categories combined).

[16]:
```
# summarise data
summary(contentAnalysis)
```

```
 pair     pair_session   condition           category    utterance
 P1:38    1:34          with    :75   Affective    :28   Min.    : 1.000
 P2:32    2:30          without:82   Formative    :43   1st Qu.: 2.000
 P3:30    3:33                       Motivational:41   Median : 4.000
 P4:26    4:29                       Summative    :45   Mean   : 4.261
 P5:31    5:31                                          3rd Qu.: 6.000
                                                        Max.   :16.000
```

Summarising data by both factors *condition* and *category*

[17]:
```
# summarise by ~ condition + catagory
plyr::ddply(contentAnalysis, ~ condition  + category , function (data) {␣
 ↪summary (data$utterance) })
```

A data.frame: 8 Œ 8

condition <fct>	category <fct>	Min. <dbl>	1st Qu. <dbl>	Median <dbl>	Mean <dbl>	3rd Qu. <dbl>	Max. <dbl>
with	Affective	2	3.00	3.0	4.307692	6.00	9
with	Formative	1	3.00	5.0	6.300000	9.25	16
with	Motivational	1	1.75	2.5	3.350000	5.00	9
with	Summative	1	2.00	4.0	3.681818	4.00	10
without	Affective	1	1.00	2.0	1.933333	2.50	4
without	Formative	1	3.00	4.0	4.608696	6.00	9
without	Motivational	1	1.00	4.0	4.095238	6.00	10
without	Summative	1	3.50	5.0	5.130435	7.00	11

Distribution of the data

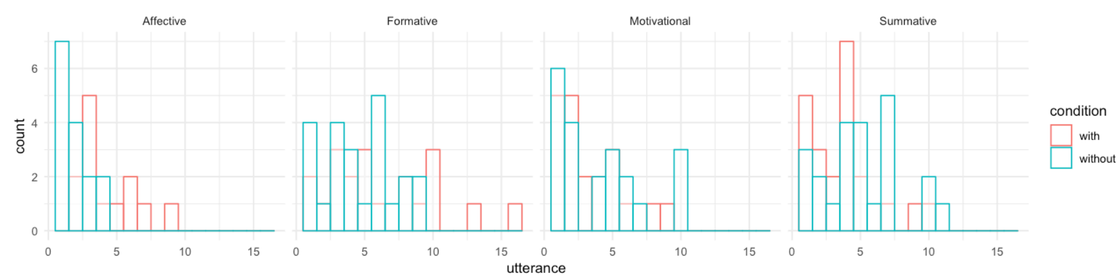
Here we plot the hist of each *catagory* under each *condition* to see how the distribution is.

As we can see below is not a normal distibution, but rather it tends towards a *poisson* one which is usually the case with count data (like, error rate etc.). We will verify this below.

[18]:
```
# Hist plots

options(repr.plot.width=12, repr.plot.height=3)

ggplot(contentAnalysis) +
geom_histogram(aes (x = utterance, color = condition),
               binwidth = 1,
               fill = "white",
               alpha = 0.5,
               position = "identity") +
facet_grid(~ category) +
theme_minimal()
```

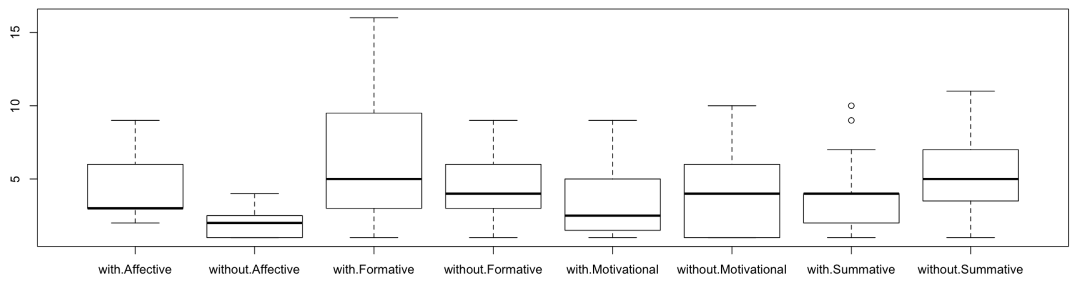


Boxplot of ~ condition + category

[19]:
```
# Boxplots

options(repr.plot.width=7, repr.plot.height=5)

boxplot( utterance ~ condition + category, data = contentAnalysis)
```



As we can see in the boxplots above:

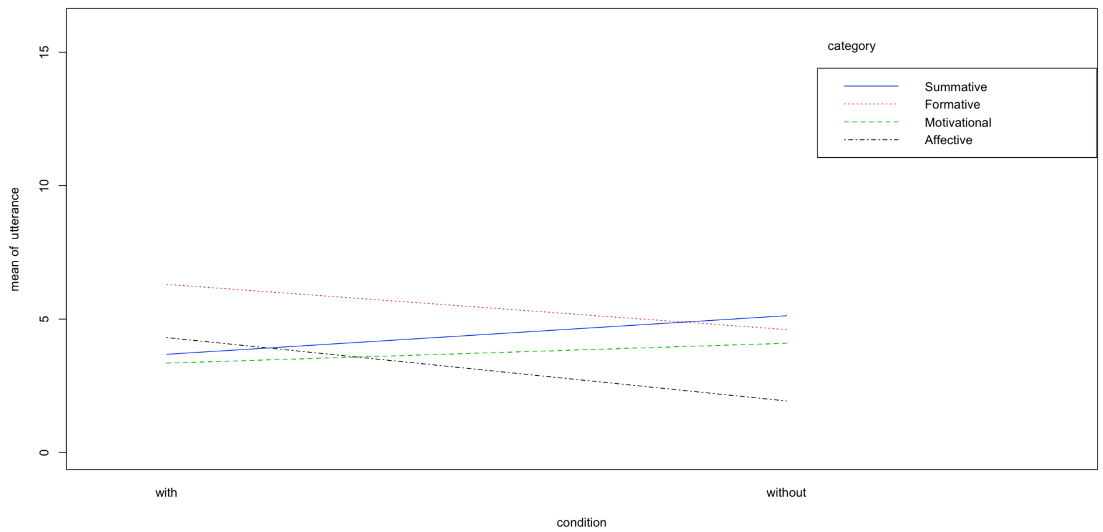
- We can see that there may be a difference in the Affective level which is increased with Emodash, and same thing with Formaive level.
- Also, there may be a difference in the Summative level which is decreased with Emodash.
- As we can see for the Motivational level, probably there is no difference as there is too much overlap in the box plot.

Interaction plot

[20]:
```
# interaction between categories

options(repr.plot.width=7, repr.plot.height=5)

with(contentAnalysis, interaction.plot( col = 1:4, leg.bty = "o", condition,␣
↪category, utterance, ylim=(c(0, max(contentAnalysis$utterance)))))
```



Interaction plot shows that there is an interaction between our variables (levels).
What is interesting is that:

- Without Emodash Formative and Summative are almost the same, but, they get different With Emodash where Formative increased and Summative decreased.
- Same thing with Affective and Motivational, where Affective level decreased without Emodash and Motivational increased.

Checking the distribution of the data

[21]:
```
# check if the data fit a poisson distribution as illustrated by histograms␣
↪above

fit = fitdist(contentAnalysis[contentAnalysis$condition == 'with' &␣
↪contentAnalysis$category == 'Affective', ]$utterance, 'pois', discrete=TRUE)
gofstat(fit)
```

```
Chi-squared statistic:  1.719688
Degree of freedom of the Chi-squared distribution:  1
Chi-squared p-value:  0.1897332
   the p-value may be wrong with some theoretical counts < 5
Chi-squared table:
     obscounts theocounts
<= 3  7.000000   4.885042
<= 6  4.000000   6.228297
> 6   2.000000   1.886661

Goodness-of-fit criteria
                               1-mle-pois
Akaike's Information Criterion   56.02720
Bayesian Information Criterion   56.59215
```

[22]:
```
fit = fitdist(contentAnalysis[contentAnalysis$condition == 'with' &␣
↪contentAnalysis$category == 'Motivational', ]$utterance, 'pois',␣
↪discrete=TRUE)
gofstat(fit)
```

```
Chi-squared statistic:  3.820542
Degree of freedom of the Chi-squared distribution:  3
Chi-squared p-value:  0.2815057
   the p-value may be wrong with some theoretical counts < 5
Chi-squared table:
     obscounts theocounts
<= 1  5.000000   3.052339
<= 2  5.000000   3.937342
<= 4  4.000000   8.078933
<= 5  3.000000   2.467097
```

```
> 5   3.000000   2.464289

Goodness-of-fit criteria
                               1-mle-pois
Akaike's Information Criterion   89.50635
Bayesian Information Criterion   90.50208
```

[23]:
```
fit = fitdist(contentAnalysis[contentAnalysis$condition == 'with' &␣
 ↪contentAnalysis$category == 'Formative', ]$utterance, 'pois', discrete=TRUE)
gofstat(fit)
```

```
Chi-squared statistic:  10.33606
Degree of freedom of the Chi-squared distribution:  4
Chi-squared p-value:  0.03513167
   the p-value may be wrong with some theoretical counts < 5
Chi-squared table:
      obscounts theocounts
<= 2  3.0000000  0.9969298
<= 3  3.0000000  1.5305416
<= 5  5.0000000  5.4479630
<= 8  3.0000000  8.3198984
<= 10 4.0000000  2.5790769
> 10  2.0000000  1.1255902

Goodness-of-fit criteria
                               1-mle-pois
Akaike's Information Criterion   122.0106
Bayesian Information Criterion   123.0063
```

[24]:
```
fit = fitdist(contentAnalysis[contentAnalysis$condition == 'with' &␣
 ↪contentAnalysis$category == 'Summative', ]$utterance, 'pois', discrete=TRUE)
gofstat(fit)
```

```
Chi-squared statistic:  5.858433
Degree of freedom of the Chi-squared distribution:  2
Chi-squared p-value:  0.0534389
   the p-value may be wrong with some theoretical counts < 5
Chi-squared table:
     obscounts theocounts
<= 1  5.000000   2.593247
<= 3  5.000000   8.361755
<= 4  7.000000   4.240992
> 4   5.000000   6.804006

Goodness-of-fit criteria
```

```
                              1-mle-pois
Akaike's Information Criterion   100.6796
Bayesian Information Criterion   101.7706
```

Same for without..

As we can see in result above, the p-values are not significant, except for Formative level but with high Chi-square, so, we dont have a significant departure from poisson distribution.

2 Modeling ~GLMM

[0]:
```
# Needed as we are using Anova from 'car' package not anova from default R
contrasts(contentAnalysis$condition) = 'contr.sum'
contrasts(contentAnalysis$category) = 'contr.sum'
contrasts(contentAnalysis$pair_session) = 'contr.sum'
```

[72]:
```
# Fit a generalized linear mixed-effects model (GLMM)
m = glmer(utterance ~ (condition + category)/pair_session + (1|pair) +␣
 ↪(1|pair_session), data=contentAnalysis, family = poisson, nAGQ = 1)
```

```
fixed-effect model matrix is rank deficient so dropping 5 columns / coefficients
boundary (singular) fit: see ?isSingular
```

[73]:
```
summary(m)
```

```
Correlation matrix not shown by default, as p = 40 > 12.
Use print(obj, correlation=TRUE)  or
    vcov(obj)        if you need it


Generalized linear mixed model fit by maximum likelihood (Laplace
  Approximation) [glmerMod]
 Family: poisson  ( log )
Formula: utterance ~ (condition + category)/pair_session + (1 | pair) +
    (1 | pair_session)
   Data: contentAnalysis

     AIC      BIC   logLik deviance df.resid
   738.9    867.3   -327.5    654.9      115

Scaled residuals:
    Min      1Q  Median      3Q     Max
-1.9119 -0.7084 -0.2507  0.5150  3.7010

Random effects:
 Groups       Name        Variance  Std.Dev.
```

```
 pair         (Intercept) 1.374e-01 3.707e-01
 pair_session (Intercept) 6.309e-10 2.512e-05
Number of obs: 157, groups:  pair, 5; pair_session, 5

Fixed effects:
                                                     Estimate Std. Error
(Intercept)                                         0.8142562  0.2628894
condition1                                         -0.3863492  0.1834301
category1                                          -0.6866038  0.2806818
category2                                           0.0184400  0.2099091
category3                                           0.1727391  0.2002580
conditionwith:categoryAffective:pair_session1       1.3576487  0.6001253
conditionwithout:categoryAffective:pair_session1    0.0119290  0.4853398
conditionwith:categoryFormative:pair_session1       1.2497449  0.4736337
conditionwithout:categoryFormative:pair_session1    0.2417132  0.3289092
conditionwith:categoryMotivational:pair_session1    0.6184279  0.4872520
conditionwithout:categoryMotivational:pair_session1 -0.1632045  0.3532960
conditionwith:categorySummative:pair_session1       0.4011697  0.3873563
conditionwithout:categorySummative:pair_session1   -0.4890892  0.3112707
conditionwith:categoryAffective:pair_session2       1.6200179  0.5806395
conditionwithout:categoryAffective:pair_session2   -0.5862826  1.0614653
conditionwith:categoryFormative:pair_session2       1.1012943  0.4794047
conditionwithout:categoryFormative:pair_session2    0.1876780  0.3333082
conditionwith:categoryMotivational:pair_session2    0.8060222  0.4878557
conditionwithout:categoryMotivational:pair_session2  0.0053822  0.3164523
conditionwith:categorySummative:pair_session2       0.6634761  0.3722195
conditionwithout:categorySummative:pair_session2   -0.1261746  0.2791499
conditionwith:categoryAffective:pair_session3       1.0009492  0.6347237
conditionwithout:categoryAffective:pair_session3    0.0582809  0.6129363
conditionwith:categoryFormative:pair_session3       1.4711993  0.4730436
conditionwithout:categoryFormative:pair_session3    0.4418910  0.3016077
conditionwith:categoryMotivational:pair_session3    0.3671715  0.5032210
conditionwithout:categoryMotivational:pair_session3 -0.0000965  0.3080828
conditionwith:categorySummative:pair_session3       0.3442245  0.3990605
conditionwithout:categorySummative:pair_session3    0.0666875  0.2579092
conditionwith:categoryAffective:pair_session4       2.0580426  0.5860530
conditionwithout:categoryAffective:pair_session4    0.2282473  0.5174483
conditionwith:categoryFormative:pair_session4       1.6534822  0.4795229
conditionwithout:categoryFormative:pair_session4    0.2452048  0.3141722
conditionwith:categoryMotivational:pair_session4    0.4775170  0.5411997
conditionwithout:categoryMotivational:pair_session4 -0.3338690  0.3616977
conditionwith:categorySummative:pair_session4      -0.2230479  0.4735172
conditionwithout:categorySummative:pair_session4   -0.4228237  0.2946527
conditionwith:categoryAffective:pair_session5       2.2121569  0.5758438
conditionwith:categoryFormative:pair_session5       1.2678637  0.4986997
conditionwith:categoryMotivational:pair_session5    0.3597488  0.5538421
                                                   z value Pr(>|z|)
(Intercept)                                          3.097 0.001953 **
```

```
condition1                                         -2.106 0.035183 *
category1                                          -2.446 0.014437 *
category2                                           0.088 0.929998
category3                                           0.863 0.388367
conditionwith:categoryAffective:pair_session1       2.262 0.023680 *
conditionwithout:categoryAffective:pair_session1    0.025 0.980391
conditionwith:categoryFormative:pair_session1       2.639 0.008324 **
conditionwithout:categoryFormative:pair_session1    0.735 0.462404
conditionwith:categoryMotivational:pair_session1    1.269 0.204364
conditionwithout:categoryMotivational:pair_session1 -0.462 0.644118
conditionwith:categorySummative:pair_session1       1.036 0.300360
conditionwithout:categorySummative:pair_session1   -1.571 0.116121
conditionwith:categoryAffective:pair_session2       2.790 0.005270 **
conditionwithout:categoryAffective:pair_session2   -0.552 0.580720
conditionwith:categoryFormative:pair_session2       2.297 0.021607 *
conditionwithout:categoryFormative:pair_session2    0.563 0.573383
conditionwith:categoryMotivational:pair_session2    1.652 0.098499 .
conditionwithout:categoryMotivational:pair_session2  0.017 0.986430
conditionwith:categorySummative:pair_session2       1.782 0.074670 .
conditionwithout:categorySummative:pair_session2   -0.452 0.651272
conditionwith:categoryAffective:pair_session3       1.577 0.114799
conditionwithout:categoryAffective:pair_session3    0.095 0.924248
conditionwith:categoryFormative:pair_session3       3.110 0.001870 **
conditionwithout:categoryFormative:pair_session3    1.465 0.142889
conditionwith:categoryMotivational:pair_session3    0.730 0.465609
conditionwithout:categoryMotivational:pair_session3  0.000 0.999750
conditionwith:categorySummative:pair_session3       0.863 0.388365
conditionwithout:categorySummative:pair_session3    0.259 0.795967
conditionwith:categoryAffective:pair_session4       3.512 0.000445 ***
conditionwithout:categoryAffective:pair_session4    0.441 0.659139
conditionwith:categoryFormative:pair_session4       3.448 0.000564 ***
conditionwithout:categoryFormative:pair_session4    0.780 0.435109
conditionwith:categoryMotivational:pair_session4    0.882 0.377598
conditionwithout:categoryMotivational:pair_session4 -0.923 0.355975
conditionwith:categorySummative:pair_session4      -0.471 0.637609
conditionwithout:categorySummative:pair_session4   -1.435 0.151290
conditionwith:categoryAffective:pair_session5       3.842 0.000122 ***
conditionwith:categoryFormative:pair_session5       2.542 0.011011 *
conditionwith:categoryMotivational:pair_session5    0.650 0.515982
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
fit warnings:
fixed-effect model matrix is rank deficient so dropping 5 columns / coefficients
convergence code: 0
boundary (singular) fit: see ?isSingular
```

3 Overall Significance of fixed effects ~Wald Chi-square test

[74]:
```
Anova(m, type=3)
```

A anova: 4 Œ 3

	Chisq <dbl>	Df <dbl>	Pr(>Chisq) <dbl>
(Intercept)	9.593476	1	0.001952699
condition	4.436280	1	0.035182823
category	9.375103	3	0.024697853
condition:category:pair_session	56.997166	35	0.010824354

The overall ANOVA (omnibus test) tells us that:

- There is a significant difference among the levels (with and without Emodash) of *condition* factor

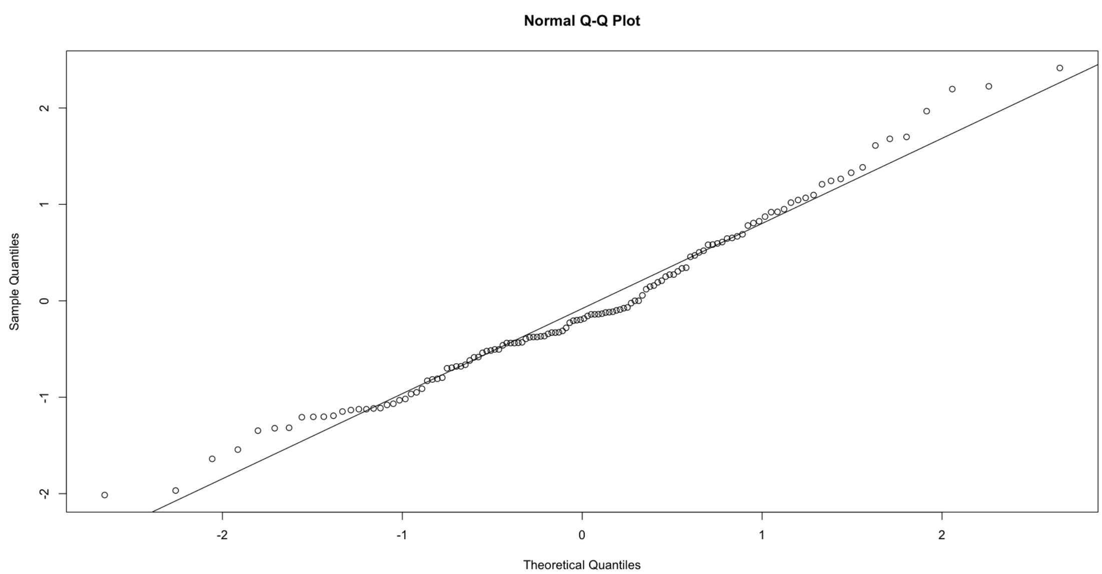
$$(\chi^2 = (1, N = 157) = 4.443, p < 0.05)$$

- There is a significant difference among the levels (Affective, Motivational, Summative, Formative) of *category* factor

$$(\chi^2 = (1, N = 157) = 9.381, p < 0.05)$$

4 QQNorm with residuals of the built model

[75]:
```
qqnorm(residuals(m));qqline(residuals(m))
```



As we can see in the result above, *qqnorm* with residuals shows that there is no departure from the normal.

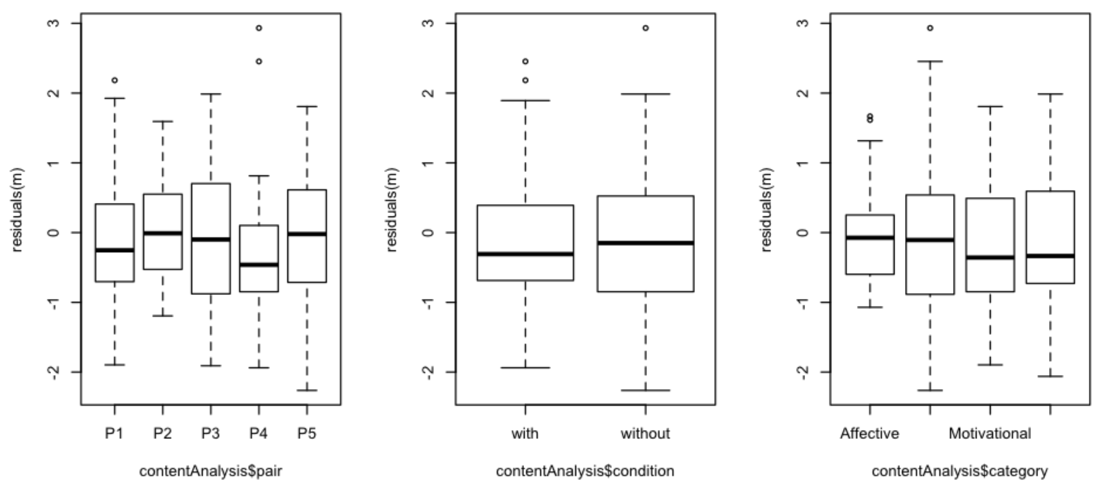
5 Homoskedasticity

GLMM's models assume that variance of the residuals is equal across groups. The main two ways to verify this is either graphically or statically using Levene's test.

Let's check the variance of residuals(m) of *Condition*, *Category*, and *Pair*

[76]:
```
par(mfrow=c(1,3))

boxplot(residuals(m) ~ contentAnalysis$pair)
boxplot(residuals(m) ~ contentAnalysis$condition)
boxplot(residuals(m) ~ contentAnalysis$category)
```



We can see that the variance is rather equal across oru groups. We can verify this statically:

[77]:
```
# pair

leveneTest(residuals(m) ~ contentAnalysis$pair)
```

A anova: 2 Œ 3

	Df	F value	Pr(>F)
	<int>	<dbl>	<dbl>
group	4	1.744336	0.1431132
	152	NA	NA

[78]:
```
# condition

leveneTest(residuals(m) ~ contentAnalysis$condition)
```

A anova: 2 Œ 3

	Df	F value	Pr(>F)
	<int>	<dbl>	<dbl>
group	1	0.924819	0.3377107
	155	NA	NA

[79]:
```
# category

leveneTest(residuals(m) ~ contentAnalysis$category)
```

A anova: 2 Œ 3

	Df <int>	F value <dbl>	Pr(>F) <dbl>
group	3	2.106086	0.1017665
	153	NA	NA

The three p-values are not significant (greater than 0.05), thus the variance of the residuals is equal and we conclude that the assumption of homoscedasticity is met.

6 Overdispersion

[0]:
```
#@title Helper
# from lme4's creator to approximate overdispersion
# https://bbolker.github.io/mixedmodels-misc/glmmFAQ.html#overdispersion
overdisp_fun <- function(model) {
    rdf <- df.residual(model)
    rp <- residuals(model,type="pearson")
    Pearson.chisq <- sum(rp^2)
    prat <- Pearson.chisq/rdf
    pval <- pchisq(Pearson.chisq, df=rdf, lower.tail=FALSE)
    c(chisq=Pearson.chisq,ratio=prat,rdf=rdf,p=pval)
}
```

[99]:
```
contentAnalysis$obs_effect<-1:nrow(contentAnalysis)

head(contentAnalysis)
```

A tibble: 6 Œ 6

pair <fct>	pair_session <fct>	condition <fct>	category <fct>	utterance <int>	obs_effect <int>
P1	1	with	Affective	3	1
P1	1	with	Formative	4	2
P1	1	with	Motivational	2	3
P1	1	with	Summative	4	4
P1	1	without	Affective	1	5
P1	1	without	Formative	6	6

[100]:
```
# Fit a generalized linear mixed-effects model (GLMM)
mm = glmer(utterance ~ (condition + category)/pair_session + (1|pair) +␣
↪(1|obs_effect), data=contentAnalysis, family = poisson, nAGQ = 1)
```

```
fixed-effect model matrix is rank deficient so dropping 5 columns / coefficients
Warning message in checkConv(attr(opt, "derivs"), opt$par, ctrl =
control$checkConv, :
Model failed to converge with max|grad| = 0.00293687 (tol = 0.001, component
```

[101]:
```
Anova(mm, type = '3')
```

A anova: 4 Œ 3

	Chisq <dbl>	Df <dbl>	Pr(>Chisq) <dbl>
(Intercept)	9.110341	1	0.002541684
condition	4.218324	1	0.039989650
category	8.861339	3	0.031192418
condition:category:pair_session	52.953796	35	0.026354632

[102]:
```
overdisp_fun(mm)
```

chisq 131.886817307352 **ratio** 1.14684188962915 **rdf** 115 **p** 0.134215230818383

#Post Hoc Test

Now we run a pairwise comparison using Holm-Bonferroni for adjustments.

[90]:
```
# Get pairwise summary

emmeans(mm, list(pairwise ~ condition | category), adjust = "holm")
```

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

$`emmeans of condition | category`
category = Affective:
 condition emmean    SE  df asymp.LCL asymp.UCL
 with       1.386 0.222 Inf     0.951      1.82
 without    0.451 0.307 Inf    -0.151      1.05

category = Formative:
 condition emmean    SE  df asymp.LCL asymp.UCL
 with       1.787 0.193 Inf     1.409      2.17
 without    1.436 0.196 Inf     1.053      1.82

category = Motivational:
 condition emmean    SE  df asymp.LCL asymp.UCL
 with       1.116 0.214 Inf     0.698      1.54
 without    1.262 0.204 Inf     0.863      1.66

category = Summative:
 condition emmean    SE  df asymp.LCL asymp.UCL
 with       1.153 0.208 Inf     0.745      1.56
 without    1.494 0.194 Inf     1.113      1.87

Results are averaged over the levels of: pair_session
Results are given on the log (not the response) scale.
Confidence level used: 0.95

$`pairwise differences of condition | category`
category = Affective:
 contrast       estimate    SE  df z.ratio p.value
 with - without    0.934 0.294 Inf   3.175  0.0015

category = Formative:
```