

# Cognition Paper: Ex. 2: Intensity Ratings

## Hypothesis

- $H_0$ : We hypothesize that there is no difference between thick positive and thick negative thick terms for our two dependent variables.
  - $\mu_{behaviour[positive]} - \mu_{behaviour[negative]} = 0$
  - $\mu_{sentence[positive]} - \mu_{sentence[negative]} = 0$

Since we work with mixed-measures ANOVA, we use our two DVs (question:behaviour, question:sentence) as a between-subject IV in interaction with valence, which will be a within-subject variable.

## Testing

### ANOVA assumptions

The assumption of homogeneity of variance of the between-subject factor (question) was checked using the Levene's test. The test is performed at each level of valence variable. The assumption is not violated, since none of the results are significant on a .01-alpha level.

```
df %>%
  dplyr::group_by(valence) %>%
  rstatix::levene_test(value ~ question)

## # A tibble: 2 x 5
##   valence    df1    df2 statistic      p
##   <fct>    <int> <int>      <dbl> <dbl>
## 1 positive      1    616      0.624 0.430
## 2 negative      1    616      5.06 0.0249
```

The data is not normally distributed, on neither of the interaction levels. Thus, the assumption is violated. Since the assumption of homogeneity of variance still holds, we can still use parametric methods and proceed with our mixed-measure ANOVA.

```
df %>%
  dplyr::group_by(valence, question) %>%
  dplyr::summarise(statistic = shapiro.test(value)$statistic,
    p.value = shapiro.test(value)$p.value)

## `summarise()` regrouping output by 'valence' (override with `.groups` argument)
## # A tibble: 4 x 4
## # Groups:   valence [2]
##   valence question statistic p.value
##   <fct>    <fct>      <dbl>   <dbl>
## 1 positive sentence    0.811 1.35e-18
## 2 positive behaviour    0.845 5.24e-17
## 3 negative sentence    0.852 1.76e-16
## 4 negative behaviour    0.897 9.77e-14
```

## The Hypothesis

- Global 2x2 mixed-measure ANOVA:
  - Y: agreement value
  - within-subject variable: valence ( $= id/valence; .. := \{positive, negative\}$ )
  - between-subject variable: question ( $:= \{behaviour, sentence\}$ )

In order to test our hypothesis, we use the model specified above. The results are shown below.

```
summary(aov(value ~ question*valence + Error(id/valence), data=df))
```

```
##
## Error: id
##           Df Sum Sq Mean Sq
## question  1  27.27    27.27
##
## Error: id:valence
##           Df Sum Sq Mean Sq
## valence   1  92.39    92.39
##
## Error: Within
##           Df Sum Sq Mean Sq F value Pr(>F)
## question    1     2    2.27   0.406 0.52398
## valence     1    38   38.05   6.817 0.00914 **
## question:valence 1     1    1.07   0.192 0.66095
## Residuals 1230  6864    5.58
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

As we can see, valence is significant on .01-alpha level, i.e. the means of positive and negative thick terms differ significantly. **Thus, we have to reject our hypothesis.** The interaction `question*valence` is not significant, on the other hand. This means that valence has a significant effect across the two levels of the between-subject variable.

In order to have a more detailed estimate of our effect, we performed a post-hoc test (Tukey's HSD with Bonferroni-correction) for pairwise contrasts. Note that `question*valence` has been combined into a single interaction variable, i.e. `questionPol`, since the `multcomp::glht`-function cannot handle the traditional formula. The Tukey Contrasts show that the valence effect within each of the questions:

- sentence.negative - sentence.positive == 0
- behaviour.negative - behaviour.positive == 0

This means that within both DVs (question:behaviour, question:sentence) the valence effect is significant.

```
m1 <- nlme::lme(value ~ questionValence, random = ~ 1 | id, method="ML", data=df)
summary(multcomp::glht(m1, linfct = multcomp::mcp(questionValence = "Tukey")),
       test = adjusted(type = "bonferroni"))
```

```
##
## Simultaneous Tests for General Linear Hypotheses
##
## Multiple Comparisons of Means: Tukey Contrasts
##
##
## Fit: lme.formula(fixed = value ~ questionValence, data = df, random = ~1 |
##       id, method = "ML")
##
## Linear Hypotheses:
```

```

##                                Estimate Std. Error z value
## behaviour.positive - sentence.positive == 0    0.007856   0.344621   0.023
## sentence.negative - sentence.positive == 0   -0.705882   0.140563  -5.022
## behaviour.negative - sentence.positive == 0   -0.585093   0.344621  -1.698
## sentence.negative - behaviour.positive == 0   -0.713738   0.344621  -2.071
## behaviour.negative - behaviour.positive == 0  -0.592949   0.139204  -4.260
## behaviour.negative - sentence.negative == 0    0.120789   0.344621   0.350
##                                Pr(>|z|)
## behaviour.positive - sentence.positive == 0    1.000000
## sentence.negative - sentence.positive == 0    3.07e-06 ***
## behaviour.negative - sentence.positive == 0    0.537289
## sentence.negative - behaviour.positive == 0    0.230108
## behaviour.negative - behaviour.positive == 0    0.000123 ***
## behaviour.negative - sentence.negative == 0    1.000000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## (Adjusted p values reported -- bonferroni method)

```