Statistical Analysis of Eye Tracking Data

Krzysztof Krejtz

SWPS UNIVERSITY

LEAD-ME Summer Training School Warsaw 2021 (5-9 July 2021)
Eye tracking in media accessibility research - methods, technologies and data analyses

1

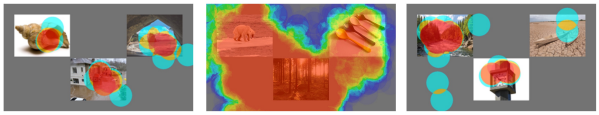
Agenda

- R language & R-Studio IDE basics
- Moderation analysis in eye tracking studies
  - Mixed-design Analysis of Variance (ANOVA) with interaction effect
  - Pairwise comparisons
  - Graphical representations of interaction effects

SWPS UNIVERSITY

2

Climate change overlooked. The role of attitudes and mood regulation in visual attention to global warming



Anna Mazurowska
SWPS University of Social Sciences and Humanities, Warsaw, Poland

Research Advisor:
dr. Krzysztof Krejtz

OBO

SWPS UNIVERSITY

2021 Symposium on Eye Tracking Research and Applications (ETRA '21), May 25–27

3

Hypothesis

Individuals with **environmental concerns** will be more likely to place visual attention on images depicting **negative consequences of climate change** than individuals without such concerns.

SWPS UNIVERSITY

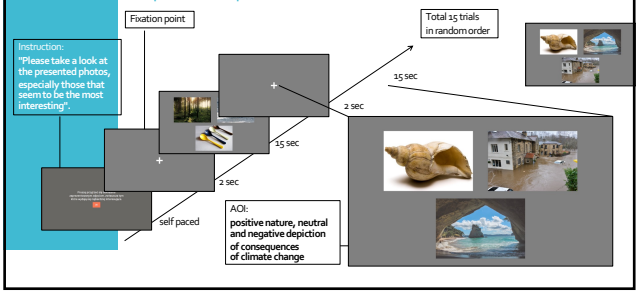
4

Method

- Participants:
  - students of psychology at SWPS University (N = 48)
- Two stage online study:
  - Stage 1: webcam-based eye tracking experiment with RealEye
  - Stage 2: New Ecological Paradigm Scale (NEP) – measure of pro-ecological attitude (sensitivity to climate change)

5

Experimantal procedure

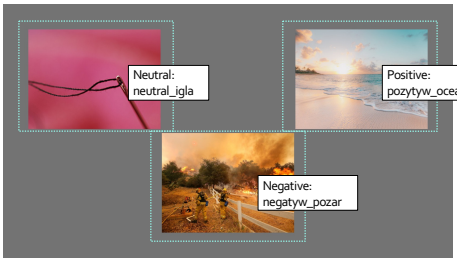


6

Exemplary recording from RealEye

7

Areas of Interest (AOI) definition



8

Questionnaire

Pro-environmental attitude measurement:

New Ecological Paradigm Scale (Dunlap et al. 2000), e.g.:

- "We are approaching the limit of the number of people the Earth can support".
- "Plants and animals have as much right as humans to exist".

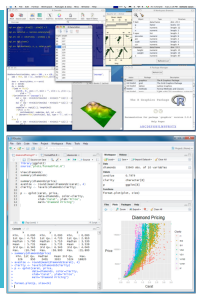
	Strongly disagree	Disagree	Neither agree nor disagree	Agree	Strongly agree
We are approaching the limit of the number of people the Earth can support	○	○	○	○	◉
Plants and animals have as much right as humans to exist	○	○	○	◉	○

9

R & R Studio Basics

10

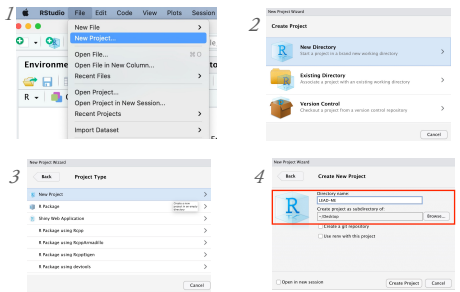
R language and R Studio IDE



- R is a free software environment for statistical computing and graphics with large worldwide community.
  - http://www.r-project.org/
  - R Core Team (2015). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria.
- RStudio is a free and open source Integrated Development Environment (IDE) for R.
  - http://www.rstudio.com/ide/

11

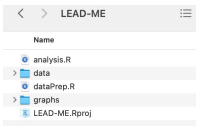
Start a new project in 4 steps



12

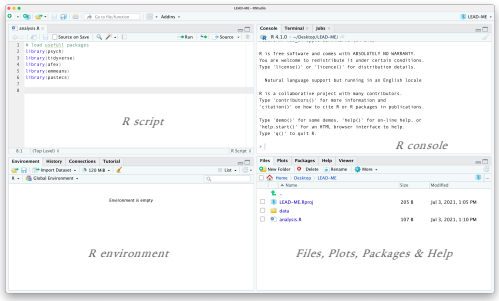
Outside RStudio

- To keep everything in place create two folders in your working directory:
  - 'data'
  - 'graphs'



13

Complete RStudio IDE interface, ready for statistical analyses



14

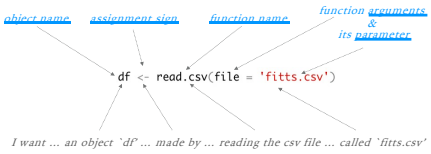
Talking in R

```
lunch <- make.pizza(dough = "thin",
                    sauce = "tomato",
                    main.ingredient =  "mozzarella",
                    doubleCheese = FALSE)
```

15

Talking in R

Ordering pizza (*oops...* analyses) with R



16

libraries
functions
parameters

Libraries (packages)
- All functions are stored in packages.
- Load packages to R memory to make its functions available
- There are packages for everything – you can download them from Internet

Functions
- Functions names are described in packages' help files

Parameters
- All you will find in the 'help' files

17

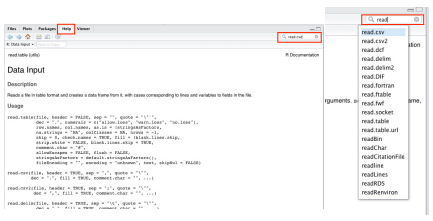
libraries
functions
parameters

```
library(psych)

describe(x, na.rm = TRUE, interp=FALSE,skew = TRUE, ranges = TRUE, trim=.1,
type=3, check=TRUE, fast=NULL, quant=NULL,IQR=FALSE)
```

18

Using help

Searching for functions' help: definition, arguments, examples of usage

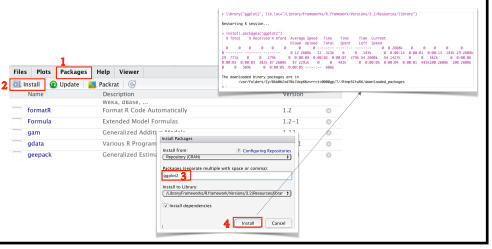


19

Installing r libraries (packages)

```
install.packages(pkgs = "ggplot2", dependencies = TRUE)
```

if TRUE will install also all dependent packages



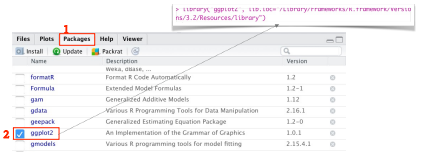
20

Loading installed libraries (packages)

Before using functions from the library we have to load them to the working memory.

```
library(ggplot2)
```

```
> library(ggplot2)
>
```



21

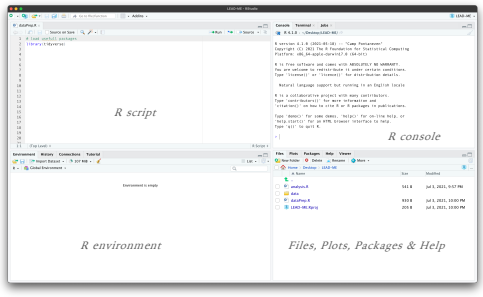
Loading large libraries

- Some libraries contain other useful packages e.g., tidyverse

```
> library(tidyverse)
── Attaching packages ─────────────────────────────────────── tidyverse 1.3.1 ──
✓ ggplot2 3.3.4     ✓ purrr   0.3.4
✓ tibble  3.1.2     ✓ dplyr   1.0.6
✓ tidyr   1.1.3     ✓ stringr 1.4.0
✓ readr   1.4.0     ✓ forcats 0.5.1
```

22

Complete RStudio IDE ready for statistical analyses

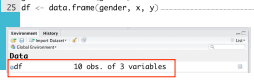


23

Data frames

- Data frame is a quadratic table of data.
  - In data frame different columns can have different classes (numeric, character, factor, etc.).
  - All variables have to have the same length.
  - This is similar to SPSS datasets.

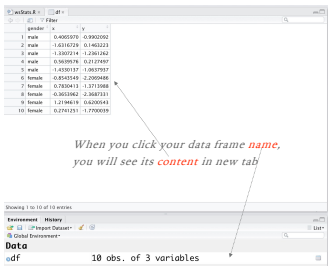
```
# Data frames
x <- rnorm(10)                                  # numeric vector of random 10 values
y <- rnorm(10)                                  # numeric vector of random 10 values
gender <- c(rep("male", 5), rep("female", 5))   # character vector of 5 „male" and 5 „female" values
gender <- factor(gender)                        # factor of gender
df <- data.frame(gender, x, y)                  # data frame with 3 variables: gender, x, and y
```



When done you'll see your data frame in Environment tab + information about its dimensions

24

Data frames



25

Data frames & variables

A set of useful functions (good to remember all of them)

```
27 class(df)
28 dim(df)
29 names(df)
30 str(df)
31 head(df)
```

information about the object class
dimensions of data frame or matrix
names of variables in the data frame
structure of data frame and all its variables
displays 6 first raws of data frame

```
> names(df)
[1] "gender" "x"      "y"
```

```
> head(df)
  gender          x          y
1   male  0.4065970 -0.9902092
2   male -1.6316729  0.1463223
3   male -1.3307214 -1.2361262
4   male  0.5639576  0.2127497
5   male -1.4330137 -1.0637937
6 female -0.8543549 -2.2069486
```

```
> str(df)
'data.frame':   10 obs. of  3 variables:
 $ gender: Factor w/ 2 levels "female","male": 2 2 2 2 2 1 1 1 1 1
 $ x     : num  0.407 -1.632 -1.331 0.564 -1.433 ...
 $ y     : num  -0.99 0.146 -1.236 0.213 -1.064 ...
```

26

Data frames

Calling the variable

- Calling/using the specific variable from the data frame can be done in two ways:
  - referring the name of the variable

    dataFrame$variableName

```
36 df$gender
```

```
> df$gender
 [1] male   male   male   male   male   female female female female female
Levels: female male
```

  - referring the index (number)of the column which stores the variable in the data frame
    - data.frames are tables so rows and columns have indices

      dataFrame[row, column]

```
38 df[ ,1]
```

```
> df[ ,1]
 [1] male   male   male   male   male   female female female female female
Levels: female male
```

27

Objects in R

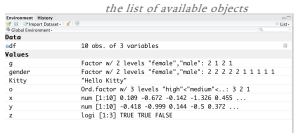
- Objects are ***bricks***, which you can use for the analyses (data frames, variables, statistics results, functions, plots, etc.)
- Objects are stored temporarily in the R environment
- Objects can be loaded, eg. data frames or created by R

create the object

```
x <- rnorm(10)
```

object name — assignment sign — function defining object

the list of available objects



remove the object

```
rm(x)
```

28

Moderation analysis with ANOVA

Mixed-design Analysis of Variance (ANOVA)

29

Basic concept of ANOVA

- ANalysis Of VAriance
- Generalizes t-tests to more than two groups.
- Testing variation among the means of several groups
  - Provides a statistical test of whether or not the means of several groups are all equal, or is there a difference between them?

30

Basic concept of ANOVA

- Null hypothesis
  - Several populations have the same means

$$\mu_0 = \mu_1 = \mu_2$$

- Test hypothesis

$$\mu_0 \neq \mu_1 \neq \mu_2$$

31

Basic concept of ANOVA

- Between group variability
  - Differences between group means
- Within group variability (error variability)
  - Differences between each score in the sample and the sample mean

32

Sources of variability

- Between group variance can come from:
  - The effect of Independent Variable 😊
    - Differences between group means – our expected effect
  - Individual differences 😐
    - Everyone is different, different motivations, reactions to tasks, etc.
  - Measurement error 😐
    - Nobody's perfect – our studies neither, non standardised instructions

33

Within-group variance (error variance)

- Variation of results within groups or conditions.
- Due to:
  - Measurement error
  - Individual differences

G1	G2	G3
1,2,3,3,3,4	3,3,3,3,3,3	5,6,2,1,7,9

One of ANOVA assumptions is the similarity of variances among the groups.

34

F - ratio

- We compare the two sources of scores' variations using statistical F test.
- We can check whether the observed differences are due to sampling error/chance or our Independent Variable.
- Ratio of the between-groups variance (explained variability)to the within-groups variance (unexplained variability)

$$\text{statistic} = \frac{\text{between-groups variability}}{\text{error variability}}$$

- When the F is below or close to 1 ...
- Reducing the error variability in the denominator of the equation will result in a larger computed statistical value, thereby making it easier to reject the Null hypothesis.

35

Usefull libraries for ANOVA analysis

```
library(tidyverse)
library(afex)
library(emmeans)
```

```
> library(tidyverse)
── Attaching packages ─────────────────────────────────────
──────── tidyverse 1.3.0 ──
✓ ggplot2 3.3.3     ✓ purrr   0.3.4
✓ tibble  3.0.5     ✓ dplyr   1.0.3
✓ tidyr   1.1.2     ✓ stringr 1.4.0
✓ readr   1.4.0     ✓ forcats 0.5.0
```

36

Load data

```
# read data frame from RData format
load("data/all_data.Rda")
```

```
> names(d)
 [1] "ID"
 [2] "aoi_type"
 [3] "aoi_size_percents"
 [4] "aoi_fixation_total_count"
 [5] "aoi_fixation_average_duration_ms"
 [6] "aoi_fixation_ttff_ms"
 [7] "aoi_fixation_average_total_time_spent_ms"
 [8] "aoi_fixation_first_fixation_average_duration_ms"
 [9] "NEP"
[10] "NEPsplit"
```

ID	aoi_type	aoi_size_percents	aoi_fixation_total_count	aoi_fixation_average_duration_ms
63148	negatyw	13.51776	11	155
63148	neutral	13.51776	0	0
63148	pozytyw	13.48463	2	149
45940	negatyw	13.51776	13	192
45940	neutral	13.51776	3	150
45940	pozytyw	13.48463	12	173
54370	negatyw	13.51776	10	145
54370	neutral	13.51776	2	137

37

Simple function for ANOVA in R

- To run ANOVA in R you have plenty of options.
- We are going to use the one **aov_ez()** function within the **afex package** (mainly for its simplicity).

```
library(afex)
```

afex package provides convenience functions for analyzing factorial experiments using ANOVA or mixed models. aov_ez(), aov_car(), and aov_4() allow specification of between, within, or mixed ANOVAs for data in long format.

- Generic form of **aov_ez()** function

```
aov_ez(id = "PARTICIPANTS ID VARIABLE",
       dv = "DEPENDENT VARIABLE",
       data = "DATA FRAME",
       between = "BETWEEN-SUBJECTS FACTOR",
       within = "WITHIN-SUBJECTS FACTOR",
       covariate = "BETWEEN-SUBJECTS COVARIATES",
       type = "ERROR TYPE - DEFAULT SET TO 3")
```

38

Test the hypotheses about fixation duration

- We want to test two hypotheses:
  - In general, negative and positive pictures of environment will evoke longer fixation durations than a neutral one
  - Negative and positive pictures of environment will evoke longer fixation durations than a neutral one BUT ONLY for people with high sensitivity to climate change
    - *sensitivity to climate change will be a moderator for the effect of picture valence*

39

Run two-way mixed-design ANOVA

RUN ANOVA TEST

```
# fixation count dependent on aoi valence and NEP score
fit <- aov_ez(data = d,
              id = "ID",
              dv = "aoi_fixation_average_duration_ms",
              within = "aoi_type",
              between = "NEPsplit")
```

Data
Subject ID variable
Dependent variable
Within-subject factor
Between-subject factor

PRINT ANOVA TEST RESULTS

```
> print(fit)
Anova Table (Type 3 tests)

Response: aoi_fixation_average_duration_ms
             Effect          df     MSE         F  ges p.value
1          NEPsplit       1, 42 1300.67      0.39 .005    .535
2          aoi_type 1.83, 76.89  546.00 10.54 *** .098   <.001
3 NEPsplit:aoi_type 1.83, 76.89  546.00    2.59 + .026    .086
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '+' 0.1 ' ' 1

Sphericity correction method: GG
```

40

Estimated means and pairwise comparisons for main effect

```
em <- emmeans(fit, ~aoi_type)
```

```
> print(em)
 aoi_type emmean   SE  df lower.CL upper.CL
 neutral     127 4.19 102      118      135
 negatyw     141 4.19 102      133      150
 pozytyw     148 4.19 102      140      157

Results are averaged over the levels of: NEPsplit
Warning: EMMs are biased unless design is perfectly balanced
Confidence level used: 0.95
```

```
# post-hoc for main effect
pairs(em)
```

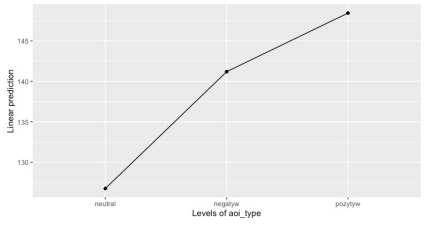
```
> pairs(em)
 contrast          estimate   SE df t.ratio p.value
 neutral - negatyw   -14.45 4.81 84 -3.003  0.0098
 neutral - pozytyw   -21.70 4.81 84 -4.510  0.0001
 negatyw - pozytyw    -7.25 4.81 84 -1.506  0.2931

Results are averaged over the levels of: NEPsplit
P value adjustment: tukey method for comparing a family of 3 estimates
```

41

Analytical graph of main effect

(not suitable for publication)

```
emmip(fit, ~ aoi_type)
```



42

Estimated means for interaction effect

```
em <- emmeans(fit, ~aoi_type:NEPsplit)
```

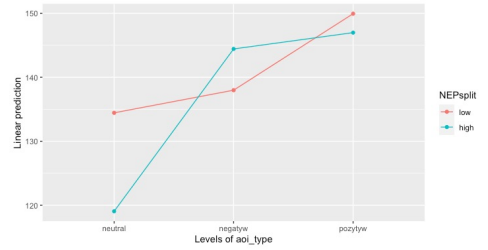
```
> print(em)
 aoi_type NEPsplit emmean   SE    df lower.CL upper.CL
 neutral  low         134 6.12 107.1      122      147
 negatyw  low         138 6.12 107.1      126      150
 pozytyw  low         150 6.12 107.1      138      162
 neutral  high        119 5.76  95.7      108      131
 negatyw  high        144 5.76  95.7      133      156
 pozytyw  high        147 5.76  95.7      136      158
```

43

Analytical graph of interaction effect

(not suitable for publication)

```
emmip(fit, NEPsplit ~ aoi_type)
```



44

Pairwise comparisons for interaction effect

```
pairs(em, by = "NEPsplit")
```

```
> pairs(em, by = "NEPsplit")
NEPsplit = low:
 contrast          estimate   SE df t.ratio p.value
 neutral - negatyw    -3.54 7.25 84 -0.488  0.8774
 neutral - pozytyw   -15.48 7.25 84 -2.135  0.0890
 negatyw - pozytyw   -11.95 7.25 84 -1.647  0.2318

NEPsplit = high:
 contrast          estimate   SE df t.ratio p.value
 neutral - negatyw   -25.36 6.32 84 -4.011  0.0004
 neutral - pozytyw   -27.91 6.32 84 -4.414  0.0001
 negatyw - pozytyw    -2.55 6.32 84 -0.403  0.9144

P value adjustment: tukey method for comparing a family of 3 estimates
```

```
pairs(em, by = "aoi_type")
```

```
> pairs(em, by = "aoi_type")
aoi_type = neutral:
 contrast   estimate   SE  df t.ratio p.value
 low - high   15.38 8.43 101  1.825  0.0709

aoi_type = negatyw:
 contrast   estimate   SE  df t.ratio p.value
 low - high    -6.44 8.43 101 -0.765  0.4463

aoi_type = pozytyw:
 contrast   estimate   SE  df t.ratio p.value
 low - high     2.95 8.43 101  0.350  0.7267
```

45

Publication ready plot of interaction effect

```
tab <- summary(em)
```

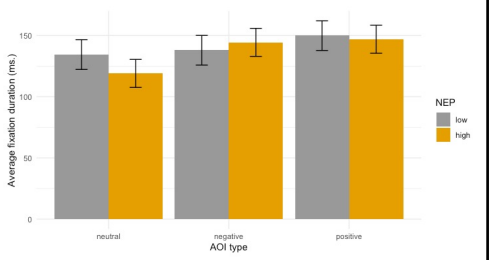
```
> print(tab)
 aoi_type NEPsplit emmean   SE    df lower.CL upper.CL
 neutral  low         134 6.12 107.1      122      147
 negatyw  low         138 6.12 107.1      126      150
 pozytyw  low         150 6.12 107.1      138      162
 neutral  high        119 5.76  95.7      108      131
 negatyw  high        144 5.76  95.7      133      156
 pozytyw  high        147 5.76  95.7      136      158
```

```
plo <- ggplot(tab, aes(x=aoi_type, y=emmean, fill=NEPsplit)) +
  geom_bar(position=position_dodge(), stat="identity") +
  geom_errorbar(aes(ymin=lower.CL, ymax=upper.CL),
                width=.2, position=position_dodge(.9)) +
  scale_y_continuous(name = "Average fixation duration (ms.)") +
  scale_x_discrete(name = "AOI type",
                   labels = c("neutral", "negative", "positive")) +
  scale_fill_manual(name = "NEP",
                    values=c("#999999", "#E69F00")) +
  theme_minimal()
```

46

Publication ready plot of interaction effect

(bargraph with error bars representing confidence intervals)



47

Save the graph to the file

Saving the graph to the file with a code.
You can choose a wide variety of formats (jpg, png, tiff, pdf, eps, etc.)

```
print(plo)
ggsave(plot = plo, filename = "graphs/AverageFixationDuration_NEP_AOI.jpg")
dev.off()
```

48

Test hypotheses about fixation count

- We want to test two hypotheses:
  - In general, negative and positive pictures of environment will evoke more fixations than a neutral one
  - Negative and positive pictures of environment will evoke more fixations than a neutral one BUT ONLY for people with high sensitivity to climate change
    - *sensitivity to climate change will be a moderator for the effect of picture valence*

49

Run two-way mixed-design ANOVA

RUN ANOVA TEST

```
fit <- aov_ez(data = d,
              id = "ID",
              dv = "aoi_fixation_total_count",
              within = "aoi_type",
              between = "NEPsplit")
```

Data
Subject ID variable
Dependent variable
Within-subject factor
Between-subject factor

PRINT ANOVA TEST RESULTS

```
> print(fit)
Anova Table (Type 3 tests)

Response: aoi_fixation_total_count
            Effect          df   MSE         F  ges p.value
1          NEPsplit       1, 42 12.08      0.57 .005    .455
2          aoi_type 1.76, 74.09 12.19 23.08 *** .260   <.001
3 NEPsplit:aoi_type 1.76, 74.09 12.19      0.67 .010    .497
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '+' 0.1 ' ' 1

Sphericity correction method: GG
```

50

Estimated means and pairwise comparisons for main effect

```
em <- emmeans(fit, ~aoi_type)
```

```
> print(em)
 aoi_type emmean    SE  df lower.CL upper.CL
 neutral     4.19 0.507 126     3.18     5.19
 negatyw     8.16 0.507 126     7.15     9.16
 pozytyw     8.50 0.507 126     7.50     9.50
```

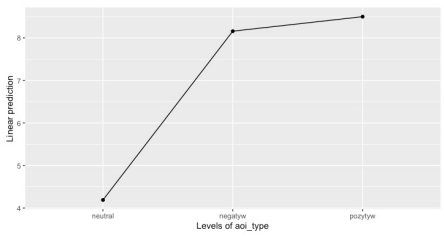
```
# post-hoc for main effect
pairs(em)
```

```
> pairs(em)
 contrast          estimate    SE df t.ratio p.value
 neutral - negatyw   -3.971 0.706 84 -5.626  <.0001
 neutral - pozytyw   -4.312 0.706 84 -6.110  <.0001
 negatyw - pozytyw   -0.341 0.706 84 -0.484  0.8792

Results are averaged over the levels of: NEPsplit
P value adjustment: tukey method for comparing a family of 3 estimates
```

51

Analytical graph of main effect

```
emmip(fit, ~ aoi_type)
```



52

Publication ready plot of main effect

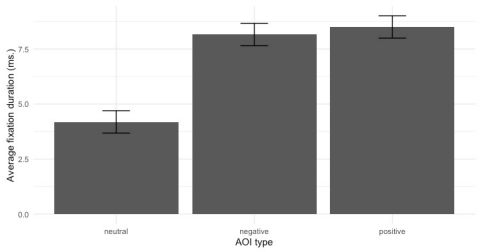
```
tab <- summary(em)
```

```
> print(tab)
 aoi_type emmean    SE  df lower.CL upper.CL
 neutral    4.19 0.507 126     3.18     5.19
 negatyw    8.16 0.507 126     7.15     9.16
 pozytyw    8.50 0.507 126     7.50     9.50
```

```
plo <- ggplot(tab, aes(x=aoi_type, y=emmean)) +
  geom_bar(position=position_dodge(), stat="identity") +
  geom_errorbar(aes(ymin=emmean-SE, ymax=emmean+SE),
                width=.2, position=position_dodge(.9)) +
  scale_y_continuous(name = "Average fixation duration (ms.)") +
  scale_x_discrete(name = "AOI type",
                   labels = c("neutral", "negative", "positive")) +
  theme_minimal()
```

53

Publication ready plot of interaction effect



54

Save the graph to the file

Saving the graph to the file with a code.
You can choose a wide variety of formats (jpg, png, tiff, pdf, eps, etc.)

```
print(plo)
ggsave(plot = plo, filename = "graphs/fixationCount_AOI.jpg")
dev.off()
```

55

Thank you!

Any questions?

56