

Project Test

Contents

Load this:

```
## Load libraries
if(!require(tidyverse)) {install.packages("tidyverse"); require(tidyverse)}
#if(!require(rstudioapi)) {install.packages("rstudioapi"); require(rstudioapi)}
if(!require(MPpsychR)) {install.packages("MPpsychR"); require(MPpsychR)}
if(!require(sjPlot)) {install.packages("sjPlot"); require(sjPlot)}
if(!require(lme4)) {install.packages("lme4"); require(lme4)}
if(!require(lmerTest)) {install.packages("lmerTest"); require(lmerTest)}
if(!require(languageR)) {install.packages("languageR"); require(languageR)}
## Set plotting preferences
ggplot2::theme_set(ggplot2::theme_bw(base_size=18))
ggplot2::theme_update(panel.grid = ggplot2::element_blank(),
                      strip.background = ggplot2::element_blank(),
                      legend.key = ggplot2::element_blank(),
                      panel.border = ggplot2::element_blank(),
                      axis.line = ggplot2::element_line(),
                      strip.text = ggplot2::element_text(face = "bold"),
                      plot.title = element_text(hjust = 0.5))
options(ggplot2.discrete.colour= c("#A31F34", "#8A8B8C"))

## Set function defaults
filter <- dplyr::filter
group_by <- dplyr::group_by
summarize <- dplyr::summarize
select <- dplyr::select

## Set working directory
#setwd(dirname(rstudioapi::getActiveDocumentContext())$`path`))
```

First, load the data and filter out the filler conditions:

```
## Load data
data <- read.csv("practice_data.csv", header = TRUE, sep = ",")

#Remove cond filler
data <- data %>% filter(cond != "filler")
#Remove conds tha contain active or passive
data <- data %>% filter(!grepl("active", cond))
data <- data %>% filter(!grepl("passive", cond))
```

Group data two different ways: - cond that contains “do” or “po” - cond that contains “implausible” or “plausible”

```
#Create variable for cond that contains "do" or "po"
data <- data %>%
  mutate(do_po = ifelse(grepl("DO", cond), "DO", "PO")) %>% #Create factor
  mutate(do_po = factor(do_po, levels = c("DO", "PO"))) #Set factor levels
#Create variable for cond that contains "implausible" or "plausible"
data <- data %>%
  mutate(plausibility = ifelse(grepl("implausible", cond), "implausible", "plausible")) %>% #Create factor
  mutate(plausibility = factor(plausibility, levels = c("implausible", "plausible"))) #Set factor levels

#Response is literal if literal_response = response
data <- data %>%
  mutate(is_literal = ifelse(response == literal_response, "literal", "nonliteral")) %>% #Create factor
  mutate(is_literal = factor(is_literal, levels = c("nonliteral", "literal"))) #Set factor levels

#Show literal response rate for do vs. po
dopo_literal <- data %>%
  group_by(do_po) %>%
  summarize(literal_response_rate = mean(is_literal == "literal"))

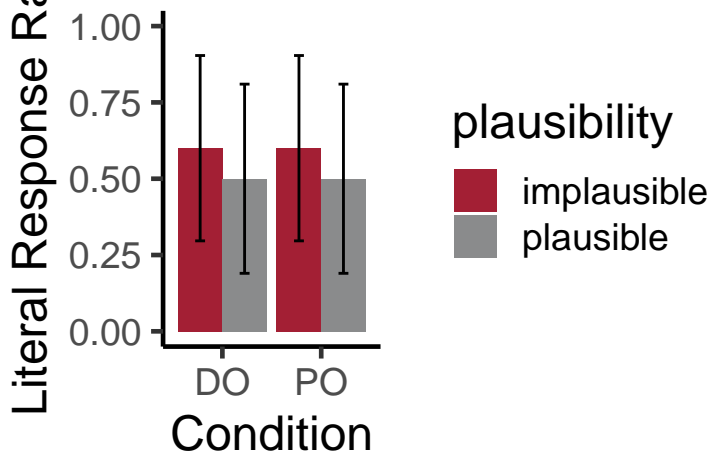
#Show literal response rate for implausible vs. plausible
plausibility_literal <- data %>%
  group_by(plausibility) %>%
  summarize(literal_response_rate = mean(is_literal == "literal"))

#Show literal response rate for do vs. po and implausible vs. plausible
both_rate <- data %>%
  group_by(do_po, plausibility) %>%
  summarize( #Get literal response rate for each condition, and calculate confidence intervals
    literal_response_rate = mean(is_literal == "literal"),
    n = n(),
    ci = 1.96 * sqrt(literal_response_rate * (1 - literal_response_rate) / n)
  )
```

Plot results:

```
#Plot literal response rate for both conditions
ggplot(both_rate, aes(x = do_po, y = literal_response_rate, fill = plausibility)) +
  geom_bar(stat = "identity", position = "dodge") +
  scale_fill_manual(values = c("implausible" = "#A31F34", "plausible" = "#8A8B8C")) +
  labs(x = "Condition", y = "Literal Response Rate", title = "Literal Response Rate by Condition") +
  scale_y_continuous(limits = c(0, 1)) +
  theme(plot.title = element_text(hjust = 0.5)) +
  geom_errorbar(aes(ymin = literal_response_rate - ci, ymax = literal_response_rate + ci), width = .2, position = "dodge")
```

Response Rate by Condition



```
# Lmer model
contrasts(data$do_po) <- c(-.5, .5)
contrasts(data$plausibility) <- c(-.5, .5)
#Contrast the is_literal variable
contrasts(data$is_literal) <- c(0,1)

#Since our result is categorical, we need to do a logistic regression
model <- glmer(is_literal ~ do_po * plausibility + (1|item), data = data, family = binomial(link = "logit"))

#Show results
summary(model)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
## Approximation) [glmerMod]
## Family: binomial ( logit )
## Formula: is_literal ~ do_po * plausibility + (1 | item)
## Data: data
##
##      AIC      BIC   logLik deviance df.resid
##    64.1    72.6   -27.1    54.1     35
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.1394 -0.9550  0.5857  0.7547  1.0280
##
## Random effects:
##  Groups Name            Variance Std.Dev.
##  item    (Intercept) 0.8064    0.898
## Number of obs: 40, groups:  item, 20
##
## Fixed effects:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)      0.2422   0.4084   0.593   0.553
## do_po1           0.1594   0.7973   0.200   0.841
## plausibility1    -0.4757   0.7476  -0.636   0.525
```

```

## do_po1:plausibility1 -0.3931    1.6203  -0.243    0.808
##
## Correlation of Fixed Effects:
##          (Intr) do_po1 plsbl1
## do_po1          0.060
## plausiblty1 -0.041 -0.102
## d_p1:plsbl1 -0.072 -0.146  0.060

```