

## 03 CSI online aphasia: Spoken - Descriptives

Kirsten Stark

22 March, 2024

### Load packages

```
library(dplyr)

##
## 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
```

```
library(tidyr)

rm(list = ls())
```

### Load and preprocess data

```
options( "encoding" = "UTF-8" )

# input
input <- "aphasia_final.csv"

# load data
df <- read.csv(here::here("data", "transient_data_files", input),
               sep = ",", na = "")
```

### Duration of the experiment

```
print("Total duration, not outlier corrected")

## [1] "Total duration, not outlier corrected"
```

```
# mean(df$timetotal, na.rm=TRUE) # 33.13 min
mean(df$time_correct)
```

```
## [1] 28.59683
```

```
sd(df$timetotal) # 21.04 min
```

```
## [1] 21.13194
```

```
range(df$timetotal) # 15 to 126 min
```

```
## [1] 12.68333 167.58333
```

```
print("Total duration, split by session")
```

```
## [1] "Total duration, split by session"
```

```
df %>% group_by(type,session) %>% summarise(mean = mean(timetotal),
                                              median=median(timetotal),
                                              sd = sd(timetotal),
                                              min = min(timetotal),
                                              max = max(timetotal))
```

```
## 'summarise()' has grouped output by 'type'. You can override using the
## '.groups' argument.
```

```
## # A tibble: 6 x 7
## # Groups:   type [2]
##   type    session mean median    sd   min   max
##   <chr>    <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 PWA         1  39.4  29.5 23.0  21.1 104.
## 2 PWA         2  32.0  27.0 22.8  19.4 126.
## 3 PWA         3  29.2  24.6 13.6  15.0  74.6
## 4 control     1  28.3  19.2 32.3  13.8 168.
## 5 control     2  18.2  17.5  3.62 12.7  26.1
## 6 control     3  19.2  16.4  8.40 13.8  52.0
```

## Description of participants

Gender:

```
(df %>% mutate(gender_char = case_when(gender == 1 ~ "female",
                                         gender == 2 ~ "male")) %>%
  group_by(type, gender_char) %>% count(gender_char) %>%
  mutate(n=n/160/3) -> gender)
```

```
## # A tibble: 4 x 3
## # Groups:   type, gender_char [4]
##   type   gender_char     n
##   <chr>   <chr>         <dbl>
## 1 PWA     female             3
## 2 PWA     male              17
## 3 control female             3
## 4 control male              17
```

```
# 1 = female, 2 = male, 3 = diverse
```

```
print("percentage female:")
```

```
## [1] "percentage female:"
```

```
sum(df$gender == 1)/nrow(df)
```

```
## [1] 0.15
```

Age:

```
print('age:')
```

```
## [1] "age:"
```

```
df %>% group_by(type) %>%
  summarise(mean=round(mean(age),2), sd=round(sd(age),2), min=min(age), max=max(age))
```

```
## # A tibble: 2 x 5
##   type   mean    sd   min   max
##   <chr> <dbl> <dbl> <int> <int>
## 1 PWA    53.4  5.43   39    62
## 2 control 53.2  5.72   38    62
```

Handedness:

```
# 1 = left handed, 2 = right handed, 3 = ambidexter/both
(handedness <- df %>% mutate(handedness_char = case_when(handedness == 1 ~ "left-handed",
                                                         handedness == 2 ~ "right-handed",
                                                         handedness==3 ~ "ambidexter/both"))) %>%
  group_by(type) %>% count(handedness_char))
```

```
## # A tibble: 5 x 3
## # Groups:   type [2]
##   type   handedness_char     n
##   <chr>   <chr>         <int>
## 1 PWA     left-handed     1600
## 2 PWA     right-handed    8000
## 3 control ambidexter/both    960
## 4 control left-handed    1120
## 5 control right-handed   7520
```

```
print("percentage right-handed:")

## [1] "percentage right-handed:"

print('PWA')

## [1] "PWA"

sum(df$handedness[df$type=="PWA"] == 2)/nrow(df[df$type=="PWA",])

## [1] 0.8333333

print('control')

## [1] "control"

sum(df$handedness[df$type=="control"] == 2)/nrow(df[df$type=="control",])

## [1] 0.7833333
```

Mother tongue (experiment was restricted to native German speakers): This seems to have worked

```
table(df$language) # 1 = yes (mother tongue is German), 2 = no

##
##      1
## 19200
```

## Attention checks

### 1) Item vs. non-item

```
## Item vs. non-item
# CH01_01 (Taube), CH01_02 (Apfel), CH02_01 (Luftballon) and CH02_02 (Biene) are items and 2 should be
# CH01_03 (Radio), CH01_04 (Sparschwein), CH02_03 (Laptop) and CH02_04 (Wattestäbchen) are non-items and
## Did participants cheat
# CH03 = 1 - yes, I worked through it till the end,
# CH03 = 2 - no, I stopped or cheated midway
# CH03 = -9 - no answer

attcheck <- data.frame(subject = unique(df$subject))

df <- df %>% mutate(itemvsnonitem1 =
  case_when(CH01_01==2 & CH01_02==2 & CH01_03==1 & CH01_04==1 ~2, # all correct
            CH01_01==2 | CH01_02==2 ~1, # one correct
            CH01_01!=2 & CH01_02!=2 ~0)) %>% # none correct
  dplyr::mutate(itemvsnonitem2 =
    case_when(CH02_01==2 & CH02_02==2 & CH02_03==1 & CH02_04==1 ~2,
              CH02_01==2 | CH02_02==2 ~1,
              CH02_01!=2 & CH02_02!=2 ~0))
df %>% group_by(type, session) %>% count(itemvsnonitem1) %>% mutate(n=n/160)
```

```
## # A tibble: 12 x 4
## # Groups:   type, session [6]
##   type      session itemvsnonitem1      n
##   <chr>      <int>      <dbl> <dbl>
## 1 PWA          1          0      1
## 2 PWA          1          1      6
## 3 PWA          1          2     13
## 4 PWA          2          1      1
## 5 PWA          2          2     19
## 6 PWA          3          1      2
## 7 PWA          3          2     18
## 8 control      1          1      4
## 9 control      1          2     16
## 10 control     2          2     20
## 11 control     3          1      1
## 12 control     3          2     19
```

```
df %>% group_by(type, session) %>% count(itemvsnonitem2) %>% mutate(n=n/160)
```

```
## # A tibble: 13 x 4
## # Groups:   type, session [6]
##   type      session itemvsnonitem2      n
##   <chr>      <int>      <dbl> <dbl>
## 1 PWA          1          1      2
## 2 PWA          1          2     18
## 3 PWA          2          0      1
## 4 PWA          2          1      1
## 5 PWA          2          2     18
## 6 PWA          3          0      1
## 7 PWA          3          1      2
## 8 PWA          3          2     17
## 9 control      1          1      1
## 10 control     1          2     19
## 11 control     2          1      1
## 12 control     2          2     19
## 13 control     3          2     20
```

```
# table(df$itemvsnonitem1)/160
# table(df$itemvsnonitem2)/160
```

All had at least one item selected correctly in the attention test

## 2) Cheating

```
df <- df %>% mutate(CH03 = case_when(CH03 == 1 ~
  " Ja, ich habe alles bis zum Ende bearbeitet.",
  CH03 == 2 ~
  "Nein, ich habe zwischendurch aufgehört oder geschummelt."))
df %>% group_by(type, session) %>% count(CH03) %>% mutate(n=n/160)
```

```
## # A tibble: 9 x 4
## # Groups:   type, session [6]
##   type      session CH03
```

n

```
##   <chr>      <int> <chr>                                     <dbl>
## 1 PWA        1 " Ja, ich habe alles bis zum Ende bearbeitet."      19
## 2 PWA        1 "Nein, ich habe zwischendurch aufgehört oder geschumme~  1
## 3 PWA        2 " Ja, ich habe alles bis zum Ende bearbeitet."      20
## 4 PWA        3 " Ja, ich habe alles bis zum Ende bearbeitet."      19
## 5 PWA        3 "Nein, ich habe zwischendurch aufgehört oder geschumme~  1
## 6 control    1 " Ja, ich habe alles bis zum Ende bearbeitet."      20
## 7 control    2 " Ja, ich habe alles bis zum Ende bearbeitet."      20
## 8 control    3 " Ja, ich habe alles bis zum Ende bearbeitet."      19
## 9 control    3 "Nein, ich habe zwischendurch aufgehört oder geschumme~  1
```

```
table(df$CH03)/160
```

```
##
##           Ja, ich habe alles bis zum Ende bearbeitet.
##                                     117
## Nein, ich habe zwischendurch aufgehört oder geschummelt.
##                                     3
```

## Comments

Comments don't indicate any problems that should lead to participant exclusion:

```
table(df$comments)/160
```

```
## numeric(0)
```

## Arrays

```
table(df$array)/160
```

```
##
##  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26
##  5  4  3  3  4  4  5  5  4  3  4  4  5  3  4  3  7  5  5  3  3  4  3  3  5  4
## 27 28 29 30
##  4  3  3  5
```

```
x <- df %>% group_by(subject, session) %>% count(array) %>% mutate(n=n/160)
array_rep = 0
for(i in 2:nrow(x)){
  if(x$subject[i-1] == x$subject[i] & x$array[i-1] == x$array[i]){
    array_rep = array_rep+1
  }
}
print(paste0(array_rep, " time an array was repeated within the same participant"))
```

```
## [1] "1 time an array was repeated within the same participant"
```

```
df %>% group_by(type, session) %>% count(array) %>% mutate(n=n/160) %>% arrange(array)
```

```
## # A tibble: 94 x 4
## # Groups:   type, session [6]
##   type      session array      n
##   <chr>      <int> <int> <dbl>
## 1 PWA          1      1      3
## 2 PWA          3      1      1
## 3 control      2      1      1
## 4 PWA          2      2      1
## 5 PWA          3      2      2
## 6 control      2      2      1
## 7 control      1      3      1
## 8 control      2      3      1
## 9 control      3      3      1
## 10 PWA          1      4      1
## # i 84 more rows
```

```
df %>% count(array) %>% mutate(n=n/160)
```

```
##   array n
## 1     1 5
## 2     2 4
## 3     3 3
## 4     4 3
## 5     5 4
## 6     6 4
## 7     7 5
## 8     8 5
## 9     9 4
## 10    10 3
## 11    11 4
## 12    12 4
## 13    13 5
## 14    14 3
## 15    15 4
## 16    16 3
## 17    17 7
## 18    18 5
## 19    19 5
## 20    20 3
## 21    21 3
## 22    22 4
## 23    23 3
## 24    24 3
## 25    25 5
## 26    26 4
## 27    27 4
## 28    28 3
## 29    29 3
## 30    30 5
```

## Fully anonymize data and reduce data frame

```
# df_a <- df %>% dplyr::select(!"gender" & !starts_with("language") &
#                               !starts_with("handedness") &
#                               !starts_with("fingers") &
#                               !starts_with("KB") &
#                               !starts_with("CHO") &
#                               !"comments" & !"type" &
#                               !"gender_char" & !"itemusnonitem1" &
#                               !"itemusnonitem2" &
#                               !starts_with("MC") &
#                               !"name" & !"time_wo_outlier" )

# write.csv(df_a, here::here("data", "data_long_anonymous.csv"))
```