PP_Narrative_Analysis_H7

Kristen Johnson

2025-04-04

```
# install necessary packages

#install.packages("lmerTest")

# Load required packages
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.0.4
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(car)       # For regression diagnostics
```

```
## Loading required package: carData
##
## Attaching package: 'car'
##
## The following object is masked from 'package:dplyr':
##
##     recode
##
## The following object is masked from 'package:purrr':
##
##     some
```

```
library(lmtest)    # For regression diagnostics
```

```
## Loading required package: zoo
##
## Attaching package: 'zoo'
##
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
library(effectsize) # For effect sizes
library(ggplot2)   # For visualization
library(here)
```

```
## here() starts at /Users/kristenjohnson/KristenWorkingDirectory/Play_Narrative
```

```
library(dplyr)
library(lme4)
```

```
## Loading required package: Matrix
##
## Attaching package: 'Matrix'
##
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

```
library(lmerTest)
```

```
##
## Attaching package: 'lmerTest'
##
## The following object is masked from 'package:lme4':
##
##     lmer
##
## The following object is masked from 'package:stats':
##
##     step
```

```
library(extrafont)
```

```
## Registering fonts with R
```

```
# find  project root directory automatically
merged_H7 <- read.csv(here("PN_Datasets", "Pretend_Play", "CSVs_of_Combined_Data_PP", "merged_H7.csv"))

H7_gesture_play <- merged_H7 %>%
  group_by(participant_id) %>%
  summarise(
    total_pretend_play = sum(c_pret, na.rm = TRUE),
    mlu = mean(c_wpu, na.rm = TRUE),
    any_pretend_play = if_else(any(c_pret == 1, na.rm = TRUE), 1, 0),
    total_rep_gestures = sum(gesture_rep == 1, na.rm = TRUE),
    total_all_gestures = sum(gesture_all == 1, na.rm = TRUE),
    group_status = first(groupstatus)
  )

write_csv(H7_gesture_play,
          here("PN_DataAnalysis", "PP_Narrative_Analysis", "H7_gesture_play.csv"))
```

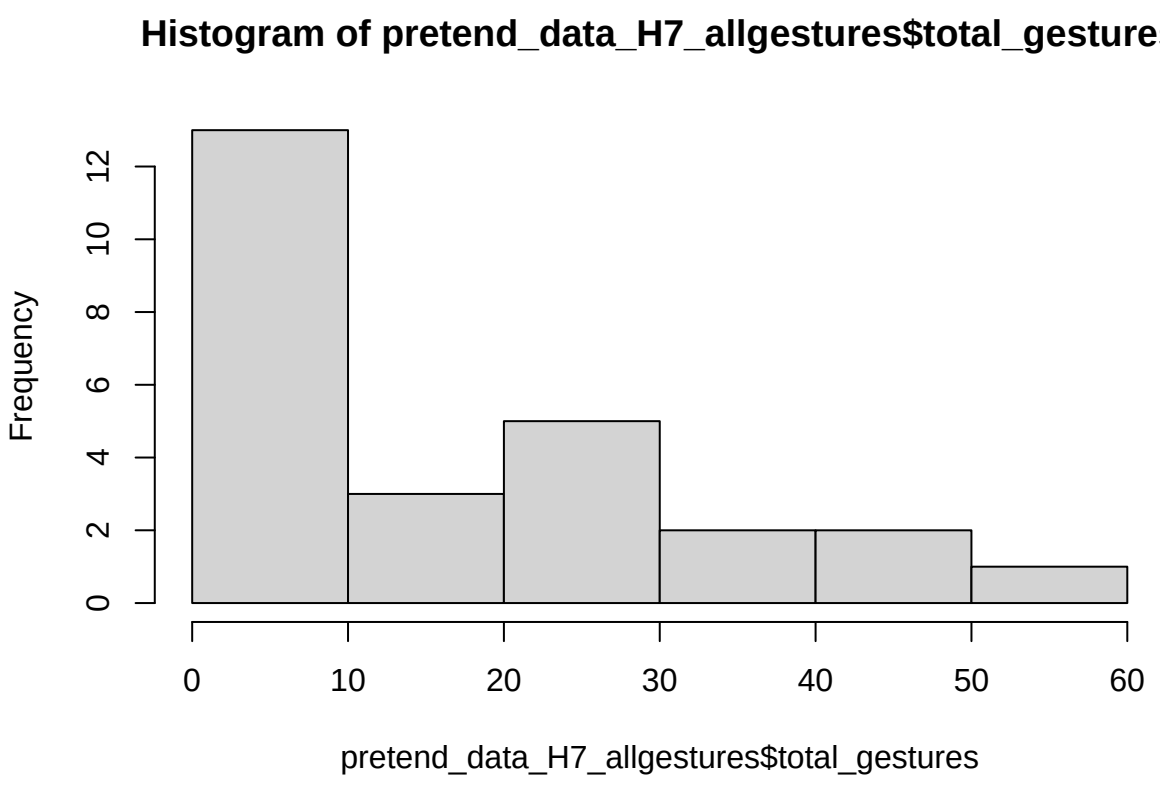
```
# read in data frame that is just gesture during instance of pretend play
pretend_data_H7_allgestures <- read.csv(here("PN_Datasets", "Pretend_Play", "CSVs_of_Combined_Data_PP",

# Aggregate by child
child_summary_H7 <- pretend_data_H7_allgestures %>%
  group_by(participant_id, groupstatus) %>%
  summarize(
    total_gestures = sum(gesture_all),
```

```
    rep_gestures = sum(gesture_rep),
    icon_gestures = sum(gesture_icon),
    prop_rep = sum(gesture_rep) / sum(gesture_all),
    prop_icon = sum(gesture_icon) / sum(gesture_all),
  )
```

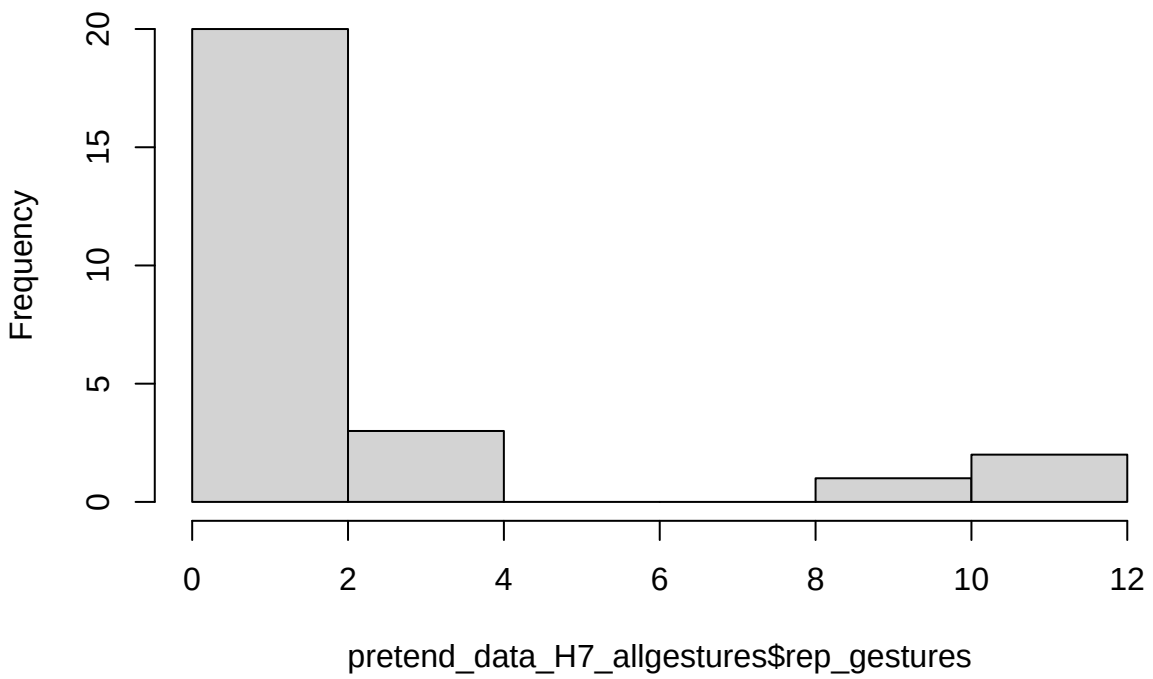
```
## `summarise()` has grouped output by 'participant_id'. You can override using
## the `.groups` argument.
```

```
# add mlu from all utterances

pretend_data_H7_allgestures <- child_summary_H7 %>%
  left_join(select(H7_gesture_play, participant_id, mlu), by = "participant_id") %>%
  left_join(select(H7_gesture_play, participant_id, total_pretend_play), by = "participant_id")

write_csv(pretend_data_H7_allgestures, here("PN_DataAnalysis", "PP_Narrative_Analysis", "pretend_data_H
```

```
# Create binary variable for representational gesture presence
pretend_data_H7_allgestures <- pretend_data_H7_allgestures %>%
  mutate(rep_gesture_present = ifelse(rep_gestures > 0, 1, 0))

# Check distributions
hist(pretend_data_H7_allgestures$total_gestures)
```




```
hist(pretend_data_H7_allgestures$rep_gestures)
```



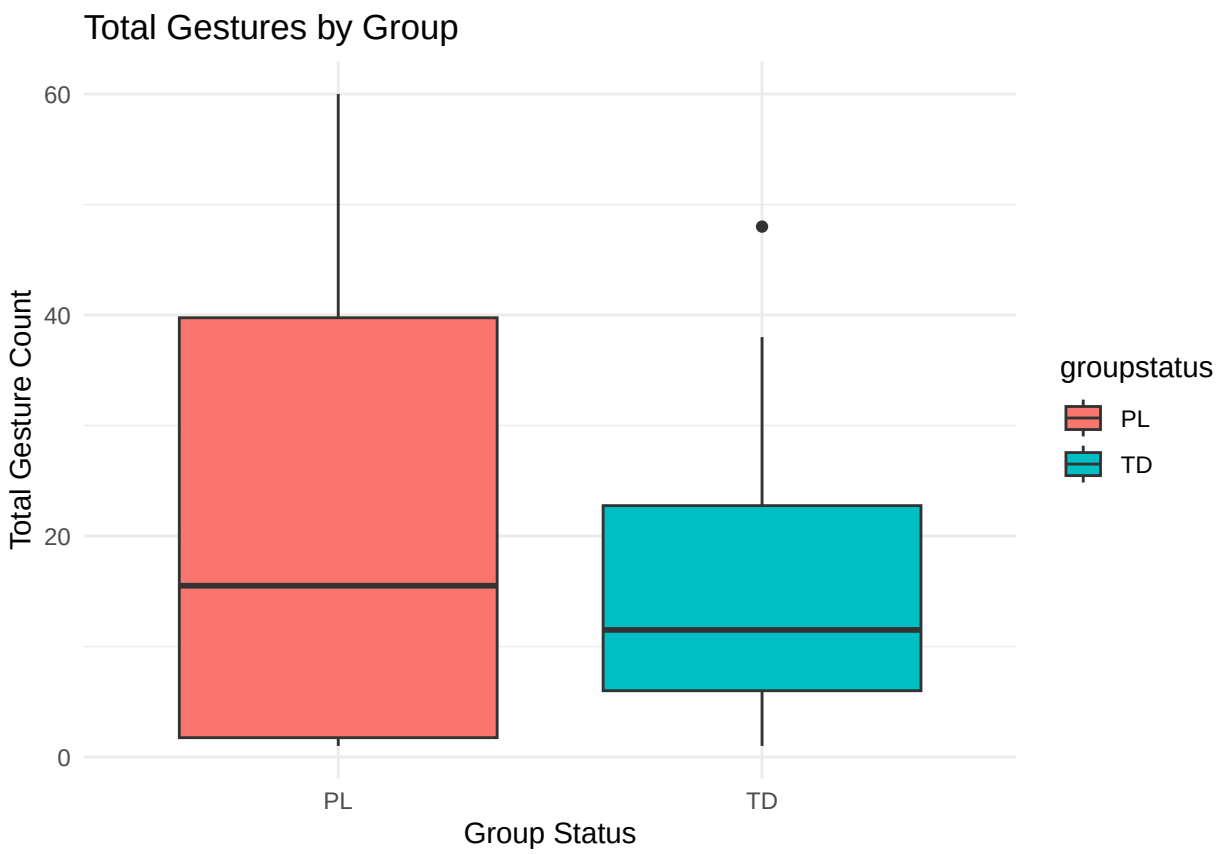

```
table(pretend_data_H7_allgestures$rep_gesture_present, pretend_data_H7_allgestures$groupstatus)
```

```
##
##     PL TD
##   0  3 10
##   1  3 10
```

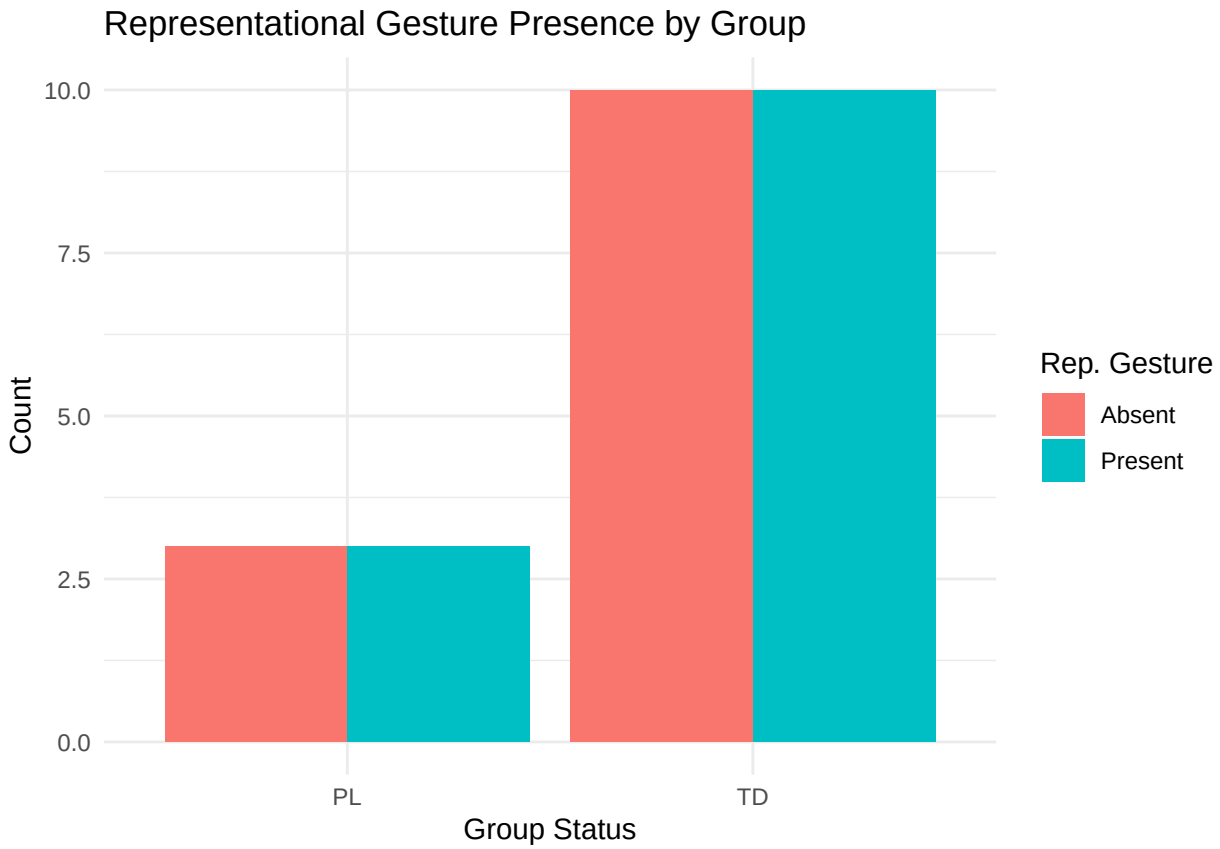
```
# Check correlation between total gestures and representational gestures
cor.test(pretend_data_H7_allgestures$total_gestures, pretend_data_H7_allgestures$rep_gestures)
```

```
##
##  Pearson's product-moment correlation
##
## data:  pretend_data_H7_allgestures$total_gestures and pretend_data_H7_allgestures$rep_gestures
## t = 5.0888, df = 24, p-value = 3.32e-05
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.4619815 0.8660832
## sample estimates:
##       cor
## 0.7204155
```

```
# 2. Data visualization
# Boxplot of total gestures by group
ggplot(pretend_data_H7_allgestures, aes(x = groupstatus, y = total_gestures, fill = groupstatus)) +
  geom_boxplot() +
  theme_minimal() +
  labs(title = "Total Gestures by Group",
       x = "Group Status",
       y = "Total Gesture Count")
```




```
# Bar chart of representational gesture presence by group
ggplot(pretend_data_H7_allgestures, aes(x = groupstatus, fill = factor(rep_gesture_present))) +
  geom_bar(position = "dodge") +
  scale_fill_discrete(name = "Rep. Gesture", labels = c("Absent", "Present")) +
  theme_minimal() +
  labs(title = "Representational Gesture Presence by Group",
       x = "Group Status",
       y = "Count")
```



#

3. Merge with narrative outcome data

```
narrative_data <- read.csv(here("PN_Datasets", "Narrative", "CSVs_of_Combined_Data_N", "Narrative_Resul

# Rename participant column
narrative_data <- narrative_data %>%
  rename(participant_id = ChildID)

write.csv(narrative_data, here("PN_DataAnalysis", "PP_Narrative_Analysis", "narrative_data.csv"))

# MERGE IT
completely_merged_data_H7 <- left_join(pretend_data_H7_allgestures, narrative_data, by = "participant_i
  mutate(groupstatus = if_else(groupstatus == "BI", "PL", groupstatus))

write_csv(completely_merged_data_H7, here("PN_DataAnalysis", "PP_Narrative_Analysis", "completely_merge
```

4. Run regression models

```
# Model 1: Total gestures as predictor
model1 <- lm(AvgHighScore ~ groupstatus + total_gestures + total_pretend_play + mlu + groupstatus:total_
             data = completely_merged_data_H7)

# Model 2: Representational gesture presence as predictor
model2 <- lm(AvgHighScore ~ groupstatus + rep_gesture_present + total_pretend_play + mlu + groupstatus:r
             data = completely_merged_data_H7)

summary(model1)
```

```
##
## Call:
## lm(formula = AvgHighScore ~ groupstatus + total_gestures + total_pretend_play +
##     mlu + groupstatus:total_gestures, data = completely_merged_data_H7)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.1347 -1.1394  0.0481  0.7714  3.4300
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  -0.081939   1.864006  -0.044   0.9654
## groupstatusTD                 3.306381   1.331909   2.482   0.0226 *
## total_gestures                0.078550   0.075468   1.041   0.3110
## total_pretend_play           -0.000032   0.006872  -0.005   0.9963
## mlu                           2.343287   0.960700   2.439   0.0247 *
## groupstatusTD:total_gestures -0.128331   0.060697  -2.114   0.0479 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.764 on 19 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.4036, Adjusted R-squared:  0.2467
## F-statistic: 2.572 on 5 and 19 DF,  p-value: 0.06127
```

```
summary(model2)
```

```
##
## Call:
## lm(formula = AvgHighScore ~ groupstatus + rep_gesture_present +
##     total_pretend_play + mlu + groupstatus:rep_gesture_present,
##     data = completely_merged_data_H7)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.54126 -0.69846  0.01946  0.71849  3.12233
##
## Coefficients:
##                                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      -2.338263   1.571149  -1.488 0.153095
## groupstatusTD                     4.529058   1.102875   4.107 0.000601 ***
## rep_gesture_present               6.992690   2.041985   3.424 0.002843 **
## total_pretend_play               -0.004265   0.003639  -1.172 0.255653
## mlu                               3.478107   0.798807   4.354 0.000342 ***
## groupstatusTD:rep_gesture_present -8.507163   2.112123  -4.028 0.000719 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.464 on 19 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.5894, Adjusted R-squared:  0.4813
## F-statistic: 5.454 on 5 and 19 DF,  p-value: 0.002812
```

```
# Calculate effect sizes
eta_squared(model1)
```

```
## # Effect Size for ANOVA (Type I)
##
## Parameter                  | Eta2 (partial) |       95% CI
## ------------------------------------------------------------
## groupstatus                |           0.05 | [0.00, 1.00]
## total_gestures             |       2.20e-03 | [0.00, 1.00]
## total_pretend_play         |           0.20 | [0.01, 1.00]
## mlu                        |           0.12 | [0.00, 1.00]
## groupstatus:total_gestures |           0.19 | [0.00, 1.00]
##
## - One-sided CIs: upper bound fixed at [1.00].
```

```
eta_squared(model2)
```

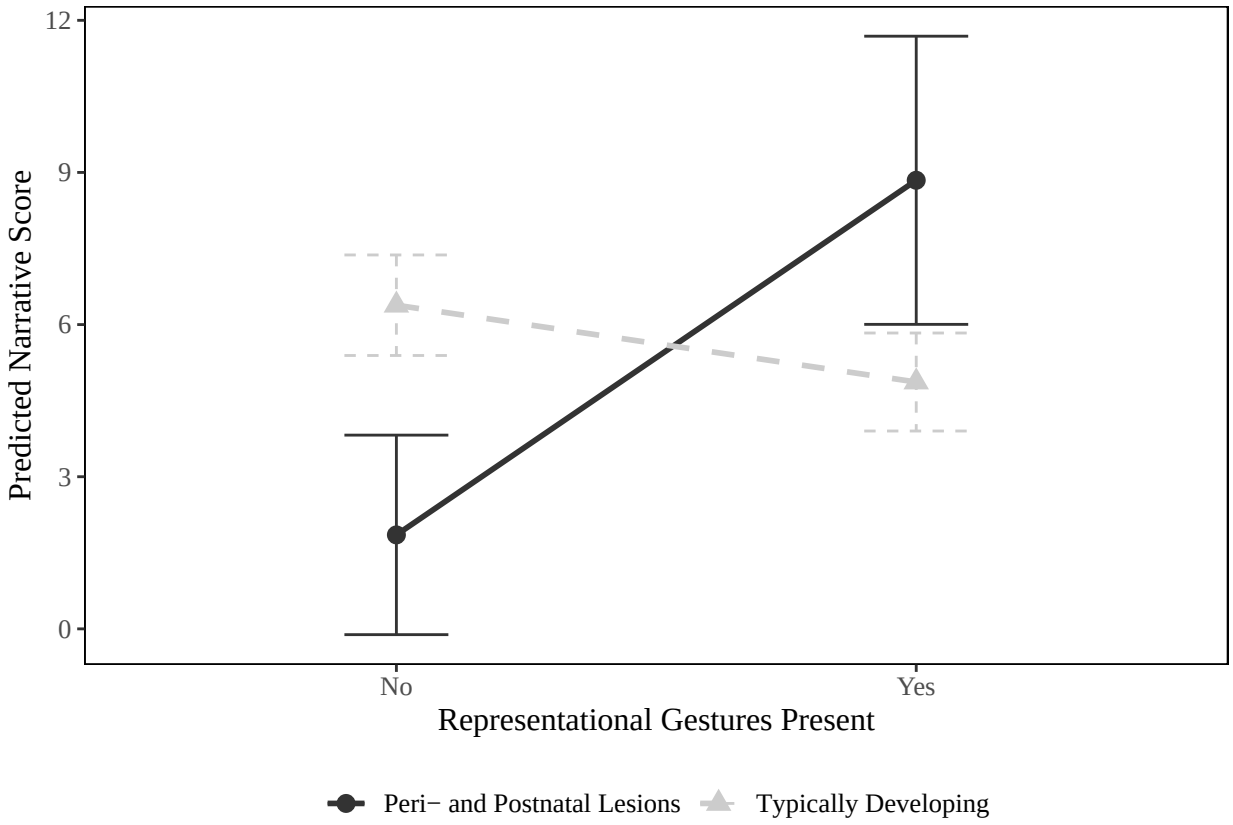
```
## # Effect Size for ANOVA (Type I)
##
## Parameter                        | Eta2 (partial) |       95% CI
## ------------------------------------------------------------------
## groupstatus                      |           0.07 | [0.00, 1.00]
## rep_gesture_present              |           0.02 | [0.00, 1.00]
## total_pretend_play               |           0.12 | [0.00, 1.00]
## mlu                              |           0.26 | [0.03, 1.00]
## groupstatus:rep_gesture_present |           0.46 | [0.18, 1.00]
##
## - One-sided CIs: upper bound fixed at [1.00].
```

5b. USE WHEN HAVE ALL DATA FROM 3 TIMEPOINTS..... mixed effects models

```
#library(lme4)
#library(lmerTest)

#mixed_model1 <- lmer(AvgHighScore ~ GroupStatus + total_gestures + total_pretend_play + mlu +
                     # GroupStatus:total_gestures + (1|participant_id),
                     # data = completely_merged_data)

#mixed_model2 <- lmer(AvgHighScore ~ GroupStatus + rep_gesture_present + total_pretend_play + mlu +
                      #GroupStatus:rep_gesture_present + (1|participant_id),
                      #data = completely_merged_data)

#summary(mixed_model1)
#summary(mixed_model2)
```

```
# Interaction Plot: Group × Representational Gesture Presence
#| label: fig-interaction
#| fig-cap: "Predicted narrative scores by group and representational gesture use. Error bars represent
#| fig-width: 6
#| fig-height: 4
#|
options(warn = 2)
```

```r
# Create a data frame for prediction (numeric variables for model compatibility)
newdata <- expand.grid(
  groupstatus = c("PL", "TD"),
  rep_gesture_present = c(0, 1),
  total_pretend_play = mean(completely_merged_data_H7$total_pretend_play, na.rm = TRUE),
  mlu = mean(completely_merged_data_H7$mlu, na.rm = TRUE)
)

# Get predictions and standard errors from the model
newdata$predicted <- predict(model2, newdata = newdata)
pred_se <- predict(model2, newdata = newdata, se.fit = TRUE)$se.fit
newdata$lower <- newdata$predicted - 1.96 * pred_se
newdata$upper <- newdata$predicted + 1.96 * pred_se

# Relabel variables as factors for plotting clarity
newdata$groupstatus <- factor(newdata$groupstatus,
                              levels = c("PL", "TD"),
                              labels = c("Peri- and Postnatal Lesions", "Typically Developing"))
newdata$rep_gesture_present <- factor(newdata$rep_gesture_present,
                                      levels = c(0, 1),
                                      labels = c("No", "Yes"))

# Create the interaction plot
ggplot(newdata, aes(x = rep_gesture_present, y = predicted,
                    group = groupstatus, color = groupstatus,
                    linetype = groupstatus, shape = groupstatus)) +
  geom_line(linewidth = 1) +
  geom_point(size = 3) +
  geom_errorbar(aes(ymin = lower, ymax = upper), width = 0.2) +
  labs(x = "Representational Gestures Present",
       y = "Predicted Narrative Score") +
  scale_color_grey(start = 0.2, end = 0.8) +
  scale_linetype_manual(values = c("solid", "dashed")) +
  scale_shape_manual(values = c(16, 17)) +
  theme_bw() +
  theme(
    text = element_text(family = "Times New Roman", size = 12),
    axis.title = element_text(size = 12),
    axis.text = element_text(size = 10),
    legend.position = "bottom",
    legend.title = element_blank(),
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    panel.border = element_rect(color = "black")
  )
```



Scatter Plot: Raw Data with Regression Lines

#| label: fig-scatter #| fig-cap: “Relationship between representational gesture presence and narrative scores by group. Shaded areas represent 95% confidence intervals around regression lines.” #| fig-width: 7 #| fig-height: 4

ggplot(completely_merged_data_H10, aes(x = as.factor(rep_gesture_present), y = AvgHighScore, fill = GroupStatus, shape = GroupStatus)) + geom_jitter(width = 0.1, height = 0, size = 3, alpha = 0.7) + geom_smooth(aes(group = GroupStatus, linetype = GroupStatus, color = GroupStatus), method = “lm”, se = TRUE, formula = y ~ x) + scale_x_discrete(labels = c(“0” = “No”, “1” = “Yes”)) + scale_shape_manual(values = c(21, 24)) + scale_fill_grey(start = 0.5, end = 0.8) + scale_color_grey(start = 0.2, end = 0.6) + scale_linetype_manual(values = c(“solid”, “dashed”)) + labs(x = “Representational Gestures Present”, y = “Narrative Score”) + facet_wrap(~ GroupStatus, scales = “free_x”) + theme_bw() + theme(text = element_text(family = “Times New Roman”, size = 12), axis.title = element_text(size = 12), axis.text = element_text(size = 10), legend.position = “none”, panel.grid.major = element_blank(), panel.grid.minor = element_blank(), panel.border = element_rect(color = “black”), strip.background = element_rect(fill = “white”), strip.text = element_text(size = 12))

Total Gestures Scatter Plot with facet_grid

#| label: fig-total-gestures-grid #| fig-cap: “Relationship between total gesture count and narrative scores, faceted by group status and the presence of representational gesture. The top row shows children who used representational gestures during pretend play, while the bottom row shows those who did not.” #| fig-width: 8 #| fig-height: 6

First, create a factor for rep_gesture_present for better labeling

completely_merged_data$rep_gesture_factor <- factor(completely_merged_data$rep_gesture_present, levels = c(0, 1), labels = c("No Representational", "Representational"))

ggplot(completely_merged_data, aes(x = total_gestures, y = AvgHighScore, color = GroupStatus, shape = GroupStatus)) + geom_point(size = 3, alpha = 0.8) + geom_smooth(aes(linetype = GroupStatus), method = "lm", se = TRUE) + labs(x = "Total Number of Gestures", y = "Narrative Score") + scale_color_grey(start = 0.2, end = 0.8) + scale_shape_manual(values = c(16, 17)) + scale_linetype_manual(values = c("solid", "dashed")) + # Add facet_grid with rep_gesture_factor in rows and GroupStatus in columns facet_grid(rep_gesture_factor ~ GroupStatus, scales = "free_x") + theme_bw() + theme(text = element_text(family = "Times New Roman", size = 12), axis.title = element_text(size = 12), axis.text = element_text(size = 10), legend.position = "bottom", legend.title = element_blank(), panel.grid.major = element_blank(), panel.grid.minor = element_blank(), panel.border = element_rect(color = "black"), strip.background = element_rect(fill = "white"), strip.text = element_text(size = 11)) "'