GLMM Analysis for Circle Time

Pourya Shahverdi

2025-06-30

Contents

1 About the Study

1.1 Introduction to Circle-Time

Circle-time is a group activity based on Applied Behavior Analysis (ABA) for children with Autism Spectrum Disorder (ASD) to prepare them for attending in traditional classroom activities alongside neurotypically developing children. In circle-time, children sit together semicircular, and an instructor gives them group instruction activities such as dance, yoga, labeling animals, finding objects, etc. The goal of circle-time is to improve children's learning behaviors, which are:

- Affect
- Communication
- Engagement
- Performance

In this study, we evaluate the efficacy of a social robot in delivering group instruction activities to children with ASD. Throughout the six months of experiment, six children participants received 10 sessions of group instructions from a human instructor and 10 sessions from a Pepper humanoid social robot as a within-subject study design. To compare children's learning behaviors between the human and the robot instructor conditions, their activities were video recorded and coded for sessions 1, 4, 7, and 10.

1.2 Study Design

For this longitudinal within-subject study with 6 participants we defined the following variables:

1.2.1 Independent Variables

- Instructor Conditions:
  - Human ~ 1
  - Robot ~ 2

- Time:
  - Session 1 ~ 1
  - Session 4 ~ 2
  - Session 7 ~ 3
  - Session 10 ~ 4

1.2.2 Dependent Variables

- Affect
- Communication
- Engagement
- Performance

1.3 Data Collection

The evaluation of the learning behavior is based on the following continuous metrics:

1.3.1 Affect

Children's happiness level was defined as:

- Positive
- Negative
- Neutral

A video was divided into 10-second intervals. A human coder, focusing on one child in the group, labeled that interval as Positive if the child was showing positive affective behaviors (e.g., smiling, clapping, laughing). An interval was labeled as Negative if the child was showing negative affective behaviors (e.g., crying, whining, frowning). Neutral was used when neither applied. The percentage of each type was used for analysis.

1.3.2 Communication

Communication was coded into 4 categories:

- With Instructor
- Instructor-Prompted
- With Behavior Therapist (BT) or peers
- Indeterminate

1.3.3 Engagement

Engagement was coded into 3 categories:

- On Target (Instructor or screen)
- With BT or peers
- Off Target

1.3.4 Performance

Children's performance was coded as:

- Positive
- Negative

1.3.5 Inter-observer Agreement (IoA)

Coders were required to exceed 80% Cohen's Kappa IoA to code independently. All session ones and tens were double-coded, along with 30% of session fours and sevens. Disagreements were resolved through joint review to ensure 100% final agreement.

2 Data Analysis

2.1 Additive and Interaction Models and Plots

In this section, we first fit an additive model to evaluate the overall effects of instructor condition (Human vs. Robot), session time, and children's VB-MAPP scores on each learning behavior. By including VB-MAPP as a covariate, we control for differences in individual skill levels, ensuring a fair comparison of the children's responses across instructional conditions over time.

Next, we extend the analysis using an interaction model that tests whether the effect of instructor condition varies depending on the session time or the VB-MAPP score. This helps uncover whether certain children or timepoints show stronger effects from robot-led instruction.

To visualize the results, we generate two types of plots for each learning behavior. The first plot overlays violin plots of bootstrapped behavioral estimates across session timepoints. These violins illustrate the distribution and spread of responses for each instructor condition. We also superimpose the mean value of each group with connecting lines, providing a clear view of trends across sessions. The second plot shows the predicted behavioral outcome as a function of VB-MAPP score for both Human and Robot conditions. Shaded confidence bands represent the uncertainty around these estimates, allowing us to visualize how instructor efficacy changes as a function of the child's developmental profile.

Together, these analyses and visualizations offer a comprehensive understanding of how social robot instruction compares to human-led instruction across time and learner profiles.

```r
library(glmmTMB)
library(performance)
library(DHARMa)
```

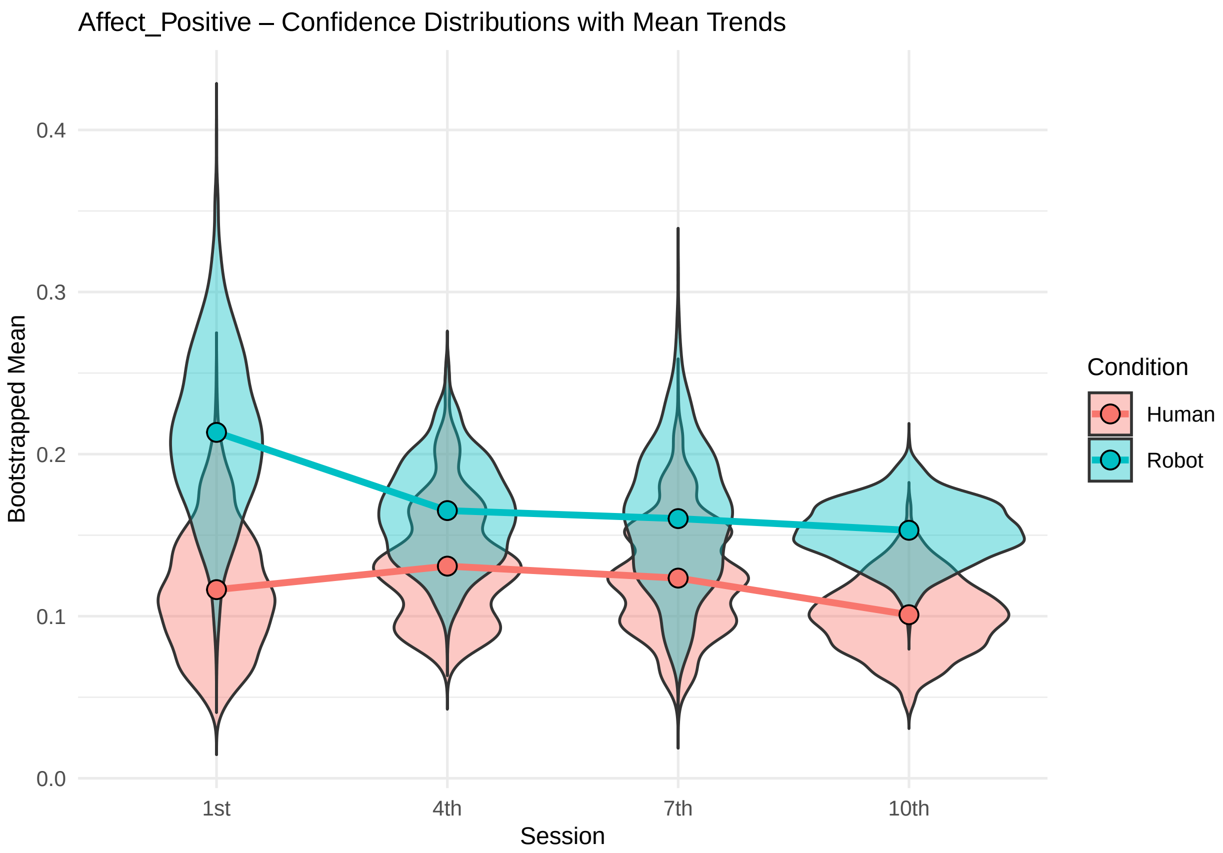
```
## This is DHARMa 0.4.7. For overview type '?DHARMa'. For recent changes, type news(package = 'DHARMa')
```

```r
library(ggplot2)
library(broom.mixed)
library(dplyr)
```

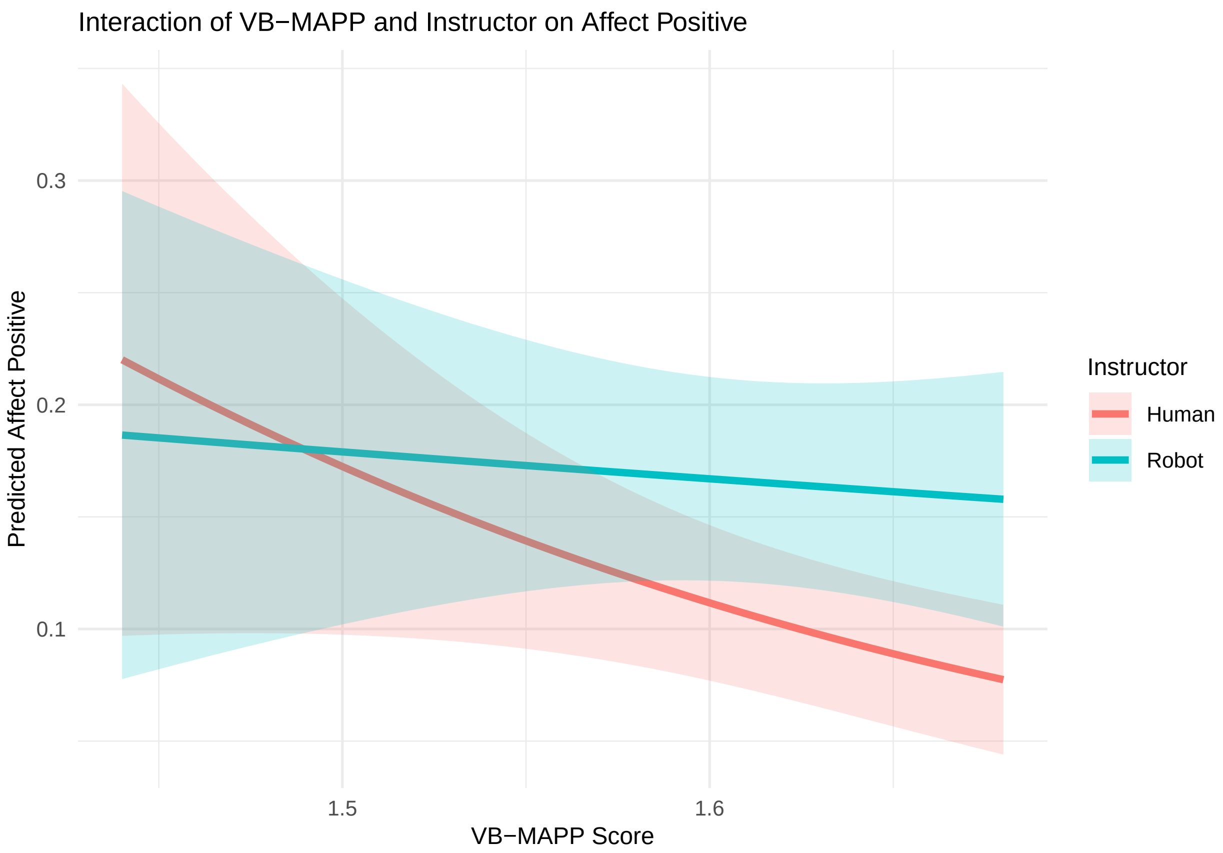
```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(boot)
library(tidyr)

# Load and preprocess data
df <- read.csv("~/GitHub/Circle-Time-Data-Analyses/CircleTimeData-VBMAPP.csv")
df$Condition <- factor(df$Condition, levels = c(1, 2), labels = c("Human", "Robot"))
df$time_c <- scale(df$time, center = TRUE, scale = FALSE)

vars <- c(
  "Affect_Positive", "Affect_Negative",
  "Communication_with_Instructor", "Communication_with_Instructor_Prompted",
  "Communication_with_Therapist", "Communication_with_Indeterminent",
  "Engagement_OnTarget", "Engagement_Therapist", "Engagement_OffTarget",
  "Performance_Positive"
)

# Apply beta transformation
epsilon <- 0.0001
for (v in vars) {
  df[[v]] <- (df[[v]] * (nrow(df) - 1) + epsilon) / nrow(df)
}

# Significance stars function
sig_stars <- function(p) {
  if (is.na(p)) return("")
  if (p < 0.001) return("***")
  else if (p < 0.01) return("**")
  else if (p < 0.05) return("*")
  else if (p < 0.1) return(".")
  else return("")
}

## Additive and Interaction Models and Plots ##
## This section evaluates the behavioral metrics of interest (e.g., Affect, Engagement) through two mode
## 1. An additive model controlling for VB-MAPP scores.
## 2. An interaction model examining moderation effects of Condition with time and VB-MAPP.
##
## We then visualize the bootstrapped confidence distributions using violin plots, mean trends, and the

for (var in vars) {
  cat("\n===============================\n")
  cat("Behavior:", var, "\n")

  # --- Additive Model ---
  form_add <- as.formula(paste0(var, " ~ Condition + time_c + VBMAPP + (1 | Subject)"))
  model_add <- glmmTMB(form_add, data = df, family = beta_family())

  table_add <- broom.mixed::tidy(model_add, effects = "fixed")
  table_add$stars <- sapply(table_add$p.value, sig_stars)

  cat("\nAdditive Model Summary:\n")
  print(table_add[, c("term", "estimate", "std.error", "statistic", "p.value", "stars")])

  # --- Interaction Model ---
  form_int <- as.formula(paste0(var, " ~ Condition * time_c + Condition * VBMAPP + (1 | Subject)"))
  model_int <- glmmTMB(form_int, data = df, family = beta_family())

  table_int <- broom.mixed::tidy(model_int, effects = "fixed")
  table_int$stars <- sapply(table_int$p.value, sig_stars)

  cat("\nInteraction Model Summary:\n")
  print(table_int[, c("term", "estimate", "std.error", "statistic", "p.value", "stars")])

  # --- Bootstrap Violin Plots with Mean Line ---
  mean_df <- df %>%
    group_by(Condition, time) %>%
    summarise(mean_value = mean(.data[[var]]), .groups = "drop")

  bootstrap_samples <- df %>%
    group_by(Condition, time) %>%
    group_modify(~ {
      boot_result <- boot(.x[[var]], statistic = function(d, i) mean(d[i]), R = 1000)
      tibble(bootstrap = boot_result$t[, 1])
    }) %>%
    ungroup()

  p_violin_line <- ggplot() +
    geom_violin(
      data = bootstrap_samples,
      aes(x = factor(time), y = bootstrap, fill = Condition),
      position = position_identity(),
      alpha = 0.4,
      trim = FALSE,
      width = 1
    ) +
    geom_line(
      data = mean_df,
      aes(x = as.numeric(factor(time)), y = mean_value, color = Condition, group = Condition),
      linewidth = 1.2
    ) +
    geom_point(
      data = mean_df,
      aes(x = as.numeric(factor(time)), y = mean_value, fill = Condition),
      color = "black",
      shape = 21,
      size = 3
    ) +
    scale_x_discrete(
      name = "Session",
      labels = c("1st", "4th", "7th", "10th")
    ) +
    labs(
      title = paste(var, "- Confidence Distributions with Mean Trends"),
      y = "Bootstrapped Mean",
      fill = "Condition",
      color = "Condition"
    ) +
      theme_minimal(base_size = 10) +
  theme(
    plot.title = element_text(size = 10),
    axis.title = element_text(size = 9),
    axis.text = element_text(size = 8),
    legend.title = element_text(size = 9),
    legend.text = element_text(size = 8)
  )

  print(p_violin_line)

  # --- VB-MAPP Interaction Plot ---
  new_data <- expand.grid(
    Condition = factor(c("Human", "Robot"), levels = c("Human", "Robot")),
    time_c = 0,
    VBMAPP = seq(min(df$VBMAPP), max(df$VBMAPP), length.out = 100),
    Subject = NA
  )
  pred <- predict(model_int, newdata = new_data, type = "response", se.fit = TRUE)
  new_data$predicted <- pred$fit
  new_data$conf.low <- pred$fit - 1.96 * pred$se.fit
  new_data$conf.high <- pred$fit + 1.96 * pred$se.fit
  new_data$group <- new_data$Condition

  p2 <- ggplot(new_data, aes(x = VBMAPP, y = predicted, color = group)) +
    geom_line(size = 1.3) +
    geom_ribbon(aes(ymin = conf.low, ymax = conf.high, fill = group), alpha = 0.2, color = NA) +
    labs(
      x = "VB-MAPP Score",
      y = paste("Predicted", gsub("_", " ", var)),
      color = "Instructor",
      fill = "Instructor",
      title = paste("Interaction of VB-MAPP and Instructor on", gsub("_", " ", var))
    ) +
    theme_minimal(base_size = 10) +
  theme(
    plot.title = element_text(size = 10),
    axis.title = element_text(size = 9),
    axis.text = element_text(size = 8),
    legend.title = element_text(size = 9),
    legend.text = element_text(size = 8)
  )

  print(p2)
}
```

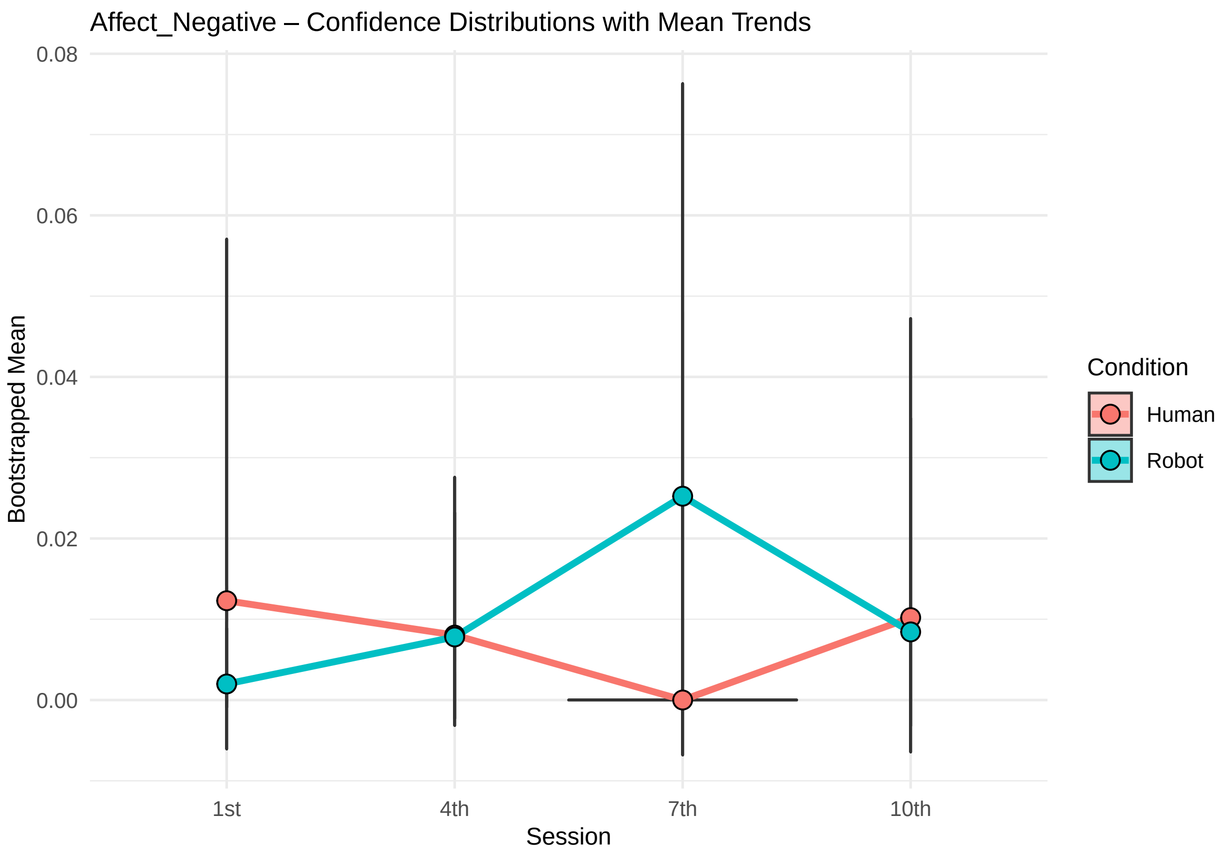
```
##
## ==============================
## Behavior: Affect_Positive
##
## Additive Model Summary:
## # A tibble: 4 x 6
##   term           estimate std.error statistic p.value stars
##   <chr>             <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)      2.23      2.89       0.770 0.441   ""
## 2 ConditionRobot   0.435     0.151      2.89  0.00390 "**"
## 3 time_c          -0.0776    0.0658    -1.18  0.238   ""
## 4 VBMAPP          -2.66      1.79      -1.49  0.138   ""
##
## Interaction Model Summary:
## # A tibble: 6 x 6
##   term                  estimate std.error statistic p.value stars
##   <chr>                    <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)             6.01      3.19       1.88  0.0597  "."
## 2 ConditionRobot         -6.28      2.35      -2.67  0.00755 "**"
## 3 time_c                 -0.0472    0.0901    -0.523 0.601   ""
## 4 VBMAPP                 -5.05      1.99      -2.54  0.0112  "*"
## 5 ConditionRobot:time_c  -0.0696    0.121     -0.574 0.566   ""
## 6 ConditionRobot:VBMAPP   4.22      1.48       2.86  0.00426 "**"
```

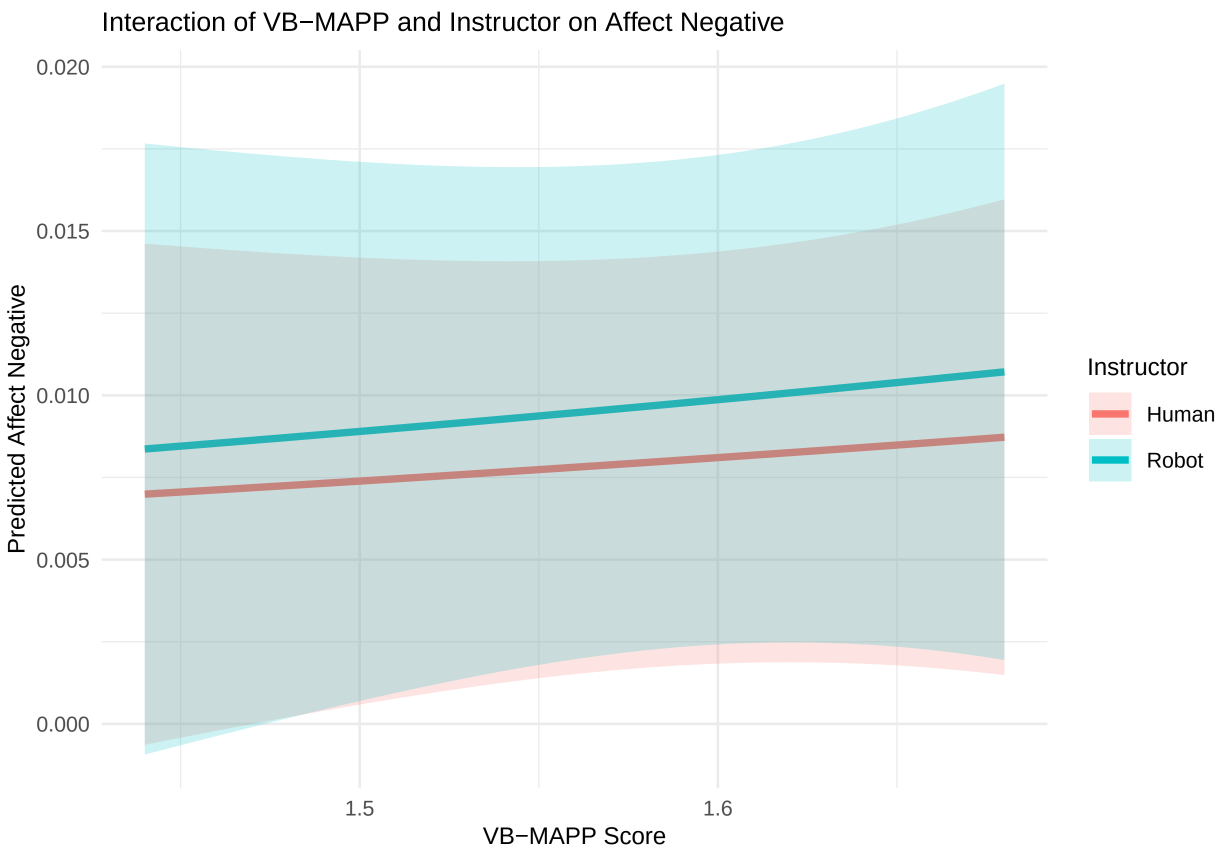


```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

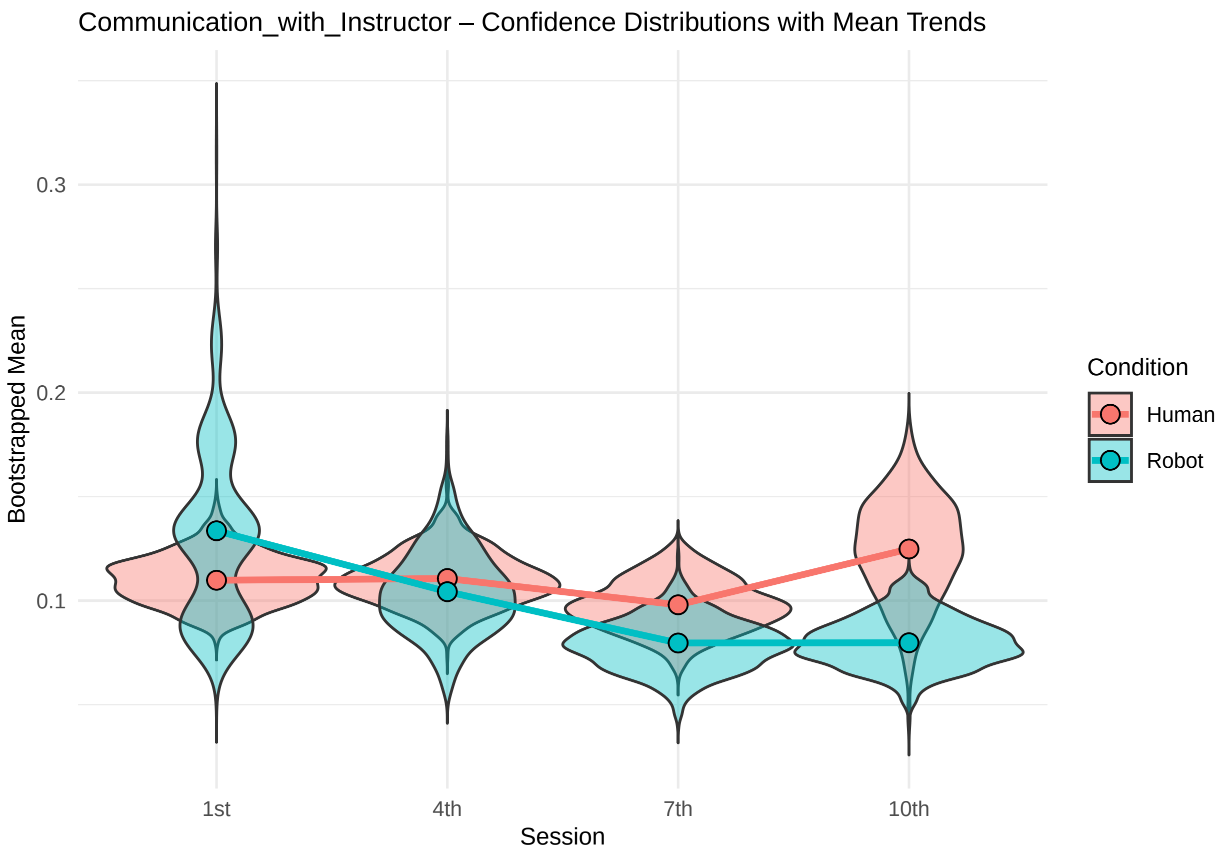


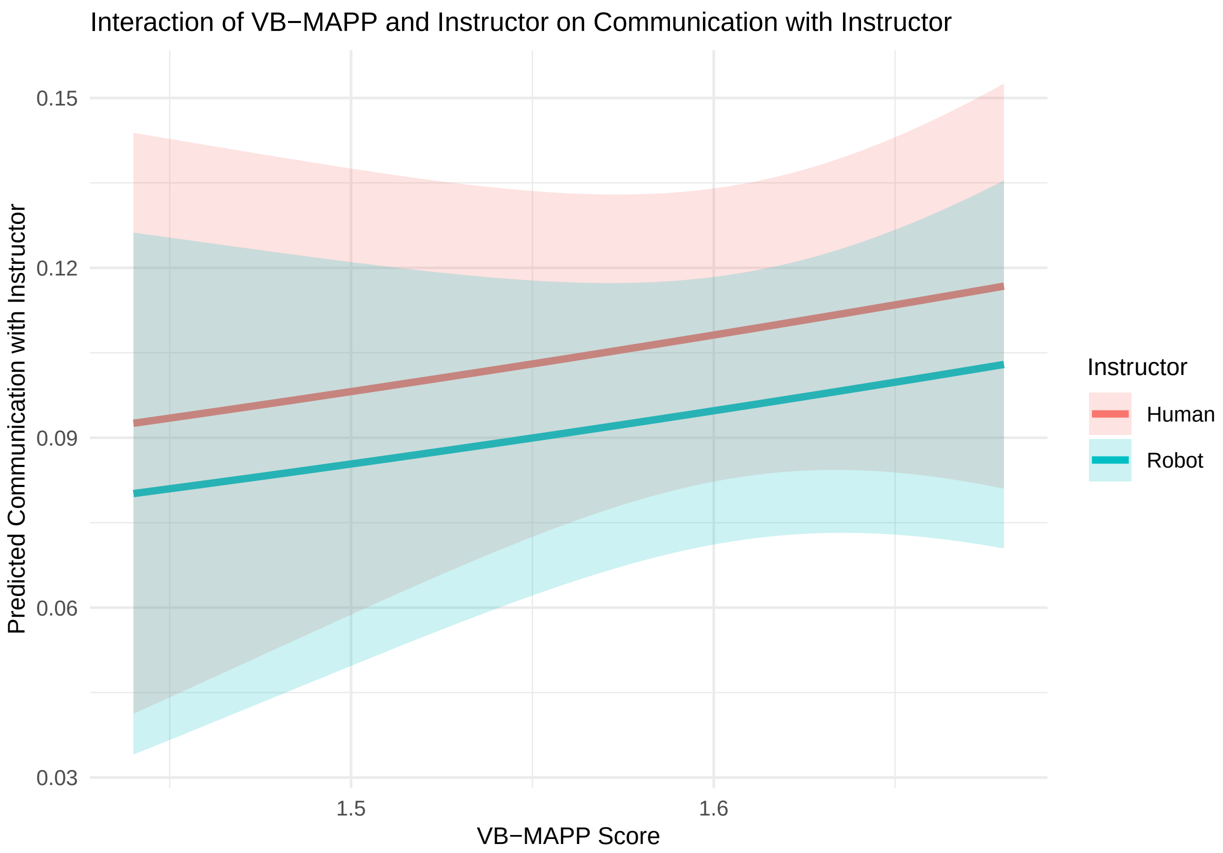
```
##
## ==============================
## Behavior: Affect_Negative
##
## Additive Model Summary:
## # A tibble: 4 x 6
##   term           estimate std.error statistic p.value stars
##   <chr>             <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)     -6.42       2.73     -2.35   0.0187 "*"
## 2 ConditionRobot   0.200      0.289     0.693  0.489  ""
## 3 time_c          -0.0267     0.126    -0.212  0.832  ""
## 4 VBMAPP           1.01       1.67      0.604  0.546  ""
##
## Interaction Model Summary:
## # A tibble: 6 x 6
##   term                  estimate std.error statistic p.value stars
##   <chr>                    <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)            -6.30       3.73   -1.69     0.0918 "."
## 2 ConditionRobot          0.0218     5.41    0.00403  0.997  ""
## 3 time_c                 -0.120      0.183  -0.658    0.510  ""
## 4 VBMAPP                  0.930      2.30    0.404    0.686  ""
## 5 ConditionRobot:time_c   0.175      0.251   0.699    0.485  ""
## 6 ConditionRobot:VBMAPP   0.110      3.36    0.0329   0.974  ""
```


Affect_Negative – Confidence Distributions with Mean Trends
0.08
0.06
0.04
0.02
0.00
Bootstrapped Mean
1st
4th
7th
10th
Session
Condition
Human
Robot

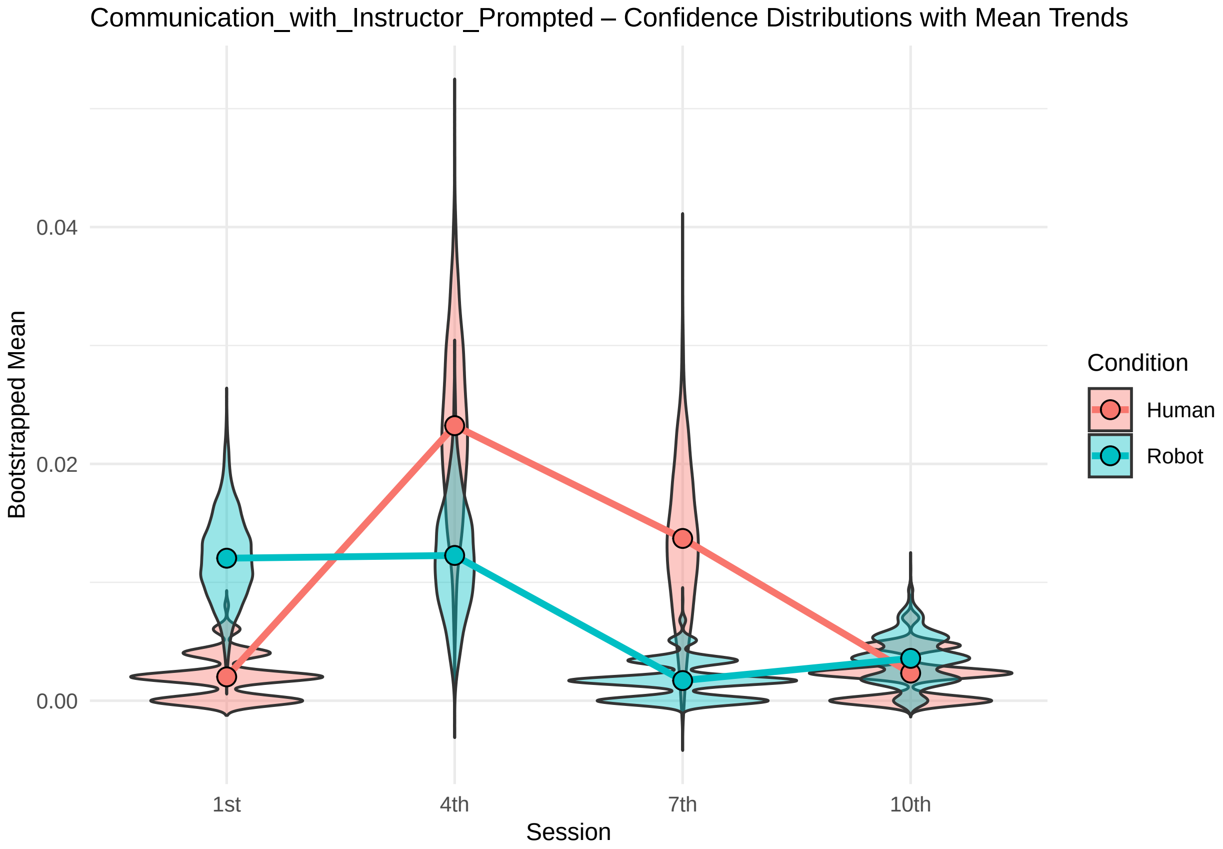


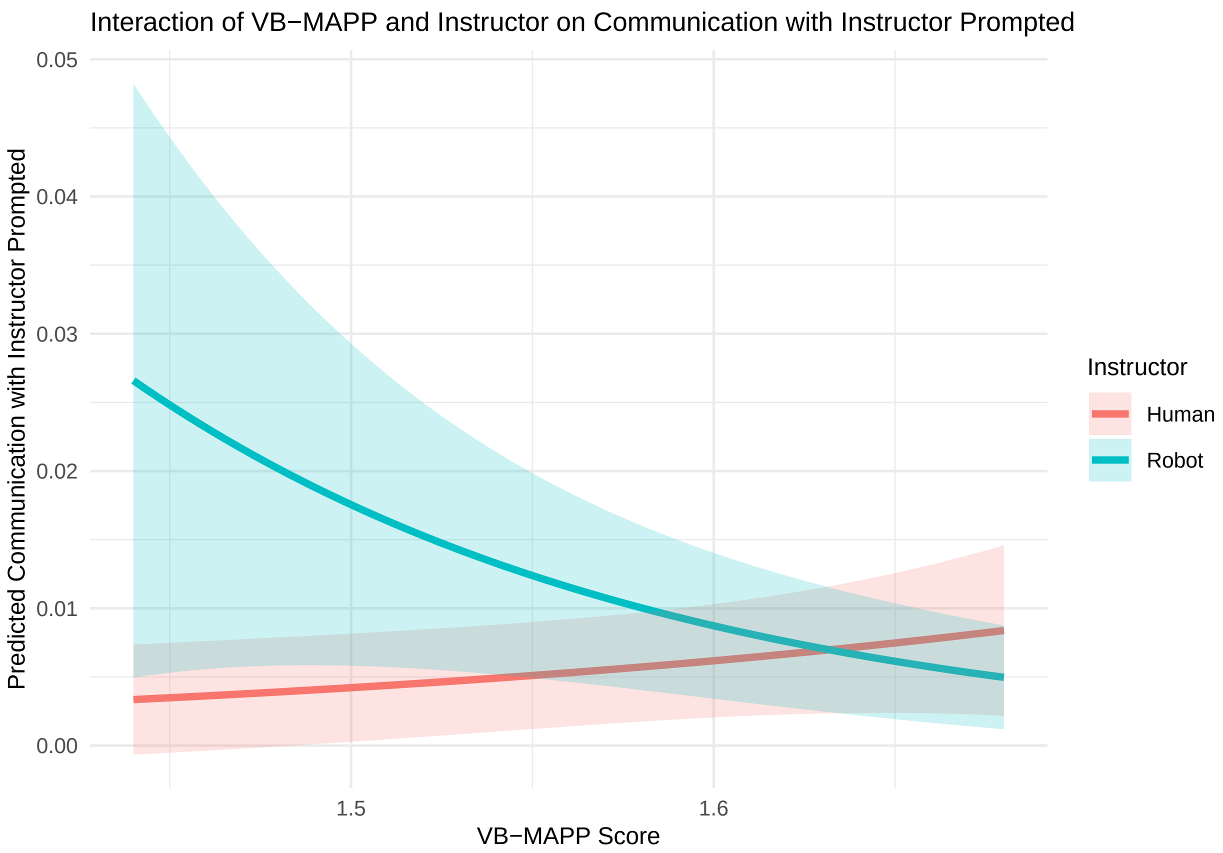
```
##
## ==============================
## Behavior: Communication_with_Instructor
##
## Additive Model Summary:
## # A tibble: 4 x 6
##   term          estimate std.error statistic p.value stars
##   <chr>            <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)    -3.81      2.25      -1.69   0.0901 "."
## 2 ConditionRobot -0.136     0.131     -1.04   0.298  ""
## 3 time_c         -0.0790    0.0594    -1.33   0.183  ""
## 4 VBMAPP          1.06      1.39       0.763  0.445  ""
##
## Interaction Model Summary:
## # A tibble: 6 x 6
##   term                  estimate std.error statistic p.value stars
##   <chr>                    <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)           -3.84       2.65     -1.45     0.147 ""
## 2 ConditionRobot        -0.254      2.54     -0.100    0.920 ""
## 3 time_c                 0.00200    0.0797    0.0251   0.980 ""
## 4 VBMAPP                 1.08       1.64      0.659    0.510 ""
## 5 ConditionRobot:time_c -0.170      0.116    -1.46     0.143 ""
## 6 ConditionRobot:VBMAPP  0.0668     1.57      0.0425   0.966 ""
```


Communication_with_Instructor – Confidence Distributions with Mean Trends
Bootstrapped Mean
0.3
0.2
0.1
1st
4th
7th
10th
Session
Condition
Human
Robot

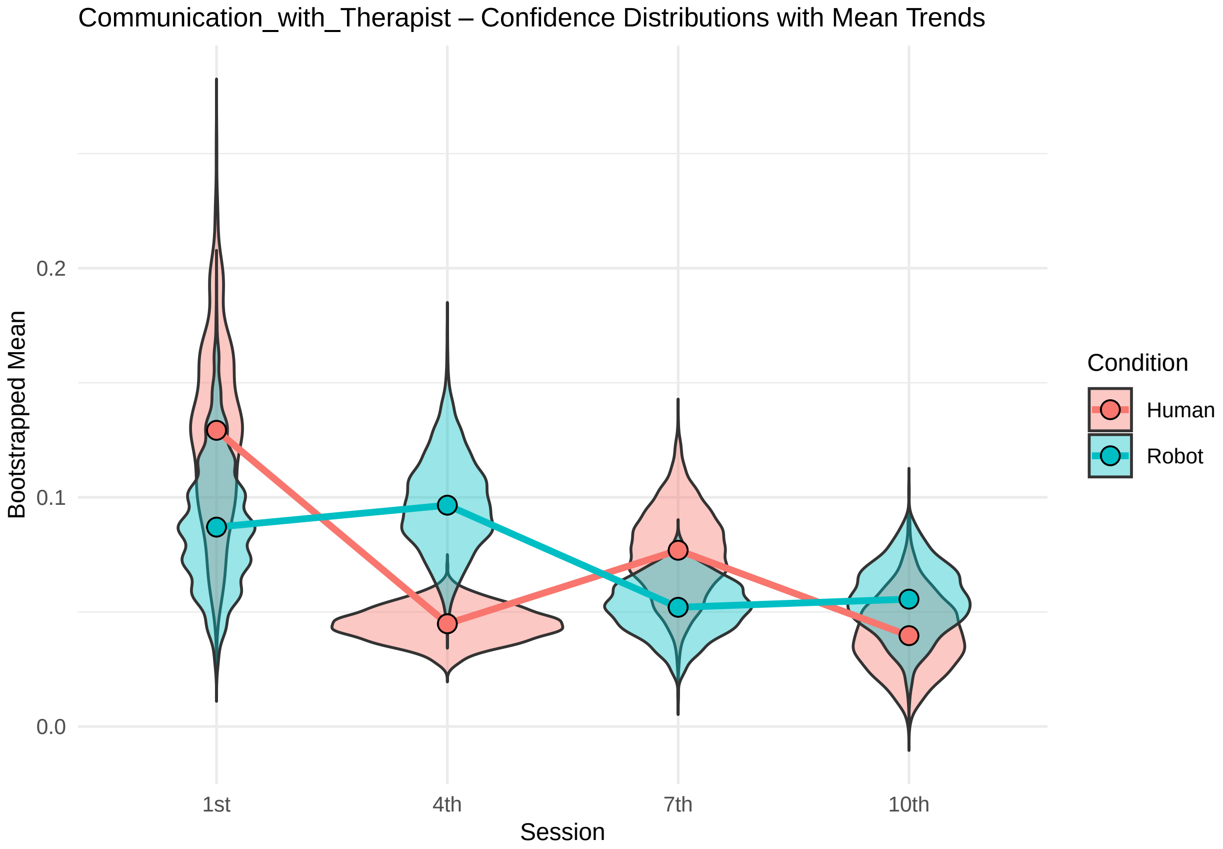


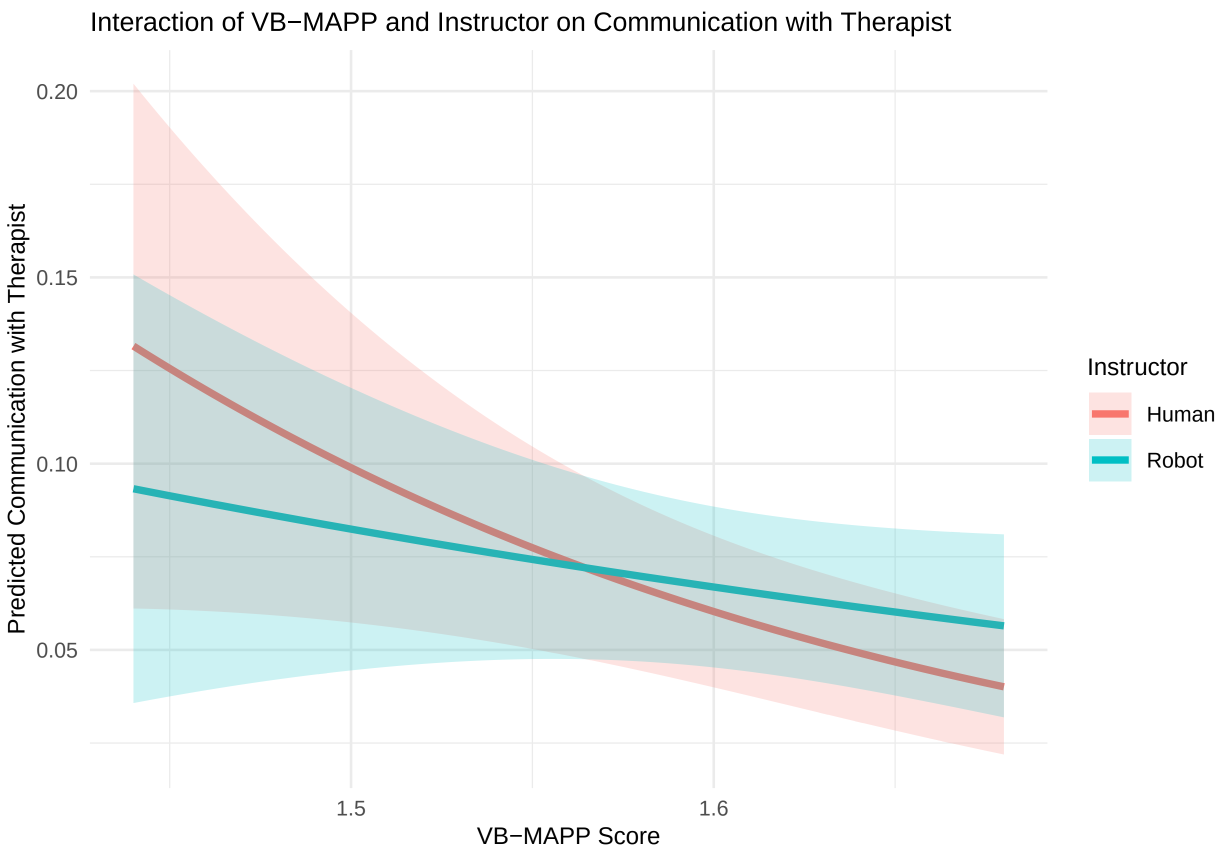
```
##
## ==============================
## Behavior: Communication_with_Instructor_Prompted
##
## Additive Model Summary:
## # A tibble: 4 x 6
##   term          estimate std.error statistic p.value stars
##   <chr>            <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)     -2.70      3.05     -0.884   0.377 ""
## 2 ConditionRobot   0.239     0.313     0.762   0.446 ""
## 3 time_c          -0.178     0.124    -1.44    0.150 ""
## 4 VBMAPP          -1.36      1.94     -0.702   0.482 ""
##
## Interaction Model Summary:
## # A tibble: 6 x 6
##   term                  estimate std.error statistic p.value stars
##   <chr>                    <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)           -11.2        4.62      -2.43  0.0152  "*"
## 2 ConditionRobot         17.8        5.88       3.03  0.00246 "**"
## 3 time_c                 -0.0287     0.153     -0.187 0.851   ""
## 4 VBMAPP                  3.84       2.85       1.35  0.178   ""
## 5 ConditionRobot:time_c  -0.337      0.242     -1.39  0.163   ""
## 6 ConditionRobot:VBMAPP -10.9        3.66      -2.99  0.00283 "**"
```


Communication_with_Instructor_Prompted – Confidence Distributions with Mean Trends
Bootstrapped Mean
0.04
0.02
0.00
1st
4th
7th
10th
Session
Condition
Human
Robot

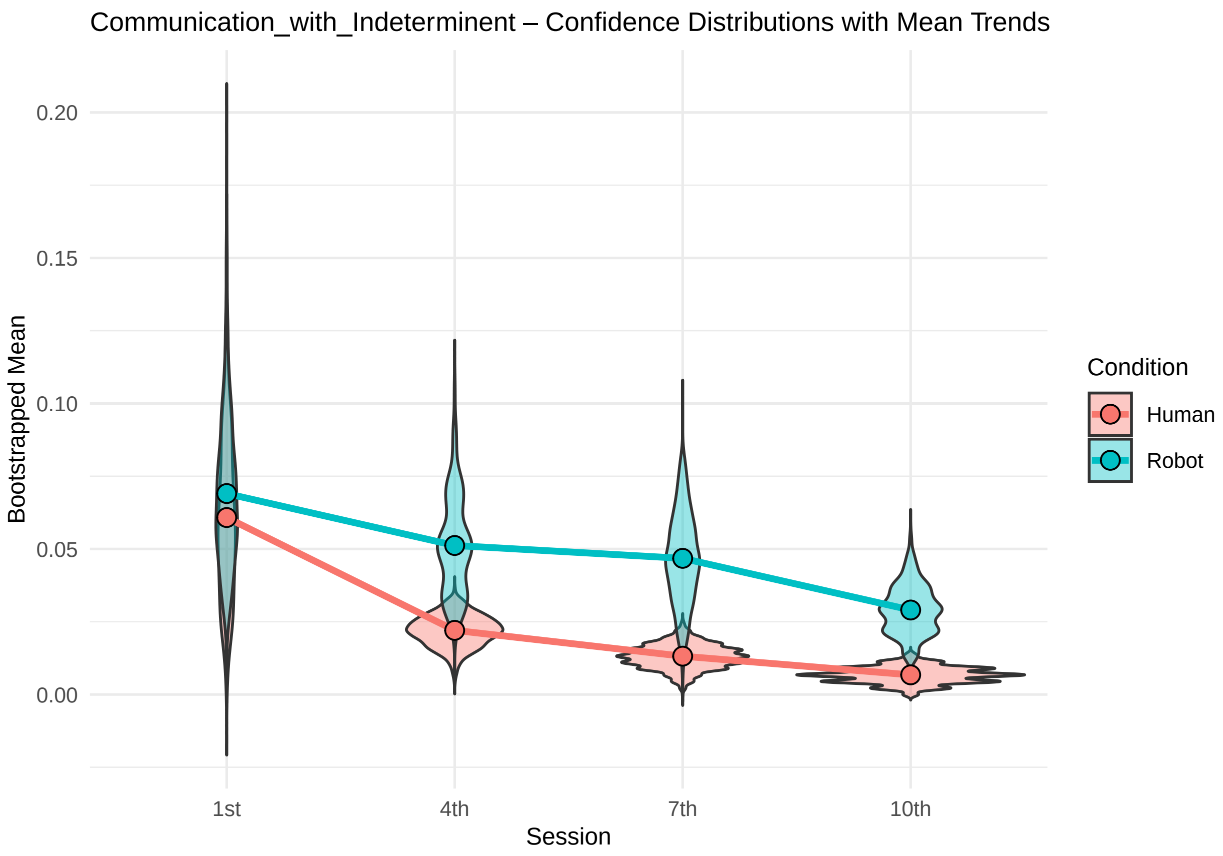


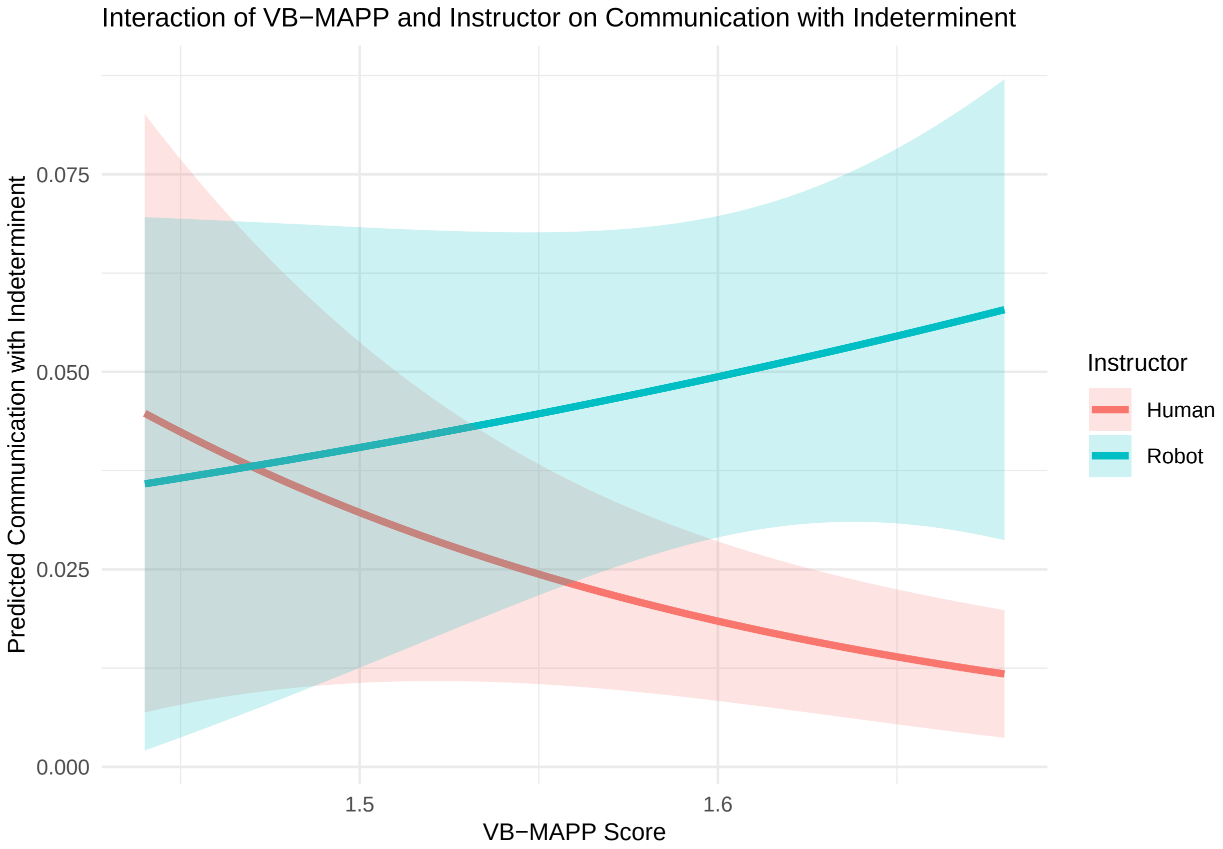
```
##
## ==============================
## Behavior: Communication_with_Therapist
##
## Additive Model Summary:
## # A tibble: 4 x 6
##   term           estimate std.error statistic   p.value stars
##   <chr>             <dbl>     <dbl>     <dbl>     <dbl> <chr>
## 1 (Intercept)      3.57       2.04      1.75  0.0805    "."
## 2 ConditionRobot   0.0322     0.226     0.143 0.887     ""
## 3 time_c          -0.450      0.113    -3.97  0.0000705 "***"
## 4 VBMAPP          -3.92       1.27     -3.07  0.00211   "**"
##
## Interaction Model Summary:
## # A tibble: 6 x 6
##   term                  estimate std.error statistic  p.value stars
##   <chr>                    <dbl>     <dbl>     <dbl>    <dbl> <chr>
## 1 (Intercept)              5.84      2.75       2.13 0.0334   "*"
## 2 ConditionRobot          -4.87      3.85      -1.26 0.207    ""
## 3 time_c                  -0.544     0.142     -3.83 0.000126 "***"
## 4 VBMAPP                  -5.37      1.73      -3.11 0.00186  "**"
## 5 ConditionRobot:time_c    0.208     0.187      1.11 0.266    ""
## 6 ConditionRobot:VBMAPP    3.11      2.41       1.29 0.197    ""
```


Communication_with_Therapist – Confidence Distributions with Mean Trends
Bootstrapped Mean
0.2
0.1
0.0
1st
4th
7th
10th
Session
Condition
Human
Robot

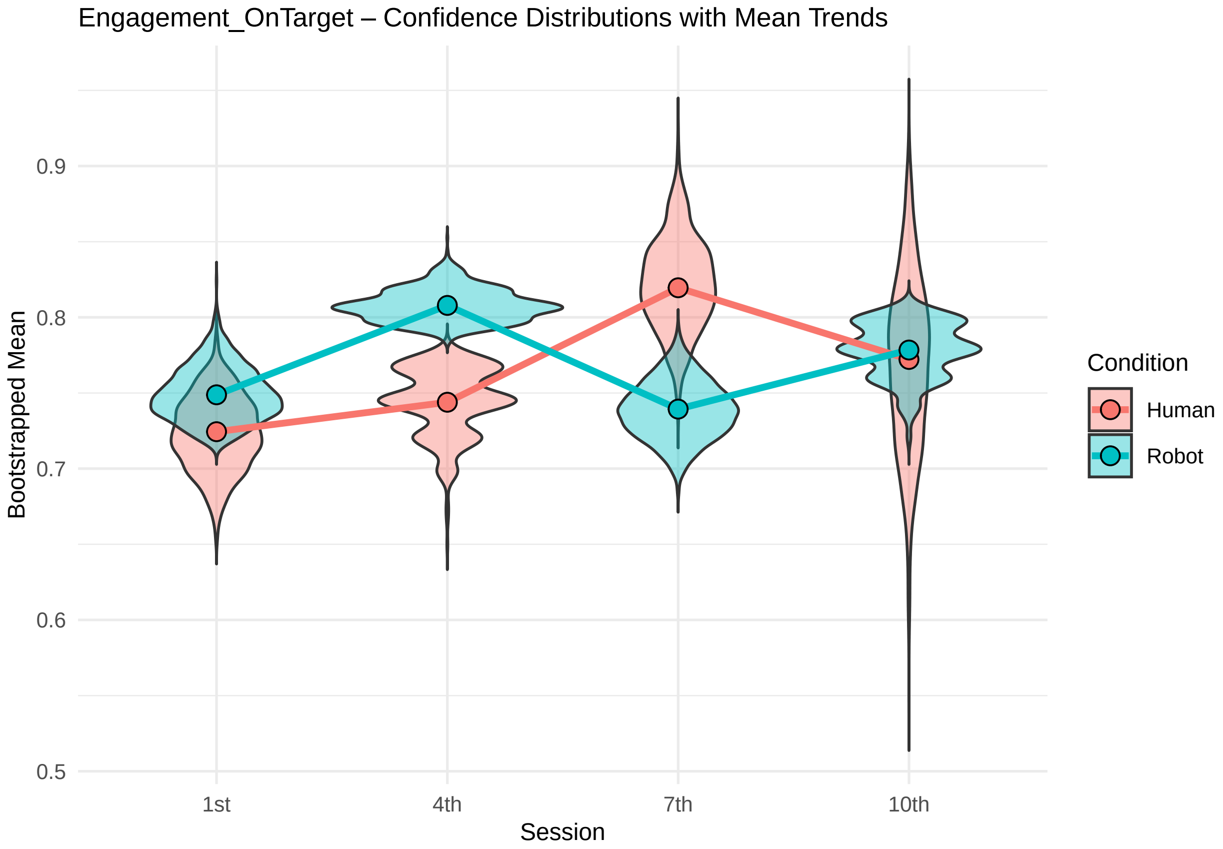


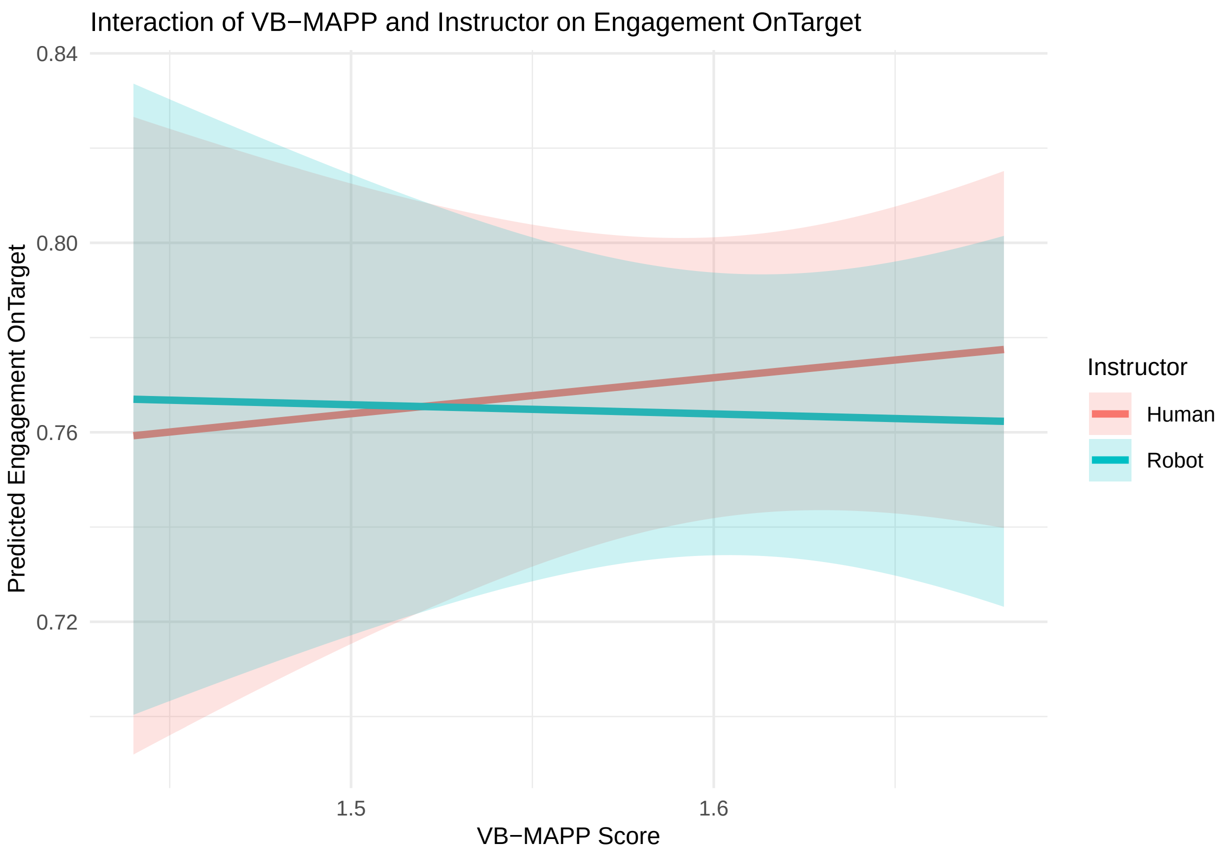
```
##
## ==============================
## Behavior: Communication_with_Indeterminent
##
## Additive Model Summary:
## # A tibble: 4 x 6
##   term          estimate std.error statistic  p.value stars
##   <chr>            <dbl>     <dbl>     <dbl>    <dbl> <chr>
## 1 (Intercept)     -1.96      3.09     -0.634 0.526    ""
## 2 ConditionRobot   1.05      0.294     3.58  0.000337 "***"
## 3 time_c          -0.226     0.122    -1.85  0.0638   "."
## 4 VBMAPP          -1.24      1.89     -0.658 0.510    ""
##
## Interaction Model Summary:
## # A tibble: 6 x 6
##   term                  estimate std.error statistic p.value stars
##   <chr>                    <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)             5.16       3.75      1.37   0.169  ""
## 2 ConditionRobot        -11.5        5.25     -2.19   0.0289 "*"
## 3 time_c                 -0.273      0.171    -1.59   0.111  ""
## 4 VBMAPP                 -5.71       2.36     -2.42   0.0153 "*"
## 5 ConditionRobot:time_c   0.0424     0.235     0.180  0.857  ""
## 6 ConditionRobot:VBMAPP   7.80       3.26      2.39   0.0168 "*"
```


Communication_with_Indeterminent – Confidence Distributions with Mean Trends
Bootstrapped Mean
0.20
0.15
0.10
0.05
0.00
1st
4th
7th
10th
Session
Condition
Human
Robot

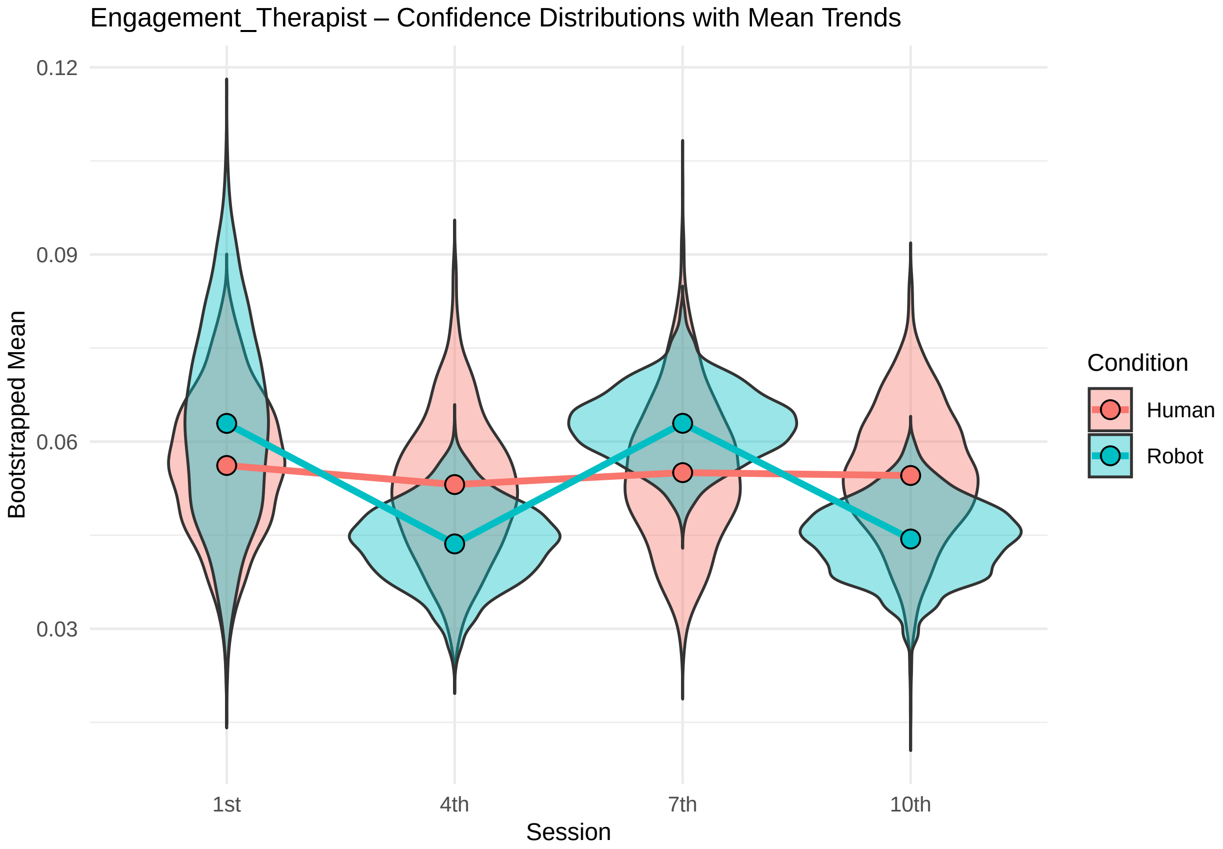


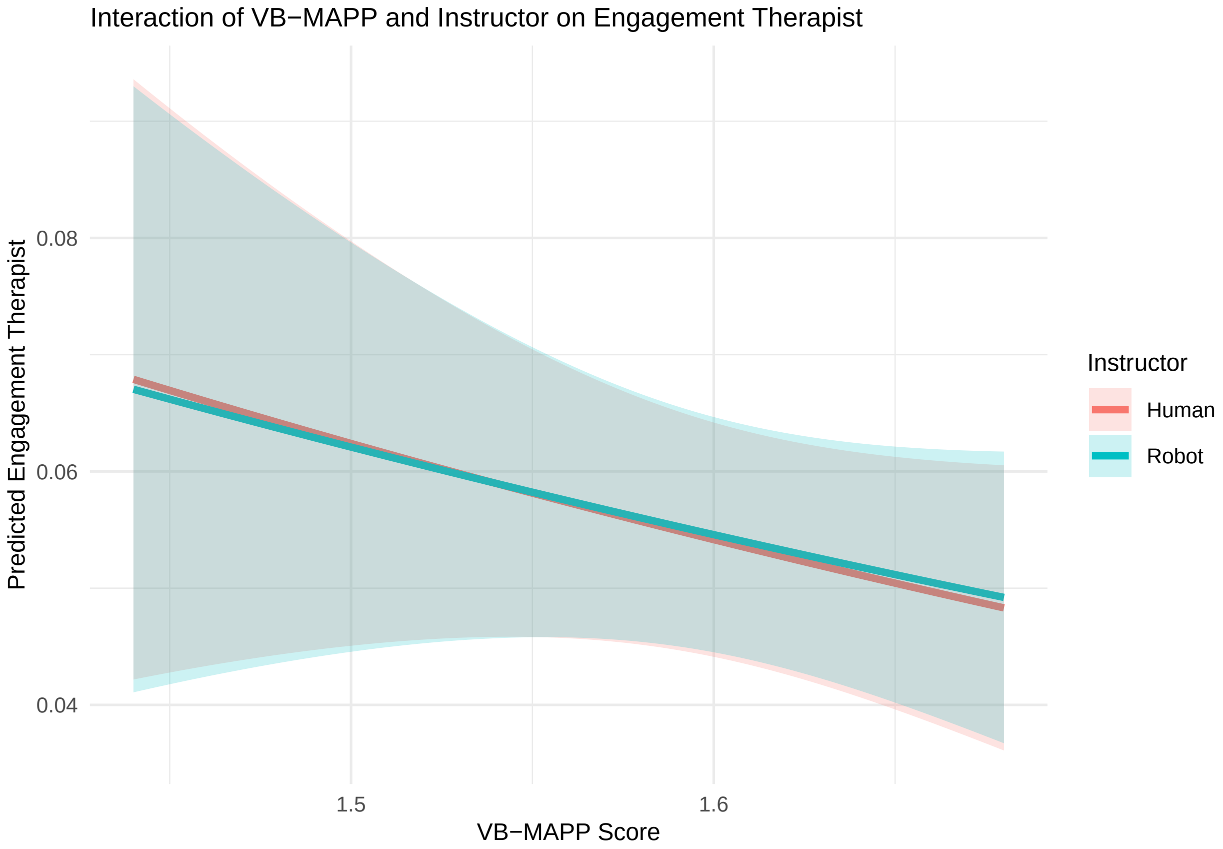
```
##
## ==============================
## Behavior: Engagement_OnTarget
##
## Additive Model Summary:
## # A tibble: 4 x 6
##   term           estimate std.error statistic p.value stars
##   <chr>             <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)      0.907     1.17       0.777   0.437 ""
## 2 ConditionRobot  -0.0382    0.122     -0.315   0.753 ""
## 3 time_c           0.0895    0.0551     1.63    0.104 ""
## 4 VBMAPP           0.191     0.723      0.264   0.792 ""
##
## Interaction Model Summary:
## # A tibble: 6 x 6
##   term                  estimate std.error statistic p.value stars
##   <chr>                    <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)              0.535    1.60       0.334  0.738  ""
## 2 ConditionRobot           0.812    2.28       0.356  0.722  ""
## 3 time_c                   0.172    0.0773     2.23   0.0261 "*"
## 4 VBMAPP                   0.426    0.994      0.429  0.668  ""
## 5 ConditionRobot:time_c   -0.162    0.108     -1.51   0.132  ""
## 6 ConditionRobot:VBMAPP   -0.534    1.42      -0.377  0.706  ""
```


Engagement_OnTarget – Confidence Distributions with Mean Trends
Bootstrapped Mean
0.9
0.8
0.7
0.6
0.5
1st
4th
7th
10th
Session
Condition
Human
Robot

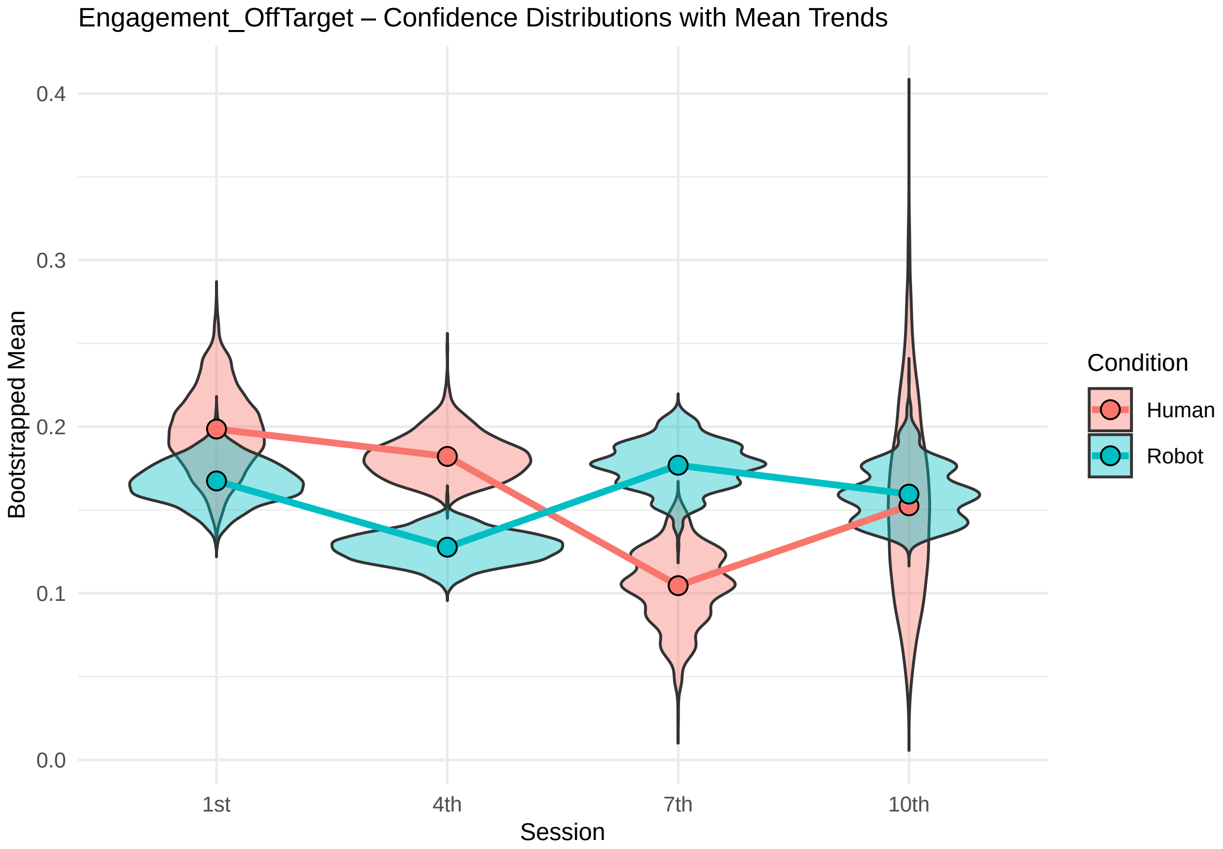


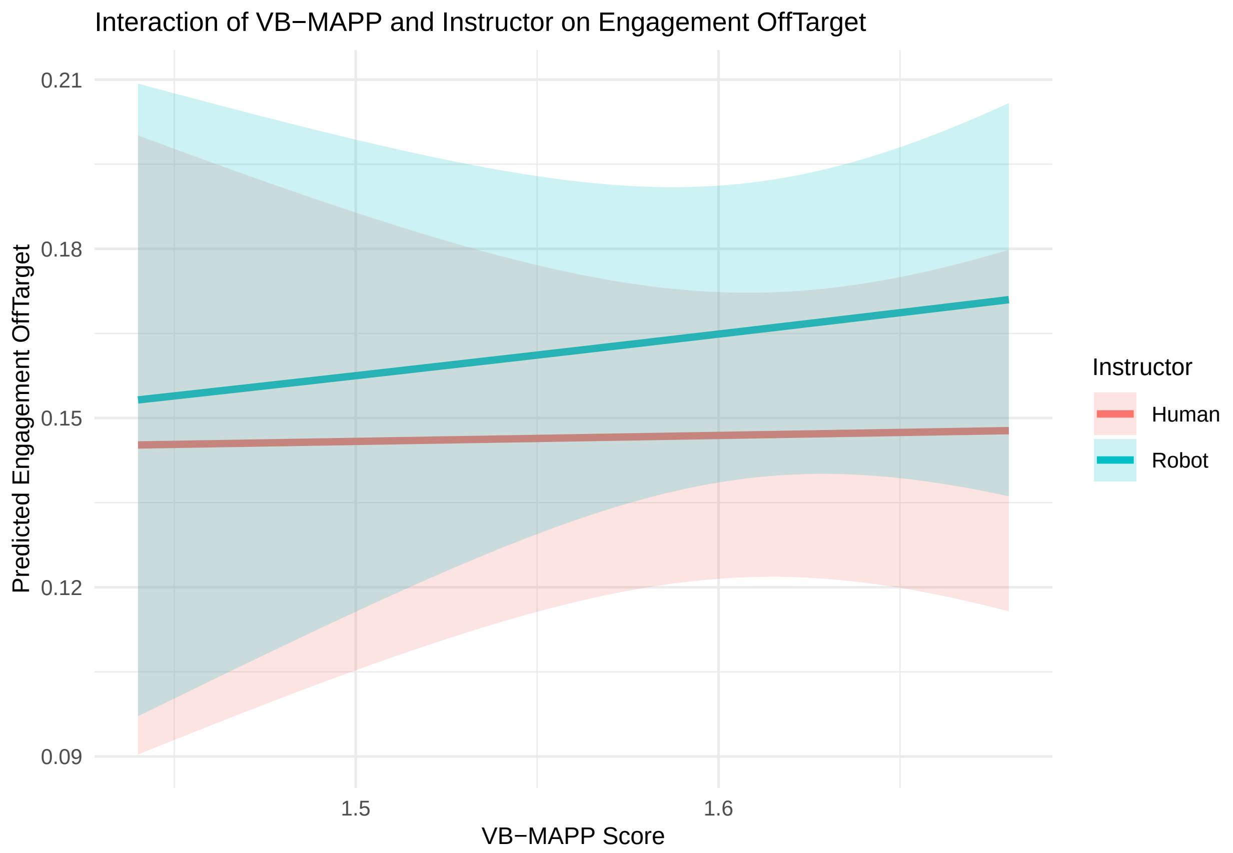
```
##
## ==============================
## Behavior: Engagement_Therapist
##
## Additive Model Summary:
## # A tibble: 4 x 6
##   term           estimate std.error statistic p.value stars
##   <chr>             <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)    -0.560      1.31     -0.428   0.669  ""
## 2 ConditionRobot  0.00880    0.137     0.0640  0.949  ""
## 3 time_c         -0.0160     0.0615   -0.260   0.795  ""
## 4 VBMAPP         -1.44       0.815    -1.77    0.0775 "."
##
## Interaction Model Summary:
## # A tibble: 6 x 6
##   term                  estimate std.error statistic p.value stars
##   <chr>                    <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)           -0.454      1.82     -0.249    0.803 ""
## 2 ConditionRobot        -0.211      2.61     -0.0807   0.936 ""
## 3 time_c                -0.00490    0.0876   -0.0559   0.955 ""
## 4 VBMAPP                -1.50       1.14     -1.32     0.186 ""
## 5 ConditionRobot:time_c -0.0219     0.123    -0.178    0.858 ""
## 6 ConditionRobot:VBMAPP  0.137      1.63      0.0840   0.933 ""
```


Engagement_Therapist – Confidence Distributions with Mean Trends
0.12
0.09
0.06
0.03
Bootstrapped Mean
1st
4th
7th
10th
Session
Condition
Human
Robot

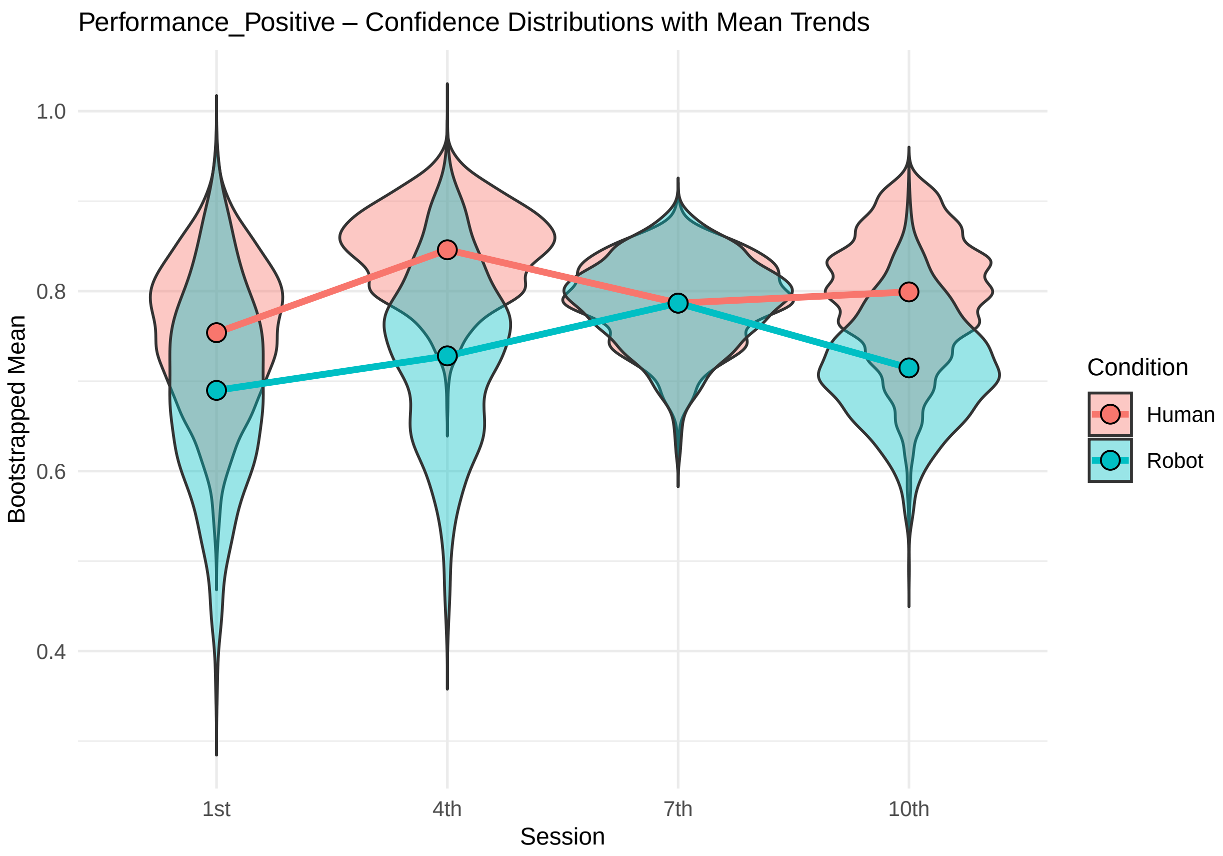


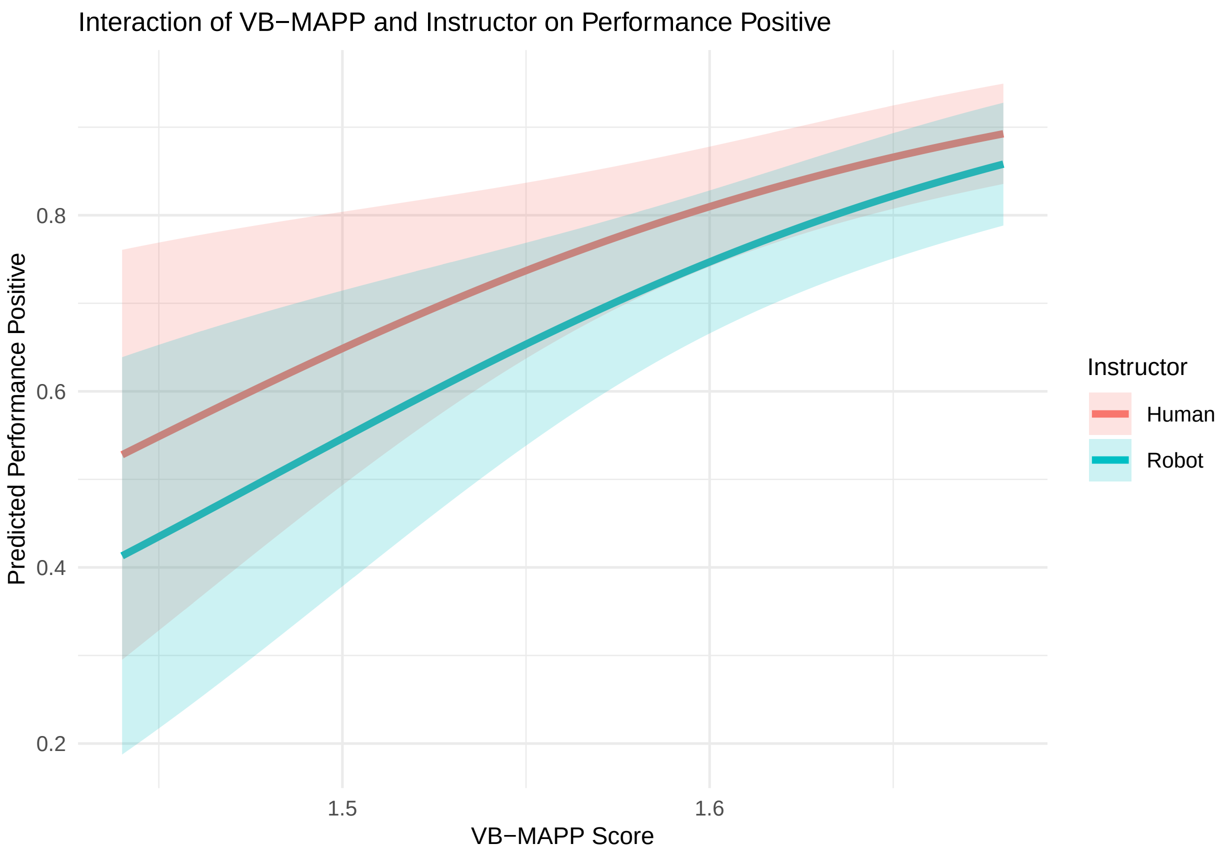
```
##
## ==============================
## Behavior: Engagement_OffTarget
##
## Additive Model Summary:
## # A tibble: 4 x 6
##   term          estimate std.error statistic p.value stars
##   <chr>            <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)     -2.13     1.39      -1.53   0.125  ""
## 2 ConditionRobot   0.114    0.145      0.790  0.430  ""
## 3 time_c          -0.120    0.0663    -1.81   0.0702 "."
## 4 VBMAPP           0.247    0.860      0.287  0.774  ""
##
## Interaction Model Summary:
## # A tibble: 6 x 6
##   term                  estimate std.error statistic p.value stars
##   <chr>                    <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)            -1.89      1.89     -1.00   0.315   ""
## 2 ConditionRobot         -0.601     2.66     -0.226  0.821   ""
## 3 time_c                 -0.269     0.0925   -2.91   0.00362 "**"
## 4 VBMAPP                  0.0847    1.17      0.0724 0.942   ""
## 5 ConditionRobot:time_c   0.284     0.127     2.24   0.0250  "*"
## 6 ConditionRobot:VBMAPP   0.461     1.65      0.280  0.779   ""
```


Engagement_OffTarget – Confidence Distributions with Mean Trends
Bootstrapped Mean
0.4
0.3
0.2
0.1
0.0
1st
4th
7th
10th
Session
Condition
Human
Robot



```
##
## ==============================
## Behavior: Performance_Positive
##
## Additive Model Summary:
## # A tibble: 4 x 6
##   term          estimate std.error statistic  p.value stars
##   <chr>            <dbl>     <dbl>     <dbl>    <dbl> <chr>
## 1 (Intercept)   -12.4       3.91      -3.18  0.00148  "**"
## 2 ConditionRobot  -0.374    0.161     -2.32  0.0201   "*"
## 3 time_c          0.0378    0.0730     0.517 0.605    ""
## 4 VBMAPP          8.67      2.43       3.57  0.000364 "***"
##
## Interaction Model Summary:
## # A tibble: 6 x 6
##   term                  estimate std.error statistic p.value stars
##   <chr>                    <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)           -11.9       4.16      -2.87  0.00414 "**"
## 2 ConditionRobot         -1.34      2.77      -0.482 0.630   ""
## 3 time_c                  0.0276    0.106      0.261 0.794   ""
## 4 VBMAPP                  8.35      2.59       3.22  0.00126 "**"
## 5 ConditionRobot:time_c   0.0190    0.144      0.132 0.895   ""
## 6 ConditionRobot:VBMAPP   0.606     1.74       0.348 0.728   ""
```


Performance_Positive – Confidence Distributions with Mean Trends
Bootstrapped Mean
1.0
0.8
0.6
0.4
1st
4th
7th
10th
Session
Condition
Human
Robot



2.2 Results: Additive Model Comparison Between Human and Robot Instructors

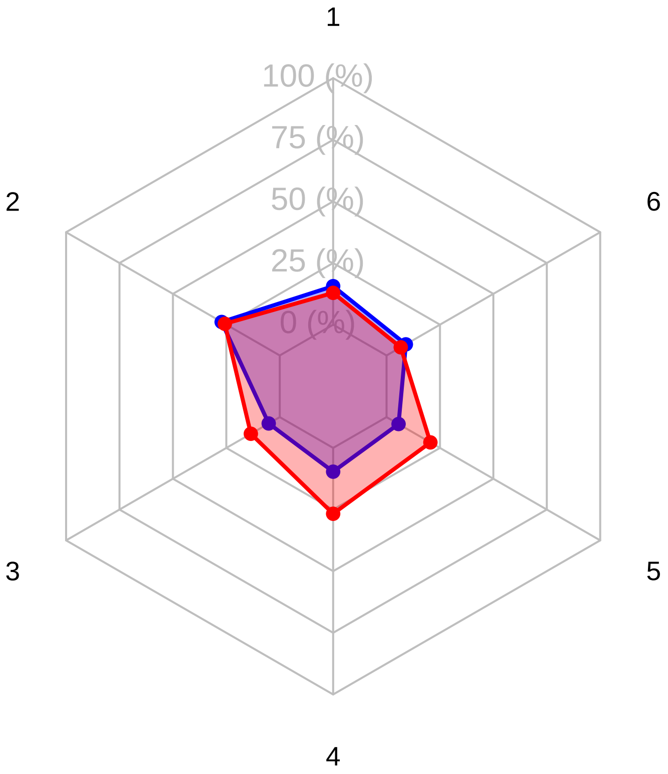
To evaluate the effect of instructor type on children's learning behaviors, we fitted additive generalized linear mixed models (GLMMs) with `Condition` (Human vs. Robot) as a fixed effect, controlling for session time (`time_c`) and individual skill level (`VB-MAPP`). A random intercept for `Subject` accounted for repeated measures. All dependent variables were modeled with a beta distribution to account for their bounded, proportional nature. Below we report the estimated effect of the Robot condition (`ConditionRobot`) relative to the Human condition for each behavioral outcome.

First, let's take a look at the hexagonal plots from each metric:

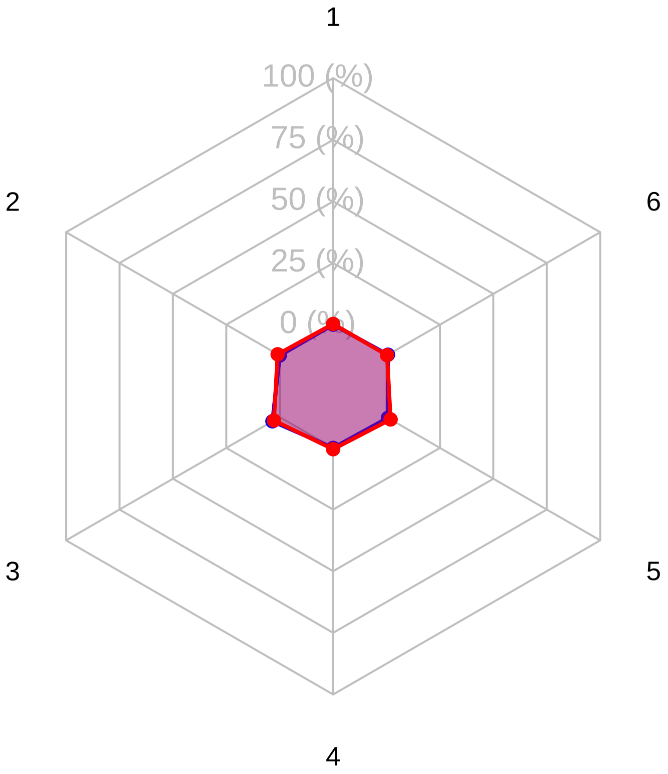
```
if (!requireNamespace("fmsb", quietly = TRUE)) install.packages("fmsb")
library(fmsb)
library(dplyr)
library(tidyr)

# Load data
df <- read.csv("~/GitHub/Circle-Time-Data-Analyses/CircleTimeData-VBMAPP.csv")
df$Condition <- factor(df$Condition, levels = c(1, 2), labels = c("Human", "Robot"))

# List of variables to plot
metrics <- c(
  "Affect_Positive", "Affect_Negative",
  "Communication_with_Instructor", "Communication_with_Instructor_Prompted",
```

```r
  "Communication_with_Therapist", "Communication_with_Indeterminent",
  "Engagement_OnTarget", "Engagement_Therapist", "Engagement_OffTarget",
  "Performance_Positive"
)

# Loop through each metric
for (metric in metrics) {
  # Compute average by Subject and Condition
  avg_df <- df %>%
    group_by(Subject, Condition) %>%
    summarise(value = mean(.data[[metric]], na.rm = TRUE), .groups = "drop") %>%
    pivot_wider(names_from = Condition, values_from = value)

  # Skip if incomplete
  if (any(!c("Human", "Robot") %in% colnames(avg_df))) next

  # Sort for consistency
  avg_df <- avg_df[order(avg_df$Subject), ]
  human <- avg_df$Human
  robot <- avg_df$Robot
  subjects <- avg_df$Subject

  # Create radar data frame
  data <- as.data.frame(rbind(
    rep(1, length(human)),
    rep(0, length(human)),
    human,
    robot
  ))
  colnames(data) <- subjects
  rownames(data) <- c("Max", "Min", "Human", "Robot")

  # Adjust title font size
  oldpar <- par(no.readonly = TRUE)
  par(cex.main = 0.9)

  # Plot
  radarchart(data,
    axistype = 1,
    pcol = c("blue", "red"),
    pfcol = c(rgb(0, 0, 1, 0.3), rgb(1, 0, 0, 0.3)),
    plwd = 2,
    plty = 1,
    cglcol = "grey",
    cglty = 1,
    axislabcol = "grey",
    vlcex = 0.8,
    title = paste(metric, "by Participant: Human (blue) vs. Robot (red)")
  )

  par(oldpar)
}
```

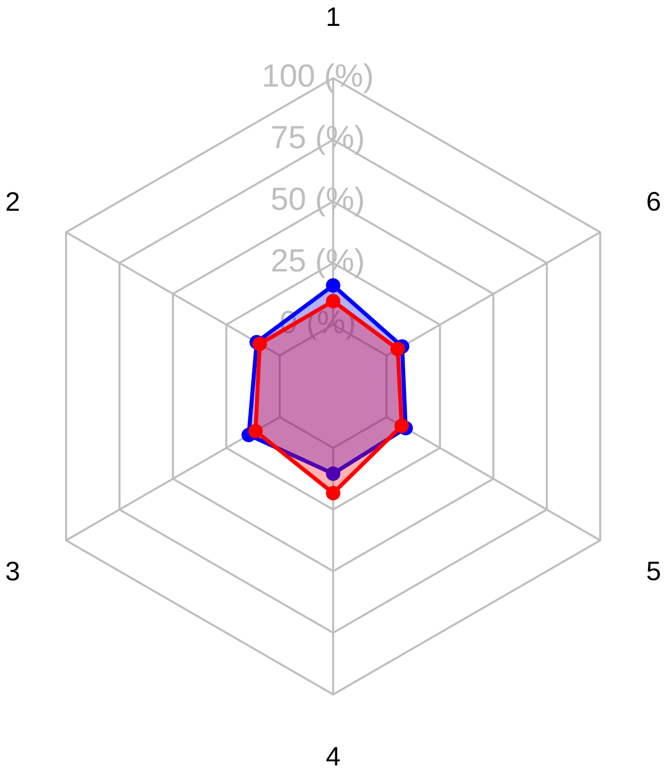
Affect_Positive by Participant: Human (blue) vs. Robot (red)



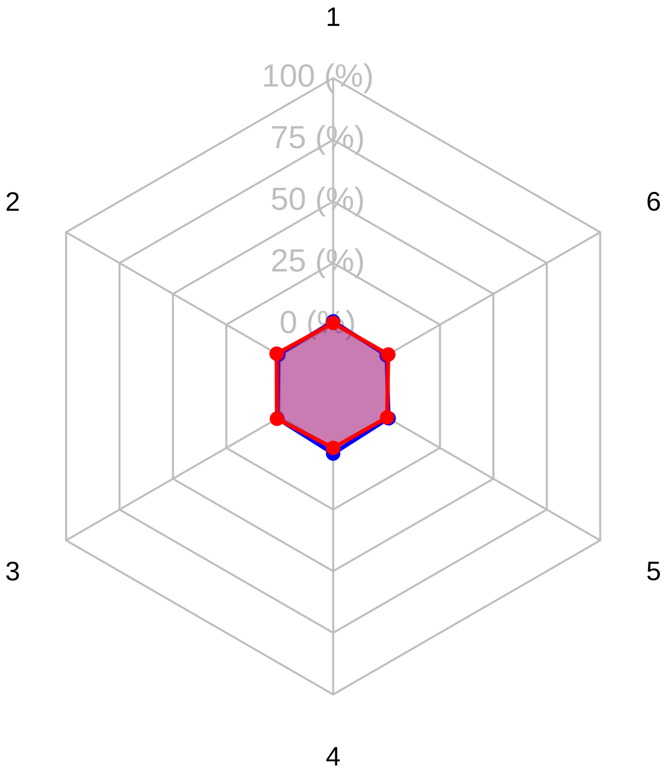
Affect_Negative by Participant: Human (blue) vs. Robot (red)



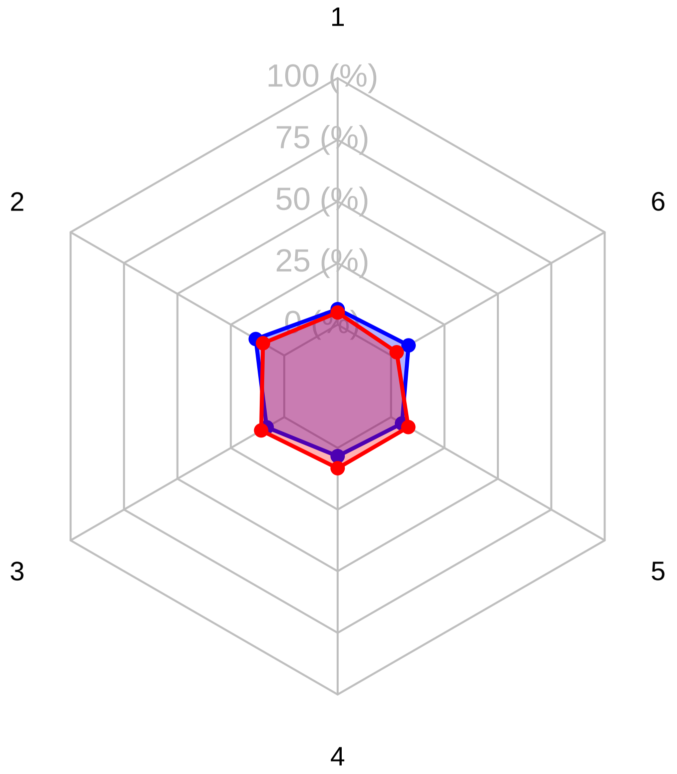
Communication_with_Instructor by Participant: Human (blue) vs. Robot (red)

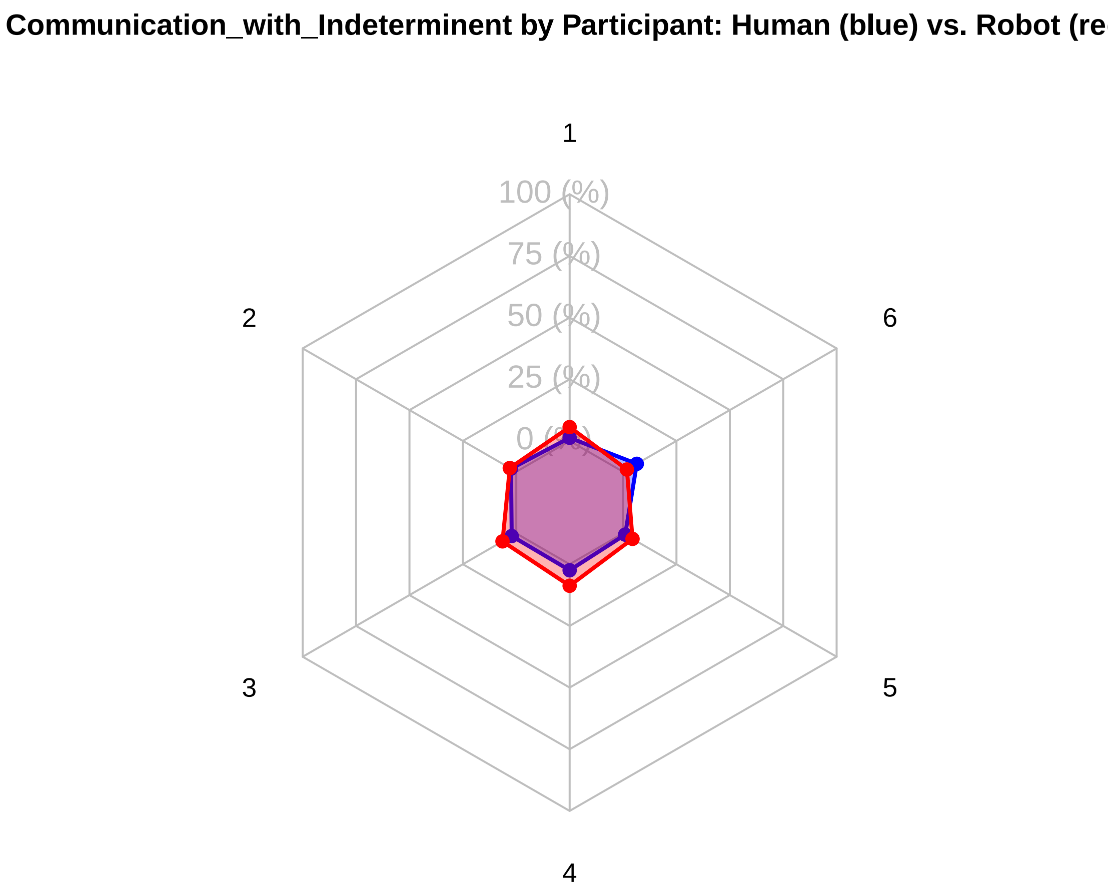


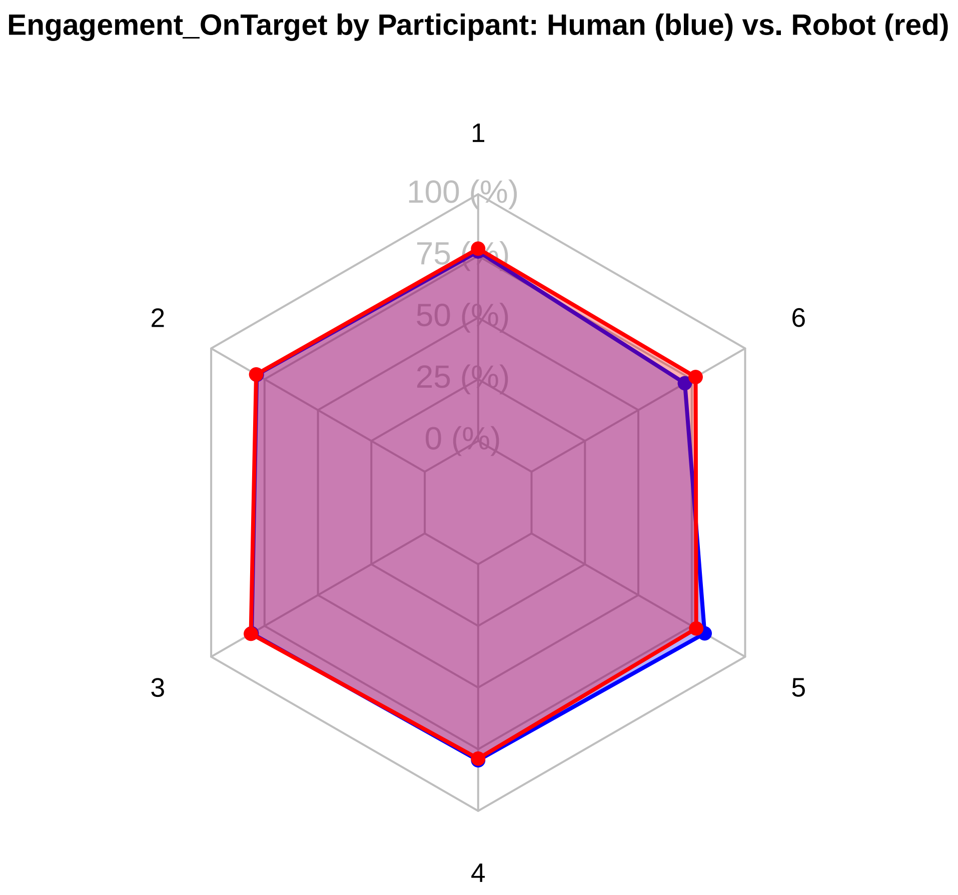
Communication_with_Instructor_Prompted by Participant: Human (blue) vs. Robot



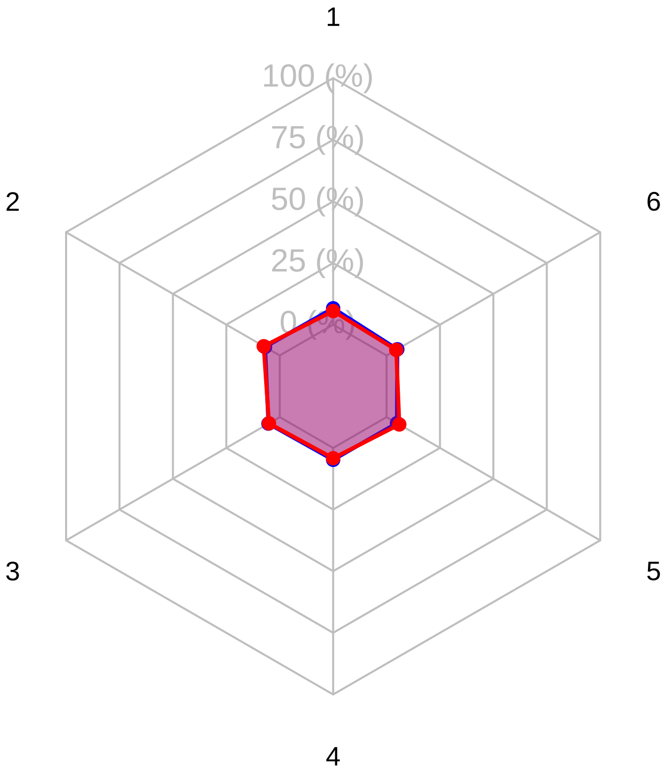
Communication_with_Therapist by Participant: Human (blue) vs. Robot (red)

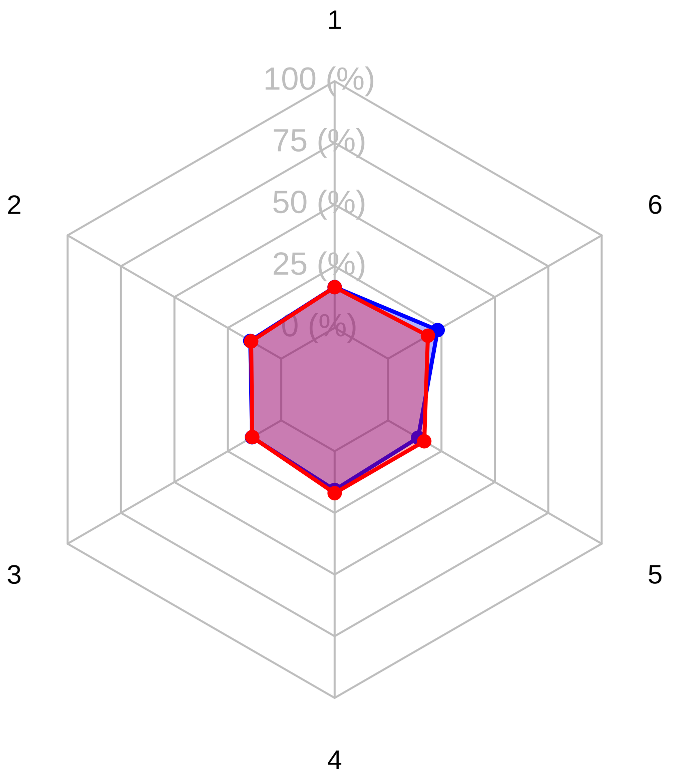



Communication_with_Indeterminent by Participant: Human (blue) vs. Robot (re
1
2
3
4
5
6
100 (%)
75 (%)
50 (%)
25 (%)
0 (%)

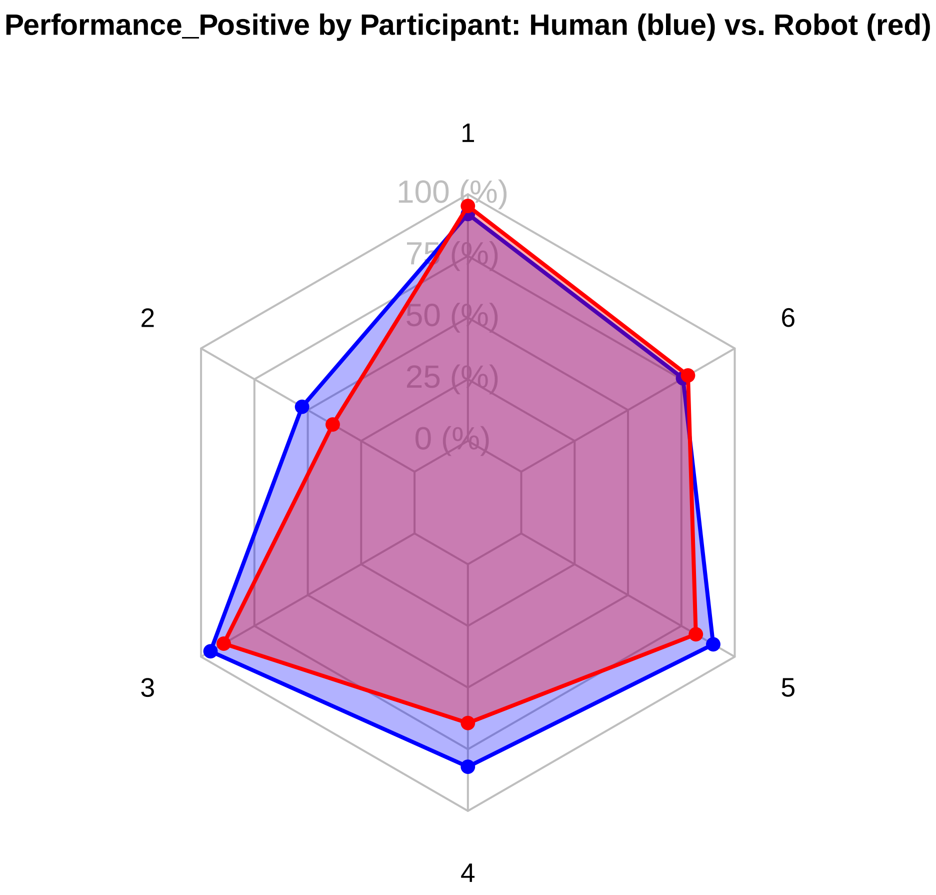

Engagement_OnTarget by Participant: Human (blue) vs. Robot (red)
1
2
3
4
5
6
100 (%)
75 (%)
50 (%)
25 (%)
0 (%)

Engagement_Therapist by Participant: Human (blue) vs. Robot (red)





Engagement_OffTarget by Participant: Human (blue) vs. Robot (red)

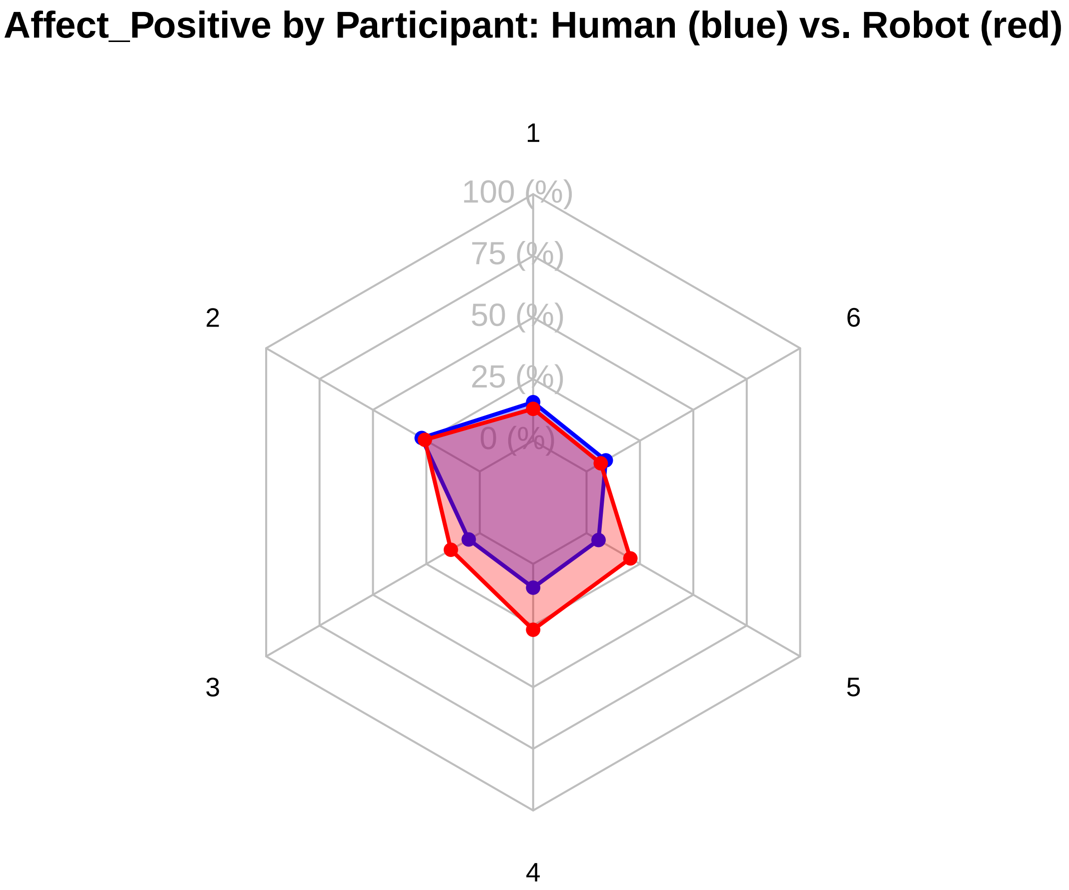


2.2.1 Affect

- **Positive Affect** was significantly higher during robot-led instruction compared to human-led instruction:
  $= 0.435, SE = 0.151, z = 2.89, p = 0.0039$
  This suggests that children expressed more positive emotional behaviors (e.g., smiling, clapping) during robot sessions.

```r
# Install and load fmsb
if (!requireNamespace("fmsb", quietly = TRUE)) {
  install.packages("fmsb")
}
library(fmsb)
library(dplyr)
library(tidyr)
```

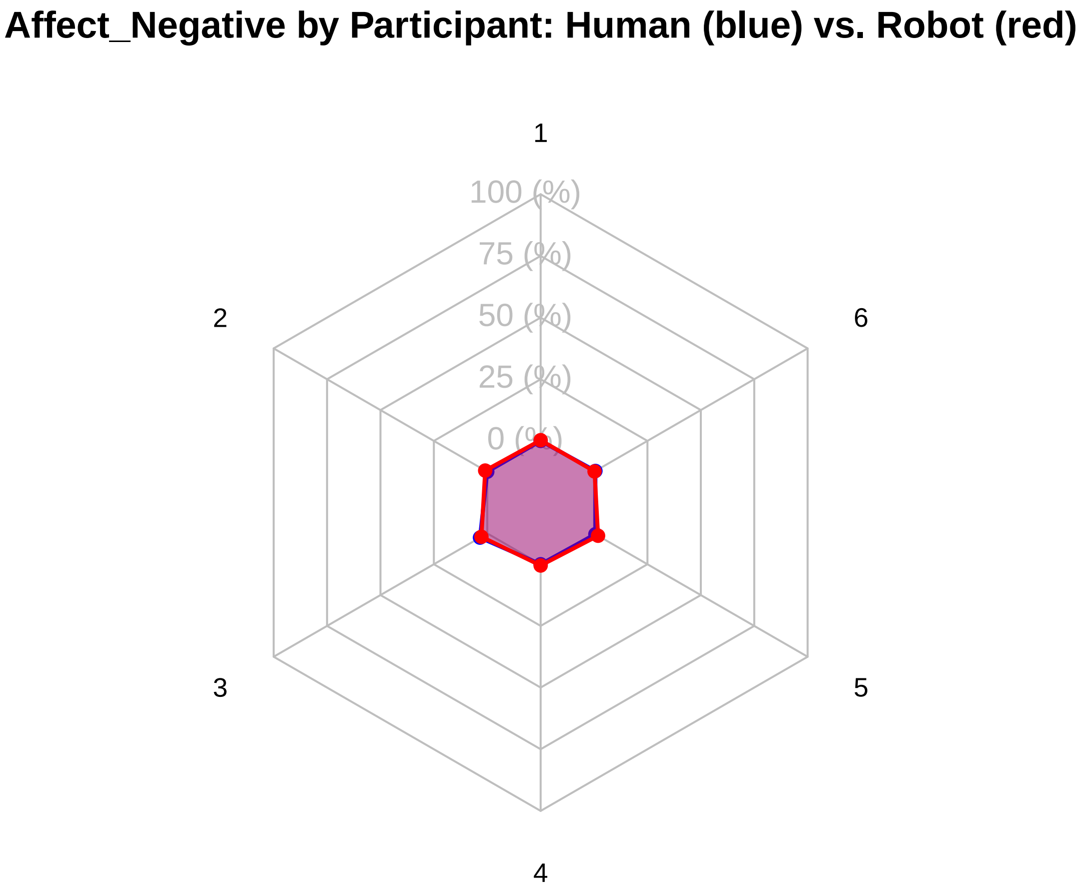
```r
# Load your dataset
df <- read.csv("~/GitHub/Circle-Time-Data-Analyses/CircleTimeData-VBMAPP.csv")

# Ensure Condition is labeled properly
df$Condition <- factor(df$Condition, levels = c(1, 2), labels = c("Human", "Robot"))
df$time_c <- scale(df$time, center = TRUE, scale = FALSE)

# Compute mean Affect_Positive per participant per condition
avg_affect <- df %>%
  group_by(Subject, Condition) %>%
  summarise(Affect_Positive = mean(Affect_Positive, na.rm = TRUE), .groups = "drop") %>%
  pivot_wider(names_from = Condition, values_from = Affect_Positive)

# Sort by Subject ID
avg_affect <- avg_affect[order(avg_affect$Subject), ]

# Extract values
human <- avg_affect$Human
robot <- avg_affect$Robot
subjects <- avg_affect$Subject

# Create data frame for radar chart
data <- as.data.frame(rbind(
  rep(1, length(human)),  # max
  rep(0, length(human)),  # min
  human,
  robot
))
colnames(data) <- subjects
rownames(data) <- c("Max", "Min", "Human", "Robot")

# Plot radar chart
radarchart(data,
  axistype = 1,
  pcol = c("blue", "red"),
  pfcol = c(rgb(0, 0, 1, 0.3), rgb(1, 0, 0, 0.3)),
  plwd = 2,
  plty = 1,
  cglcol = "grey",
  cglty = 1,
  axislabcol = "grey",
  vlcex = 0.8,
  title = "Affect_Positive by Participant: Human (blue) vs. Robot (red)"
)
```



- **Negative Affect** did not differ significantly between conditions:
  $= 0.200, SE = 0.289, z = 0.69, p = 0.489$

```r
# Install and load fmsb
if (!requireNamespace("fmsb", quietly = TRUE)) {
  install.packages("fmsb")
}
library(fmsb)
library(dplyr)
library(tidyr)

# Load your dataset
df <- read.csv("~/GitHub/Circle-Time-Data-Analyses/CircleTimeData-VBMAPP.csv")

# Ensure Condition is labeled properly
df$Condition <- factor(df$Condition, levels = c(1, 2), labels = c("Human", "Robot"))
```

```r
df$time_c <- scale(df$time, center = TRUE, scale = FALSE)

# Compute mean Affect_Positive per participant per condition
avg_affect <- df %>%
  group_by(Subject, Condition) %>%
  summarise(Affect_Negative = mean(Affect_Negative, na.rm = TRUE), .groups = "drop") %>%
  pivot_wider(names_from = Condition, values_from = Affect_Negative)

# Sort by Subject ID
avg_affect <- avg_affect[order(avg_affect$Subject), ]

# Extract values
human <- avg_affect$Human
robot <- avg_affect$Robot
subjects <- avg_affect$Subject

# Create data frame for radar chart
data <- as.data.frame(rbind(
  rep(1, length(human)),  # max
  rep(0, length(human)),  # min
  human,
  robot
))
colnames(data) <- subjects
rownames(data) <- c("Max", "Min", "Human", "Robot")

# Plot radar chart
radarchart(data,
  axistype = 1,
  pcol = c("blue", "red"),
  pfcol = c(rgb(0, 0, 1, 0.3), rgb(1, 0, 0, 0.3)),
  plwd = 2,
  plty = 1,
  cglcol = "grey",
  cglty = 1,
  axislabcol = "grey",
  vlcex = 0.8,
  title = "Affect_Negative by Participant: Human (blue) vs. Robot (red)"
)
```

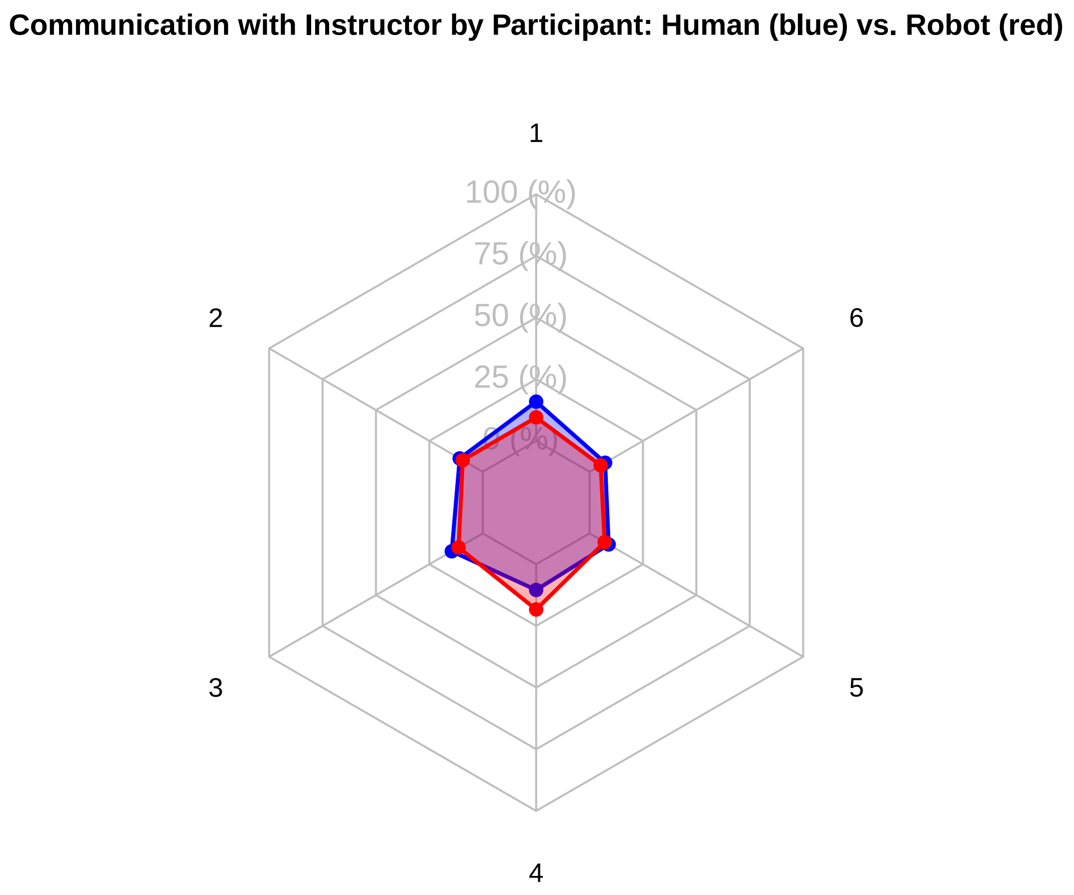


2.2.2 Communication

- **Communication with Instructor** showed no significant difference between conditions: *= -0.136, SE = 0.131, z = -1.04, p = 0.298*

```r
# Install and load fmsb
if (!requireNamespace("fmsb", quietly = TRUE)) {
  install.packages("fmsb")
}
library(fmsb)
library(dplyr)
library(tidyr)

# Load dataset
df <- read.csv("~/GitHub/Circle-Time-Data-Analyses/CircleTimeData-VBMAPP.csv")
df$Condition <- factor(df$Condition, levels = c(1, 2), labels = c("Human", "Robot"))
```

```r
# Select variable
comm_var <- "Communication_with_Instructor"

# Compute mean per participant/condition
avg_comm <- df %>%
  group_by(Subject, Condition) %>%
  summarise(value = mean(.data[[comm_var]], na.rm = TRUE), .groups = "drop") %>%
  pivot_wider(names_from = Condition, values_from = value)

avg_comm <- avg_comm[order(avg_comm$Subject), ]

# Prepare values
human <- avg_comm$Human
robot <- avg_comm$Robot
subjects <- avg_comm$Subject

data <- as.data.frame(rbind(
  rep(1, length(human)),
  rep(0, length(human)),
  human,
  robot
))
colnames(data) <- subjects
rownames(data) <- c("Max", "Min", "Human", "Robot")

# Reduce title font size
oldpar <- par(no.readonly = TRUE)
par(cex.main = 0.9)  # smaller title font

radarchart(data,
  axistype = 1,
  pcol = c("blue", "red"),
  pfcol = c(rgb(0, 0, 1, 0.3), rgb(1, 0, 0, 0.3)),
  plwd = 2,
  plty = 1,
  cglcol = "grey",
  cglty = 1,
  axislabcol = "grey",
  vlcex = 0.8,
  title = "Communication with Instructor by Participant: Human (blue) vs. Robot (red)"
)
```



```
par(oldpar)  # reset plotting parameters
```

- **Instructor-Prompted Communication** was also not significantly affected:
  = 0.239, SE = 0.313, z = 0.76, p = 0.446
- **Communication with Therapist or Peers** remained similar across conditions:
  = 0.032, SE = 0.226, z = 0.14, p = 0.887
- **Indeterminate Communication** (where the communication partner was unclear) was significantly higher under the robot condition:
  = 1.05, SE = 0.294, z = 3.58, p = 0.0003
  This may reflect increased ambiguity in children's social orientation during robot-led sessions.

2.2.3 Engagement

- **On-Target Engagement** (attention to instructor or screen) did not differ significantly:
  = -0.038, SE = 0.122, z = -0.32, p = 0.753

- **Engagement with Therapist or Peers** showed no condition effect:
  = 0.009, SE = 0.137, z = 0.06, p = 0.949
- **Off-Target Engagement** (distraction or disengagement) also did not differ:
  = 0.114, SE = 0.145, z = 0.79, p = 0.430

2.2.4 Performance

- **Positive Performance** was significantly lower in the robot condition:
  = -0.374, SE = 0.161, z = -2.32, p = 0.020
  This indicates children demonstrated fewer correct task responses during robot-led sessions, after adjusting for time and VB-MAPP scores.

2.2.5 Summary

In summary, the robot instructor was associated with **greater positive affect** but **lower task performance**. Communication directed to unknown partners was significantly more frequent in robot-led sessions, suggesting possible issues with clarity or focus. Engagement and most communication behaviors did not significantly differ between conditions. These findings highlight the **potential emotional benefits** of social robot instruction, while also identifying **critical areas—especially performance—that warrant refinement** in robot-mediated pedagogy.

2.3 Interaction Effects: Moderator Influence of Time and VB-MAPP on Robot-Led Instruction

To explore whether the effect of robot-led instruction varied across time or with the learner's developmental profile, we modeled interaction terms between `ConditionRobot` and two continuous covariates: `time_c` (centered session time) and `VBMAPP` (individual skill level). Below we interpret these moderation effects for each behavioral outcome.

2.3.1 Affect

- **Positive Affect**
  - The interaction between `ConditionRobot` and `VBMAPP` was significant:
    = 4.22, SE = 1.48, z = 2.86, p = 0.0043
    This suggests that the advantage of robot instruction in eliciting positive affect increased with higher VB-MAPP scores.
  - No significant moderation by time was observed:
    = -0.070, SE = 0.121, z = -0.57, p = 0.566
- **Negative Affect**
  - Neither time nor VB-MAPP significantly moderated the robot condition's effect:
    Time interaction: = 0.175, p = 0.485
    VB-MAPP interaction: = 0.110, p = 0.974

2.3.2 Communication

- **With Instructor**
  - No significant moderation by time (*= -0.170, p = 0.143*) or VB-MAPP (*= 0.067, p = 0.966*)
- **Instructor-Prompted**
  - VB-MAPP significantly moderated the effect of robot instruction:
    = -10.9, SE = 3.66, z = -2.99, p = 0.0028
    This negative interaction indicates that children with higher VB-MAPP scores showed fewer prompted communications under the robot condition.
  - Time was not a significant moderator: *= -0.337, p = 0.163*
- **With Therapist/Peers**
  - No significant moderation by time or VB-MAPP:
    Time: = 0.208, p = 0.266
    VB-MAPP: = 3.11, p = 0.197
- **Indeterminate**
  - Both moderators were significant:
    - *VB-MAPP interaction: = 7.80, SE = 3.26, z = 2.39, p = 0.0168*
      This suggests the ambiguity in children's social orientation during robot instruction increased with VB-MAPP score.
    - *Time interaction was not significant: = 0.042, p = 0.857*

2.3.3 Engagement

- **On Target**
  - No significant moderation by VB-MAPP: *= -0.534, p = 0.706*
  - Time showed a non-significant trend: *= -0.162, p = 0.132*
- **With Therapist/Peers**
  - Neither moderator had a significant effect:
    Time: = -0.022, p = 0.858
    VB-MAPP: = 0.137, p = 0.933
- **Off Target**
  - Time significantly moderated the robot effect:
    = 0.284, SE = 0.127, z = 2.24, p = 0.0250
    This implies that robot-led instruction was associated with increased off-target engagement as time progressed.
  - No significant interaction with VB-MAPP: *= 0.461, p = 0.779*

2.3.4 Performance

- **Positive Performance**
  - No significant moderation by either variable:
    Time: = 0.019, p = 0.895
    VB-MAPP: = 0.606, p = 0.728

2.3.5 Summary of Interaction Effects

Learning Behavior	VB-MAPP Moderation	Time Moderation	Interpretation
Affect_Positive	p = 0.0043		Robot effect stronger for children with higher VB-MAPP
Affect_Negative			No differential effect
Comm._Instructor			No moderation observed
Comm._Prompted	p = 0.0028		Fewer prompted responses from higher-skilled children under robot condition
Comm._Therapist/Peers			No differential effect
Comm._Indeterminate	p = 0.0168		More ambiguous social behaviors with robot, especially for higher VB-MAPP
Engagement_OnTarget		(trend)	No clear pattern
Engagement_Therapist/Peers			No differential effect
Engagement_OffTarget		p = 0.025	Robot instruction led to more off-task behaviors over time
Performance_Positive			No significant moderation

Overall, robot-led instruction was **more emotionally engaging for higher-performing children**, but it also led to **more ambiguous and off-task behaviors**, especially as sessions progressed. These nuanced findings underscore the importance of **personalizing robotic instruction** based on learner profile and session context.

2.4 Predictive Role of Learning Behaviors on Performance in Human vs. Robot-Led Circle Time

2.4.1 Overview

This section explores how various learning behaviors predict children's **positive performance** during circle-time sessions, separately under **Human** and **Robot** instruction conditions. Using Generalized Linear Mixed Models (GLMMs) with beta regression, we analyzed each behavioral predictor while controlling for children's VB-MAPP developmental levels and session time.

2.4.2 Modeling Approach

For each learning behavior, we modeled:

```
Performance_Positive ~ Behavior + VBMAPP + time_c + (1 | Subject)
```

```
## Performance_Positive ~ Behavior + VBMAPP + time_c + (1 | Subject)
```

This was done independently for the **Robot** and **Human** groups using the `glmmTMB` package in R. The response variable was beta-transformed for appropriate modeling. ### How does each learning behavior predict performance under the robot condition only? ##

```
# Create time_c before subsetting
df$time_c <- scale(df$time, center = TRUE, scale = FALSE)
```

```r
# Subset robot condition
robot_df <- df %>% filter(Condition == "Robot")

#  Beta transformation for Performance_Positive
n <- nrow(robot_df)
robot_df$Performance_Positive <- (robot_df$Performance_Positive * (n - 1) + 0.5) / n

# Define predictors (excluding Performance variables)
predictors <- c(
  "Affect_Positive", "Affect_Negative",
  "Communication_with_Instructor", "Communication_with_Instructor_Prompted",
  "Communication_with_Therapist", "Communication_with_Indeterminent",
  "Engagement_OnTarget", "Engagement_Therapist", "Engagement_OffTarget"
)

# Response variable
response <- "Performance_Positive"

# Loop through predictors to model their effect on performance
for (pred in predictors) {
  cat("\n==============================\n")
  cat("Predictor:", pred, "=>", response, "\n")

  # Define formula
  form <- as.formula(paste(response, "~", pred, "+ VBMAPP + time_c + (1 | Subject)"))

  # Fit model
  model <- glmmTMB(form, data = robot_df, family = beta_family())

  # Tidy output
  results <- broom.mixed::tidy(model, effects = "fixed")
  results$stars <- sapply(results$p.value, sig_stars)

  # Display summary
  print(results[, c("term", "estimate", "std.error", "statistic", "p.value", "stars")])
}
```

```
##
## ==============================
## Predictor: Affect_Positive => Performance_Positive
## # A tibble: 4 x 6
##   term            estimate std.error statistic   p.value stars
##   <chr>              <dbl>     <dbl>     <dbl>     <dbl> <chr>
## 1 (Intercept)     -11.3        3.36     -3.37  0.000748  "***"
## 2 Affect_Positive  -2.92       2.08     -1.40  0.160     ""
## 3 VBMAPP            8.07       2.04      3.96  0.0000754 "***"
## 4 time_c           -0.0300     0.126    -0.238 0.812     ""
##
## ==============================
## Predictor: Affect_Negative => Performance_Positive
## # A tibble: 4 x 6
##   term            estimate std.error statistic p.value stars
##   <chr>              <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)     -12.9        4.72     -2.73  0.00635 "**"
## 2 Affect_Negative  -4.02       7.46     -0.538 0.590   ""
## 3 VBMAPP           8.76       2.94      2.98  0.00287 "**"
## 4 time_c           0.0606     0.122     0.498 0.619   ""
##
## ==============================
## Predictor: Communication_with_Instructor => Performance_Positive
## # A tibble: 4 x 6
##   term                          estimate std.error statistic  p.value stars
##   <chr>                            <dbl>     <dbl>     <dbl>    <dbl> <chr>
## 1 (Intercept)                   -13.4        3.85     -3.48  0.000498 "***"
## 2 Communication_with_Instructor  -4.74       2.11     -2.25  0.0246   "*"
## 3 VBMAPP                          9.35       2.41      3.88  0.000104 "***"
## 4 time_c                         -0.0510     0.117    -0.437 0.662    ""
##
## ==============================
## Predictor: Communication_with_Instructor_Prompted => Performance_Positive
## # A tibble: 4 x 6
##   term                                estimate std.error statistic p.value stars
##   <chr>                                  <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)                         -13.3        4.72     -2.82  0.00487 "**"
## 2 Communication_with_Instructor_Prom~   6.12      23.2       0.264 0.792    ""
## 3 VBMAPP                                8.96       2.90      3.09  0.00201 "**"
## 4 time_c                                0.0597     0.141     0.423 0.673    ""
##
## ==============================
## Predictor: Communication_with_Therapist => Performance_Positive
## # A tibble: 4 x 6
##   term                         estimate std.error statistic p.value stars
##   <chr>                           <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)                  -12.3        4.56     -2.69  0.00713 "**"
## 2 Communication_with_Therapist  -3.46       2.72     -1.27  0.202   ""
## 3 VBMAPP                         8.53       2.83      3.01  0.00260 "**"
## 4 time_c                        -0.0290     0.123    -0.236 0.813   ""
##
## ==============================
## Predictor: Communication_with_Indeterminent => Performance_Positive
## # A tibble: 4 x 6
##   term                             estimate std.error statistic p.value stars
##   <chr>                               <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)                      -13.8        4.64     -2.98  0.00285 "**"
## 2 Communication_with_Indeterminent  -4.40       3.08     -1.43  0.153   ""
## 3 VBMAPP                             9.48       2.91      3.26  0.00111 "**"
## 4 time_c                            -0.0265     0.118    -0.224 0.823   ""
##
## ==============================
## Predictor: Engagement_OnTarget => Performance_Positive
## # A tibble: 4 x 6
##   term                estimate std.error statistic p.value stars
##   <chr>                  <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)         -12.8        5.03    -2.54   0.0110  "*"
## 2 Engagement_OnTarget  -0.0813     2.62    -0.0310 0.975   ""
## 3 VBMAPP                8.71       2.86     3.05   0.00231 "**"
## 4 time_c                0.0381     0.113    0.336  0.737   ""
```

```
##
## ==============================
## Predictor: Engagement_Therapist => Performance_Positive
## # A tibble: 4 x 6
##   term                 estimate std.error statistic p.value stars
##   <chr>                   <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)          -12.7        4.72     -2.69  0.00720 "**"
## 2 Engagement_Therapist  -0.750      5.31     -0.141 0.888   ""
## 3 VBMAPP                 8.64       2.90      2.98  0.00292 "**"
## 4 time_c                 0.0340     0.116     0.292 0.770   ""
##
## ==============================
## Predictor: Engagement_OffTarget => Performance_Positive
## # A tibble: 4 x 6
##   term                 estimate std.error statistic p.value stars
##   <chr>                   <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)          -12.9        4.63     -2.79  0.00535 "**"
## 2 Engagement_OffTarget   0.820      3.68      0.223 0.824   ""
## 3 VBMAPP                 8.66       2.89      2.99  0.00275 "**"
## 4 time_c                 0.0352     0.114     0.310 0.756   ""
```

2.4.3 How does each learning behavior predict performance under the human condition only?

```r
# Subset data to human-only condition
human_df <- df %>% filter(Condition == "Human")

#  Beta-safe transformation of Performance_Positive
n_human <- nrow(human_df)
human_df$Performance_Positive <- (human_df$Performance_Positive * (n_human - 1) + 0.5) / n_human


# Define predictors (excluding Performance variables)
predictors <- c(
  "Affect_Positive", "Affect_Negative",
  "Communication_with_Instructor", "Communication_with_Instructor_Prompted",
  "Communication_with_Therapist", "Communication_with_Indeterminent",
  "Engagement_OnTarget", "Engagement_Therapist", "Engagement_OffTarget"
)

# Response variable
response <- "Performance_Positive"

# Loop through predictors to model their effect on performance
for (pred in predictors) {
  cat("\n==============================\n")
  cat("Predictor:", pred, "=>", response, "\n")

  # Define formula
  form <- as.formula(paste(response, "~", pred, "+ VBMAPP + time_c + (1 | Subject)"))

  # Fit model
  model <- glmmTMB(form, data = human_df, family = beta_family())
  # Tidy output
  results <- broom.mixed::tidy(model, effects = "fixed")
  results$stars <- sapply(results$p.value, sig_stars)

  # Display summary
  print(results[, c("term", "estimate", "std.error", "statistic", "p.value", "stars")])
}
```

```
##
## ==============================
## Predictor: Affect_Positive => Performance_Positive
## # A tibble: 4 x 6
##   term             estimate std.error statistic p.value stars
##   <chr>               <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)     -11.5        4.41     -2.62   0.00882 "**"
## 2 Affect_Positive  -0.390      2.34     -0.166  0.868   ""
## 3 VBMAPP            8.17       2.63      3.11   0.00189 "**"
## 4 time_c            0.00174    0.0964    0.0181 0.986   ""
##
## ==============================
## Predictor: Affect_Negative => Performance_Positive
## # A tibble: 4 x 6
##   term            estimate std.error statistic  p.value stars
##   <chr>              <dbl>     <dbl>     <dbl>    <dbl> <chr>
## 1 (Intercept)     -11.3       1.76      -6.45  1.14e-10 "***"
## 2 Affect_Negative  21.3       8.22       2.60  9.42e- 3 "**"
## 3 VBMAPP            7.90      1.11       7.09  1.35e-12 "***"
## 4 time_c            0.0448    0.0933     0.480 6.31e- 1 ""
##
## ==============================
## Predictor: Communication_with_Instructor => Performance_Positive
## # A tibble: 4 x 6
##   term                           estimate std.error statistic     p.value stars
##   <chr>                             <dbl>     <dbl>     <dbl>       <dbl> <chr>
## 1 (Intercept)                   -11.9        2.38     -5.00   0.000000580 "***"
## 2 Communication_with_Instructor   5.72       2.57      2.23   0.0261      "*"
## 3 VBMAPP                          7.97       1.51      5.28   0.000000126 "***"
## 4 time_c                         -0.00378    0.0899   -0.0421 0.966       ""
##
## ==============================
## Predictor: Communication_with_Instructor_Prompted => Performance_Positive
## # A tibble: 4 x 6
##   term                                estimate std.error statistic p.value stars
##   <chr>                                  <dbl>     <dbl>     <dbl>   <dbl> <chr>
## 1 (Intercept)                         -11.8        3.34     -3.53  4.20e-4 "***"
## 2 Communication_with_Instructor_Prom~   5.96       7.73      0.771 4.40e-1 ""
## 3 VBMAPP                                8.27       2.09      3.96  7.61e-5 "***"
## 4 time_c                                0.0208     0.0876    0.237 8.12e-1 ""
##
## ==============================
## Predictor: Communication_with_Therapist => Performance_Positive
## # A tibble: 4 x 6
##   term                         estimate std.error statistic  p.value stars
##   <chr>                           <dbl>     <dbl>     <dbl>    <dbl> <chr>
## 1 (Intercept)                  -10.3       3.59      -2.87  0.00416  "**"
## 2 Communication_with_Therapist  -2.90      1.66      -1.75  0.0808   "."
## 3 VBMAPP                         7.51      2.21       3.39  0.000699 "***"
## 4 time_c                        -0.0865    0.0997    -0.868 0.386    ""
##
## ==============================
## Predictor: Communication_with_Indeterminent => Performance_Positive
## # A tibble: 4 x 6
##   term                              estimate std.error statistic   p.value stars
##   <chr>                                <dbl>     <dbl>     <dbl>     <dbl> <chr>
## 1 (Intercept)                     -12.0         3.16    -3.80   0.000143  "***"
## 2 Communication_with_Indeterminent  -0.220       2.58    -0.0855 0.932     ""
## 3 VBMAPP                             8.46        1.97     4.29   0.0000180 "***"
## 4 time_c                             0.00269     0.104    0.0258 0.979     ""
##
## ==============================
## Predictor: Engagement_OnTarget => Performance_Positive
## # A tibble: 4 x 6
##   term                estimate std.error statistic   p.value stars
##   <chr>                  <dbl>     <dbl>     <dbl>     <dbl> <chr>
## 1 (Intercept)         -13.3       3.17      -4.20  0.0000261 "***"
## 2 Engagement_OnTarget   1.77      0.970      1.82  0.0680    "."
## 3 VBMAPP                8.41      1.93       4.35  0.0000137 "***"
## 4 time_c               -0.0156    0.0842    -0.186 0.853     ""
##
## ==============================
## Predictor: Engagement_Therapist => Performance_Positive
## # A tibble: 4 x 6
##   term                  estimate std.error statistic   p.value stars
##   <chr>                    <dbl>     <dbl>     <dbl>     <dbl> <chr>
## 1 (Intercept)          -12.0        3.22     -3.72   0.000196  "***"
## 2 Engagement_Therapist  -0.553      4.01     -0.138  0.890     ""
## 3 VBMAPP                 8.44       1.99      4.24   0.0000221 "***"
## 4 time_c                 0.00517    0.0913    0.0566 0.955     ""
##
## ==============================
## Predictor: Engagement_OffTarget => Performance_Positive
## # A tibble: 4 x 6
##   term                 estimate std.error statistic    p.value stars
##   <chr>                   <dbl>     <dbl>     <dbl>      <dbl> <chr>
## 1 (Intercept)          -11.8       3.07      -3.82  0.000131   "***"
## 2 Engagement_OffTarget  -2.19      1.07      -2.05  0.0402     "*"
## 3 VBMAPP                 8.51      1.92       4.44  0.00000891 "***"
## 4 time_c                -0.0121    0.0825    -0.147 0.883      ""
```

2.4.4 Summary of Results

2.4.4.1 Table 1. Predictive Effects of Learning Behaviors on Performance (Robot Condition)

Predictor	Estimate	Std. Error	z-value	p-value	Significance
Affect__Positive	-3.09	2.13	-1.45	0.148	
Affect__Negative	-4.11	7.72	-0.53	0.594	
Communication__with__Instructor	-4.91	2.19	-2.25	0.0247	*
Communication__with__Instructor__Prompted	6.32	23.9	0.26	0.792	
Communication__with__Therapist	-3.60	2.81	-1.28	0.200	
Communication__with__Indeterminent	-4.52	3.19	-1.42	0.156	
Engagement__OnTarget	-0.07	2.71	-0.03	0.979	
Engagement__Therapist	-0.83	5.49	-0.15	0.880	
Engagement__OffTarget	0.85	3.80	0.22	0.824	

2.4.4.2 Interpretation – Robot Condition

- **Only one behavior**, `Communication_with_Instructor`, significantly predicted performance ($p = 0.025$). Interestingly, the relationship was **negative**, suggesting that more communication with the robot instructor might be linked to **lower** performance. This could be due to the child's reliance on prompts or social behavior not aligning with task success.
- All other behaviors were not significantly predictive of performance in the robot-led sessions, indicating either weaker or non-existent associations in this context.

2.4.4.3 Table 2. Predictive Effects of Learning Behaviors on Performance (Human Condition)

Predictor	Estimate	Std. Error	z-value	p-value	Significance
Affect__Positive	-0.43	2.42	-0.18	0.860	
Affect__Negative	22.0	8.45	2.60	0.009	**
Communication__with__Instructor	5.89	2.64	2.23	0.026	*
Communication__with__Instructor__Prompted	6.12	7.97	0.77	0.443	
Communication__with__Therapist	-3.00	1.71	-1.75	0.080	.
Communication__with__Indeterminent	-0.22	2.65	-0.08	0.934	
Engagement__OnTarget	1.83	1.00	1.82	0.068	.
Engagement__Therapist	-0.57	4.13	-0.14	0.891	
Engagement__OffTarget	-2.25	1.10	-2.05	0.040	*

2.4.4.4 Interpretation – Human Condition

- **Affect__Negative** had a **strong positive association** with performance ($p < 0.01$). This counterintuitive finding may suggest that children expressing frustration or difficulty were **engaging with the task**, potentially improving with guidance.
- **Communication__with__Instructor** was also positively predictive ($p = 0.026$), contrasting with its negative effect under the robot condition. This may highlight the natural fluidity and effectiveness of human-led interactions.
- **Engagement__OffTarget** significantly predicted **lower** performance ($p = 0.040$), as expected.
- Marginal trends were observed for `Engagement_OnTarget` ($p = 0.068$) and `Communication_with_Therapist` ($p = 0.080$), suggesting potential relevance with larger samples.

2.4.5 Comparative Insights

- **Communication_with_Instructor** showed opposite effects:
  - **Negative** under robot instruction ($p = 0.025$)
  - **Positive** under human instruction ($p = 0.026$)
    This suggests that while human instructors likely foster meaningful instructional interactions, interactions with the robot may signal confusion, prompting, or a lack of task engagement.

- **Affect_Negative** was **positively predictive** of performance only under the **human** condition. This may reflect effective human responses to emotional cues, adjusting support to improve performance.

- None of the **Engagement metrics** significantly predicted performance in robot-led sessions. However, `Engagement_OffTarget` was **negatively associated** with performance under human instruction ($p = 0.040$), aligning with expectations.

- Across all models, **VB-MAPP score** was a **strong and consistent positive predictor** of performance ($p < 0.01$), validating the role of developmental ability.