Binomial Model: Graphs and Stats

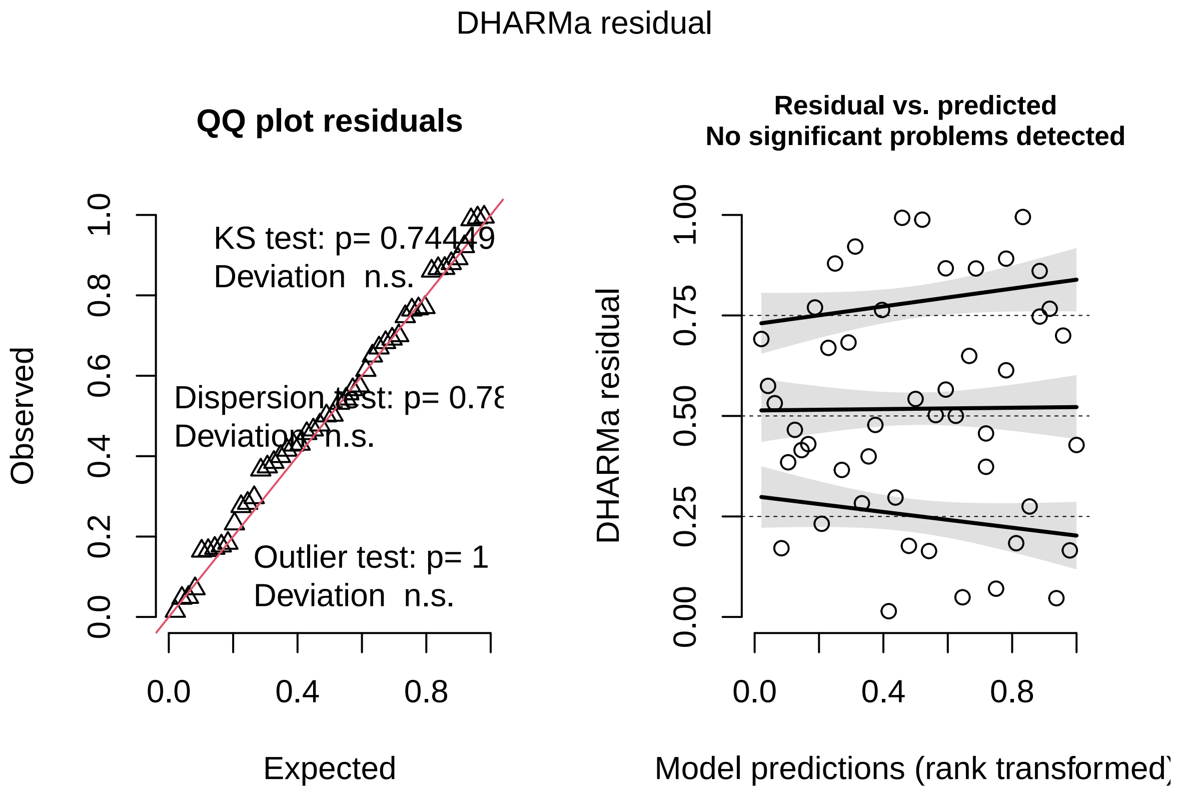
Bri Over

2024-12-11

```
# Load the dataset and alter to be used in making the model
nudi_data <- read.csv("clean_data/Meyknecht_Over_Parker_MRNE475_2024_combinedraw.csv")%>%
  mutate(temp_grp = as.factor(temp_grp)) #change how the data is structured so that treatment group is
nudi_data$trial_type[nudi_data$trial_type=="conspecific "] <- "conspecific" #ensure all cue types are l
# View the structure of the updated dataset to confirm changes:
str(nudi_data)
```

```
## ’data.frame’:    48 obs. of  9 variables:
##  $ nudi_num       : int  2 2 3 3 4 4 5 5 6 6 ...
##  $ temp_grp       : Factor w/ 4 levels "12","14","16",..: 3 3 4 4 1 1 2 2 1 1 ...
##  $ trial_type     : chr  "conspecific" "predator" "conspecific" "predator" ...
##  $ mean_angle     : num  156 160 195 188 311 ...
##  $ sd_angle       : num  102 112 103 105 100 ...
##  $ mean_vel       : num  0.001379 0.001311 0.001302 0.000949 0.000613 ...
##  $ sd_vel         : num  0.00059 0.000769 0.00089 0.000788 0.000358 ...
##  $ avg_length_m   : num  0.033 0.033 0.037 0.037 0.0095 0.0095 0.04 0.04 0.034 0.034 ...
##  $ direction_binom: int  0 0 1 1 1 0 0 1 0 0 ...
```

```
#create the binomial model, looking at treatment group, body length and cue type as factors
#along with the interaction between treatment group and body length
glm_binom2 <- glmmTMB(direction_binom ~ temp_grp * avg_length_m + trial_type, family = binomial(link =
simulateResiduals(glm_binom2)%>%
  plot() #plot residuals to visually determine fit of model, if no red text or lines data fits the bino
```

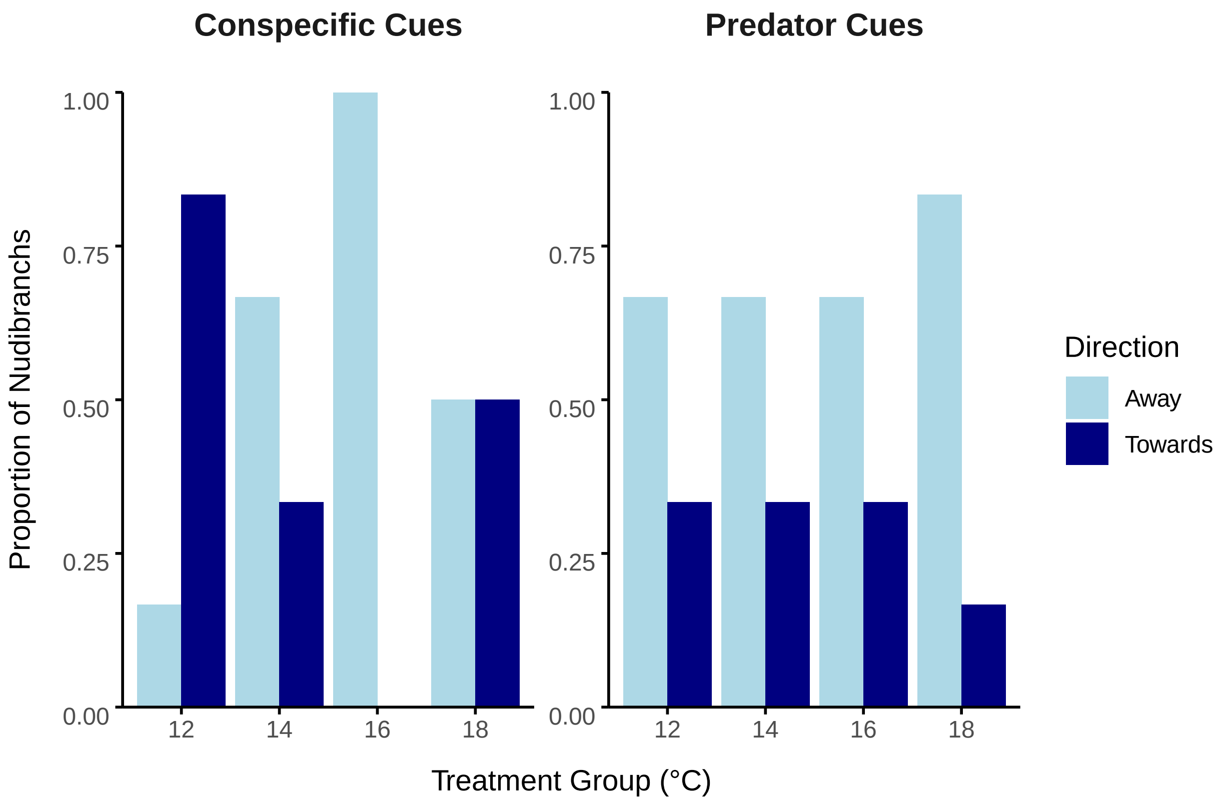


```
#Residuals show that the binomial model fits the data

# Extract the coefficients directly from the model summary
binom_table <- as.data.frame(summary(glm_binom2)$coefficients$cond)

summary(glm_binom2) #check data before exporting
```

```
##  Family: binomial  ( logit )
## Formula:          direction_binom ~ temp_grp * avg_length_m + trial_type
## Data: nudi_data
##
##      AIC      BIC   logLik deviance df.resid
##     64.2     81.0    -23.1     46.2       39
##
##
## Conditional model:
##                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)              1.6804     1.6808   1.000   0.3174
## temp_grp14             -17.5703    10.5224  -1.670   0.0950 .
## temp_grp16             -13.5196    11.3046  -1.196   0.2317
## temp_grp18              -7.1760     3.5701  -2.010   0.0444 *
## avg_length_m           -40.9577    64.3139  -0.637   0.5242
## trial_typepredator      -0.7745     0.7328  -1.057   0.2906
## temp_grp14:avg_length_m 453.1847   276.1705   1.641   0.1008
## temp_grp16:avg_length_m 349.7955   321.5785   1.088   0.2767
## temp_grp18:avg_length_m 205.3878   116.0277   1.770   0.0767 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# Save the table as a CSV file
write.csv(binom_table, "bri_code/Meyknecht_Over_Parker_MRNE475_2024_binom_table.csv", row.names = TRUE)
```

```r
# Summarize the data: Count nudibranchs that moved towards or away for each combination of trial_type,
nudi_data_summary <- nudi_data %>%
  group_by(trial_type, temp_grp, direction_binom) %>%
  summarise(count = n(), .groups = 'drop')

# Calculate total number of nudibranchs in each trial_type and temp_grp
nudi_data_totals <- nudi_data %>%
  group_by(trial_type, temp_grp) %>%
  summarise(total = n(), .groups = 'drop')

# Join the summarized data with the total counts to calculate proportions
nudi_data_prop <- nudi_data_summary %>%
  left_join(nudi_data_totals, by = c("trial_type", "temp_grp")) %>%
  mutate(proportion = count / total)  # Calculate proportion for each direction

# Ensure 'direction_binom' is a factor with appropriate levels
nudi_data_prop$direction_binom <- factor(nudi_data_prop$direction_binom, levels = c(0, 1))

# Handle missing data for "towards" direction in case there are no nudibranchs for that direction
nudi_data_prop_complete <- nudi_data_prop %>%
  complete(trial_type, temp_grp, direction_binom, fill = list(count = 0))  # Ensures missing rows are a

# Set any NA values in proportion to 0 (this handles cases where count is 0 or missing)
nudi_data_prop_complete$proportion[is.na(nudi_data_prop_complete$proportion)] <- 0

# Get the maximum proportion to set the y-axis limit
y_max <- max(nudi_data_prop_complete$proportion, na.rm = TRUE)

# Create the bar chart with the correct facet to have both conspecific and predator cues, with the same
plot_rxn_probs <- ggplot(nudi_data_prop_complete, aes(x = factor(temp_grp), y = proportion, fill = facto
  geom_bar(stat = "identity", position = "dodge") +  # dodge bars so they do not overlap
  facet_wrap(~ trial_type, scales = "free_y",
             labeller = labeller(trial_type = c("conspecific" = "Conspecific Cues", "predator" = "Preda
  labs(
    title = "", #Set axes and legend title and remove graph title
    x = "Treatment Group (°C)",
    y = "Proportion of Nudibranchs",
    fill = "Direction"
  ) +
  scale_fill_manual(values = c("0" = "lightblue", "1" = "navyblue"), labels = c("Away", "Towards")) +
  scale_y_continuous(limits = c(0, y_max), expand = c(0, 0)) +  #Proportion bars touch the x-axis
  scale_x_discrete(expand = c(0.2, 0)) +  # Increase space between the treatment groups on the x-axis
  theme_minimal() +
  theme(
    axis.text.y = element_text(vjust = 1),  # Adjust vertical alignment of y-axis text
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    axis.line = element_line(color = "black", linewidth = 0.5),
    axis.ticks = element_line(color = "black", linewidth = 0.5),
    axis.title.x = element_text(margin = margin(t = 10)),  # Add space between x-axis labels and number
    axis.title.y = element_text(margin = margin(r = 10)),  # Add space between y-axis labels and number
    strip.text = element_text(size = 12, face = "bold", margin = margin(b = 20))  # Add space between f
  )

plot_rxn_probs
```



```
#optional look at summary table for proportion values of bar chart, may save and include
#double check that each treatment group is represented properly

summary_table <- nudi_data_prop_complete %>%
  dplyr::select(trial_type, temp_grp, direction_binom, proportion) %>%
  arrange(trial_type, temp_grp, direction_binom)

print(summary_table)
```

```
## # A tibble: 16 x 4
##    trial_type  temp_grp direction_binom proportion
##    <chr>       <fct>    <fct>                <dbl>
##  1 conspecific 12       0                    0.167
##  2 conspecific 12       1                    0.833
##  3 conspecific 14       0                    0.667
##  4 conspecific 14       1                    0.333
##  5 conspecific 16       0                    1
##  6 conspecific 16       1                    0
##  7 conspecific 18       0                    0.5
##  8 conspecific 18       1                    0.5
##  9 predator    12       0                    0.667
## 10 predator    12       1                    0.333
## 11 predator    14       0                    0.667
## 12 predator    14       1                    0.333
## 13 predator    16       0                    0.667
## 14 predator    16       1                    0.333
## 15 predator    18       0                    0.833
## 16 predator    18       1                    0.167
```

```
#save plot to git folder

ggsave(plot = plot_rxn_probs, filename = here("bri_code", "Meyknecht_Over_Parker_MRNE475_2024_plot_rxn_p
```