

Study 1 Multinomial Logistic Regression Analysis

Load the data

```
data<- read.csv("/Users/daryani/Desktop/cleaned_data.csv")  
  
# Load required libraries  
library(nnet)      # For multinomial logistic regression  
library(effects)   # For plotting effects
```

Loading required package: carData

```
lattice theme set by effectsTheme()  
See ?effectsTheme for details.
```

```
library(ggplot2)    # For visualization  
library(car)        # For Anova function  
library(emmeans)    # For estimated marginal means
```

Welcome to emmeans.

Caution: You lose important information if you filter this package's results.
See '? untidy'

```
library(dplyr)
```

Attaching package: 'dplyr'

```
The following object is masked from 'package:car':
```

```
recode
```

```
The following objects are masked from 'package:stats':
```

```
filter, lag
```

```
The following objects are masked from 'package:base':
```

```
intersect, setdiff, setequal, union
```

```
library(tidyr)
```

Data Preparation

```
# Create moral foundation type variable (individualizing vs. binding)
data$moral.foundation_type <- ifelse(data$care == 1 | data$fairness == 1,
                                      "Individualizing",
                                      ifelse(data$purity == 1 | data$loyalty == 1 | data$authoritarian == 1,
                                             "Binding", NA))

# Ensure Label is a factor with "Only Prolife" as the reference level
data$stance <- factor(data$Label,
                       levels = c("Only Prolife",
                                 "Prolife talking about Prochoice",
                                 "Prochoice talking about Prolife",
                                 "Only Prochoice"))

# Convert moral.foundation_type to factor
data$moral.foundation_type <- factor(data$moral.foundation_type)
```

Multinomial Logistic Regression Model with main effects only

```
# Define the model
model1 <- multinom(stance ~ moral.foundation_type + sentiment_score, data = data)
```

```

# weights: 16 (9 variable)
initial value 235958.390617
iter 10 value 225282.813666
final value 225082.706656
converged

# Display model summaries
summary(model1)

Call:
multinom(formula = stance ~ moral.foundation_type + sentiment_score,
  data = data)

Coefficients:
                               (Intercept)
Prolife talking about Prochoice -0.8464497
Prochoice talking about Prolife -0.5029017
Only Prochoice                 -1.5050929
                               moral.foundation_typeIndividualizing
Prolife talking about Prochoice          0.6154509
Prochoice talking about Prolife          0.8850781
Only Prochoice                         1.3418277
                               sentiment_score
Prolife talking about Prochoice      -0.54652321
Prochoice talking about Prolife       -0.73622146
Only Prochoice                        0.07596918

Std. Errors:
                               (Intercept)
Prolife talking about Prochoice  0.03800024
Prochoice talking about Prolife  0.03413618
Only Prochoice                  0.04692993
                               moral.foundation_typeIndividualizing
Prolife talking about Prochoice          0.03880818
Prochoice talking about Prolife          0.03480210
Only Prochoice                         0.04749825
                               sentiment_score
Prolife talking about Prochoice      0.01254937
Prochoice talking about Prolife       0.01091203
Only Prochoice                        0.01232374

Residual Deviance: 450165.4

```

AIC: 450183.4

```
# Calculate z-values and p-values (multinom doesn't provide these automatically)
z_values <- summary(model1)$coefficients / summary(model1)$standard.errors
p_values <- (1 - pnorm(abs(z_values))) * 2
print(p_values)
```

	(Intercept)
Prolife talking about Prochoice	0
Prochoice talking about Prolife	0
Only Prochoice	0
	moral.foundation_typeIndividualizing
Prolife talking about Prochoice	0
Prochoice talking about Prolife	0
Only Prochoice	0
	sentiment_score
Prolife talking about Prochoice	0.000000e+00
Prochoice talking about Prolife	0.000000e+00
Only Prochoice	7.072536e-10

Model with Interaction

```
# Interaction model
model2 <- multinom(stance ~ moral.foundation_type * sentiment_score, data = data)
```

```
# weights: 20 (12 variable)
initial value 235958.390617
iter 10 value 226301.168880
final value 225007.723305
converged
```

```
# Display model summaries
summary(model2)
```

Call:
multinom(formula = stance ~ moral.foundation_type * sentiment_score,
data = data)

Coefficients:

	(Intercept)
Prolife talking about Prochoice	-0.7441994
Prochoice talking about Prolife	-0.4163190
Only Prochoice	-1.3339700
	moral_foundation_typeIndividualizing
Prolife talking about Prochoice	0.513834
Prochoice talking about Prolife	0.799432
Only Prochoice	1.170072
	sentiment_score
Prolife talking about Prochoice	-1.1020948
Prochoice talking about Prolife	-1.3405173
Only Prochoice	-0.6642977
	moral_foundation_typeIndividualizing:sentiment_score
Prolife talking about Prochoice	0.5781803
Prochoice talking about Prolife	0.6266439
Only Prochoice	0.7625772

Std. Errors:

	(Intercept)
Prolife talking about Prochoice	0.03990867
Prochoice talking about Prolife	0.03626715
Only Prochoice	0.04960321
	moral_foundation_typeIndividualizing
Prolife talking about Prochoice	0.04065141
Prochoice talking about Prolife	0.03688181
Only Prochoice	0.05016979
	sentiment_score
Prolife talking about Prochoice	0.06998381
Prochoice talking about Prolife	0.06410015
Only Prochoice	0.08582039
	moral_foundation_typeIndividualizing:sentiment_score
Prolife talking about Prochoice	0.07113931
Prochoice talking about Prolife	0.06505269
Only Prochoice	0.08672270

Residual Deviance: 450015.4

AIC: 450039.4

```
# Calculate z-values and p-values
z_values <- summary(model2)$coefficients / summary(model2)$standard.errors
p_values <- (1 - pnorm(abs(z_values))) * 2
print(p_values)
```

	(Intercept)
Prolife talking about Prochoice	0
Prochoice talking about Prolife	0
Only Prochoice	0
	moral_foundation_typeIndividualizing
Prolife talking about Prochoice	0
Prochoice talking about Prolife	0
Only Prochoice	0
	sentiment_score
Prolife talking about Prochoice	0.000000e+00
Prochoice talking about Prolife	0.000000e+00
Only Prochoice	9.992007e-15
	moral_foundation_typeIndividualizing:sentiment_score
Prolife talking about Prochoice	4.440892e-16
Prochoice talking about Prolife	0.000000e+00
Only Prochoice	0.000000e+00

Model Comparison

```
# Compare models with and without interaction
anova(model1, model2)
```

Likelihood ratio tests of Multinomial Models

Response: stance	Model	Resid.	df	Resid.	Dev	Test	Df
1 moral.foundation_type + sentiment_score	510615	450165.4					
2 moral.foundation_type * sentiment_score	510612	450015.4	1 vs 2				3
LR stat.	Pr(Chi)						
1							
2 149.9667	0						

```
# Compare models
AIC(model1, model2)
```

	df	AIC
model1	9	450183.4
model2	12	450039.4

```
BIC(model1, model2)
```

```
      df      BIC
model1 9 450273.8
model2 12 450160.0
```

Visualization-Interaction Effect Between Moral Foundations and Sentiment on Stance

```
# Create a prediction grid
moral_types <- c("Binding", "Individualizing")
sentiment_points <- c(-0.75, -0.25, 0.25, 0.75) # Representative sentiment values
newdata <- expand.grid(
  moral.foundation_type = moral_types,
  sentiment_score = sentiment_points
)

# Generate predictions from the model
pred_probs <- predict(model2, newdata = newdata, type = "probs")
newdata <- cbind(newdata, as.data.frame(pred_probs))

# Reshape and prepare for plotting
plot_data <- newdata %>%
  pivot_longer(
    cols = c("Only Prolife", "Prolife talking about Prochoice",
            "Prochoice talking about Prolife", "Only Prochoice"),
    names_to = "stance",
    values_to = "probability"
  ) %>%
  mutate(sentiment_group = factor(
    case_when(
      sentiment_score == -0.75 ~ "Strong Negative (-0.75)",
      sentiment_score == -0.25 ~ "Moderate Negative (-0.25)",
      sentiment_score == 0.25 ~ "Moderate Positive (0.25)",
      sentiment_score == 0.75 ~ "Strong Positive (0.75)"
    ),
    levels = c("Strong Negative (-0.75)", "Moderate Negative (-0.25)",
              "Moderate Positive (0.25)", "Strong Positive (0.75)")
  ))
)
```

```

# Define the custom colors for each stance
custom_colors <- c(
  "Only Prolife" = "#B31529",
  "Prolife talking about Prochoice" = "#F6A482",
  "Prochoice talking about Prolife" = "#8EC4DE",
  "Only Prochoice" = "#1065AB"
)

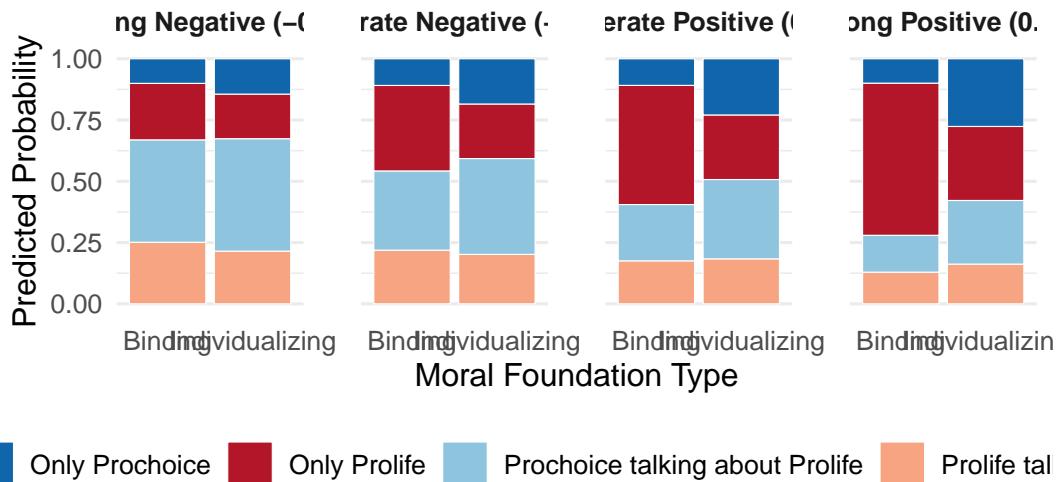
# Create the improved visualization with manual colors
ggplot(plot_data, aes(x = factor(moral.foundation_type, levels = moral_types), y = probability))
  geom_bar(stat = "identity", position = "stack", color = "white", size = 0.2) +
  facet_wrap(~ sentiment_group, nrow = 1) +
  labs(
    title = "Sentiment Moderates the Effect of Moral Foundations on Abortion Stance",
    subtitle = "Predicted probability of each stance by moral foundation type across sentiment groups",
    x = "Moral Foundation Type",
    y = "Predicted Probability",
    fill = "Stance",
    caption = "Note: Sentiment scores range from -1 (extremely negative) to 1 (extremely positive)"
  ) +
  scale_fill_manual(values = custom_colors) +
  theme_minimal(base_size = 12) +
  theme(
    legend.position = "bottom",
    strip.text = element_text(face = "bold", size = 10),
    plot.title = element_text(size = 14, face = "bold"),
    plot.subtitle = element_text(size = 11),
    panel.grid.major.x = element_blank(),
    panel.spacing = unit(1.5, "lines"),
    axis.text.x = element_text(angle = 0, size = 10),
    plot.caption = element_text(hjust = 0, size = 9, face = "italic")
)

```

Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
 i Please use `linewidth` instead.

Sentiment Moderates the Effect of Moral Foundations

Predicted probability of each stance by moral foundation type across sentiment scores from -1 to 1.



Legend: Only Prochoice (dark blue), Only Prolife (dark red), Prochoice talking about Prolife (light blue), Prolife talking about Prochoice (orange).

Note: Sentiment scores range from -1 (extremely negative) to 1 (extremely positive).

```
# Save the plot
ggsave("/Users/daryani/Desktop/stance_by_moral_sentiment.jpg",
       width = 10, height = 5, dpi = 300)
```

Visualization-Main effect of Morality

```
# Data based on model predictions
main_effect_data <- data.frame(
  stance = rep(c("Only Prolife", "Prolife talking about Prochoice",
                "Prochoice talking about Prolife", "Only Prochoice"), each = 2),
  moral.foundation.type = rep(c("Binding", "Individualizing"), 4),
  probability = c(0.4433, 0.2434, # Only Prolife (Binding, Individualizing)
                 0.1902, 0.1932, # Prolife talking about Prochoice
                 0.2681, 0.3567, # Prochoice talking about Prolife
                 0.0984, 0.2067), # Only Prochoice
  # For confidence intervals, we'd need to calculate them using the standard errors
  lower_ci = c(0.40, 0.22,
              0.17, 0.17,
              0.24, 0.32,
              0.09, 0.19),
  upper_ci = c(0.49, 0.27,
              0.21, 0.21,
```

```

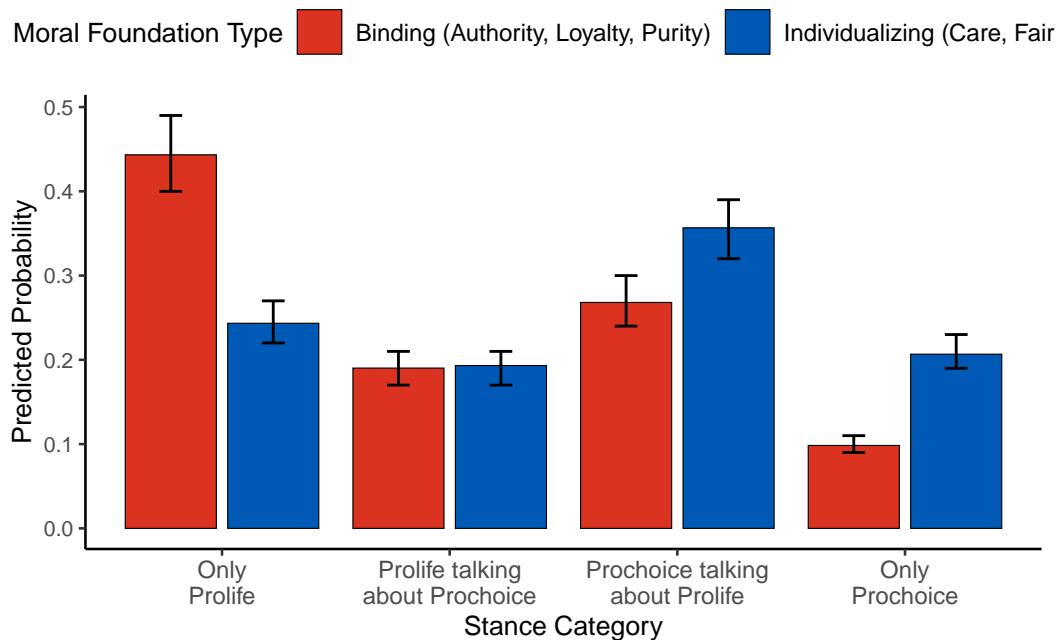
        0.30, 0.39,
        0.11, 0.23)
)

# Use multi-line stance labels to avoid rotation
main_effect_data$stance <- factor(main_effect_data$stance,
                                    levels = c("Only Prolife", "Prolife talking about Prochoice",
                                              "Prochoice talking about Prolife", "Only Prochoice"),
                                    labels = c("Only\nProlife", "Prolife talking\nabout Prochoice",
                                              "Prochoice talking\nabout Prolife", "Only\nProchoice"))

# Build plot
p <- ggplot(main_effect_data, aes(x = stance, y = probability, fill = moral.foundation_type))
  geom_bar(stat = "identity", position = position_dodge(width = 0.9),
            width = 0.8, color = "black", linewidth = 0.2) +
  geom_errorbar(aes(ymin = lower_ci, ymax = upper_ci),
                width = 0.2, position = position_dodge(width = 0.9)) +
  scale_fill_manual(
    values = c("Binding" = "#DC3220", "Individualizing" = "#005AB5"),
    labels = c("Binding (Authority, Loyalty, Purity)",
              "Individualizing (Care, Fairness)"))
) +
  labs(
    x = "Stance Category",
    y = "Predicted Probability",
    fill = "Moral Foundation Type"
) +
  theme_classic(base_size = 10) +
  theme(
    legend.position = "top",
    axis.text.x = element_text(size = 9),
    axis.title = element_text(size = 10),
    legend.title = element_text(size = 10),
    legend.text = element_text(size = 9),
    panel.grid.major.x = element_blank()
)
)

# Display plot
print(p)

```



```
ggsave("/Users/daryani/Desktop/morality_main_effect.jpg",
       width = 10, height = 5, dpi = 300)
```