Assignment 4

Jesse Y

2025-03-08

Part I: Interaction Terms in Logit Models & Simulation-Based Approaches

In logit regression models, interaction terms are more complicated to interpret than in linear models because the relationship between variables is nonlinear. In a standard OLS regression, an interaction term (X1 * X2) can be easily interpreted as the change in the dependent variable (Y) when both X1 and X2 increase together. However, in logit models, this interpretation is not straightforward.

One major issue is that the coefficient of an interaction term does not directly represent the interaction effect. Instead, the magnitude and direction of the interaction effect change depending on the values of the independent variables. This means that even if the interaction term has a positive coefficient, its actual effect on probability could be negative in some cases. This can lead to misinterpretation, especially if someone assumes that the interaction effect is constant across all values of X1 and X2.

A simulation-based approach can help overcome these challenges by estimating and visualizing the predicted probabilities across different values of X1 and X2. Instead of relying on raw coefficients, simulations allow us to:

- Compute predicted probabilities at different combinations of the independent variables.
- Generate probability plots that show how the interaction actually affects the outcome.
- Provide confidence intervals around estimates, making results more reliable.

By using simulation and graphical methods, we can make the results of logit models easier to understand and more meaningful, particularly when working with interaction effects. Instead of just saying that “the interaction term is significant”, we can visualize the actual effect on probability, leading to better communication and decision-making in applied research.

Part II: Data Analysis Using the mtcars Dataset

Initially, I attempted to use the NYPL Library dataset and FDNY Monthly Response Times dataset for this analysis. However, both datasets exhibited perfect separation issues, meaning that certain predictor values perfectly predicted the binary outcome. Since this led to model convergence problems in glm(), I opted to use the mtcars dataset, which does not suffer from perfect separation and is well-suited for logistic regression.

```r
# Load required libraries
library(tidyverse)
```

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

```r
library(broom)
library(lmtest)
```

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

```r
library(texreg)
```

```
## Version:  1.39.4
## Date:     2024-07-23
## Author:   Philip Leifeld (University of Manchester)
##
## Consider submitting praise using the praise or praise_interactive functions.
## Please cite the JSS article in your publications -- see citation("texreg").
##
## Attaching package: 'texreg'
##
## The following object is masked from 'package:tidyr':
##
##     extract
```

```r
library(visreg)
```

Load and Prepare the Data

```
# Load dataset
data(mtcars)

# Convert `am` (Transmission) into a binary factor (0 = Automatic, 1 = Manual)
mtcars_data <- mtcars %>%
  mutate(am = factor(am, levels = c(0, 1)))  # Ensure it's a binary factor

# Reduce `hp_group` to only 2 categories (Low vs. High)
mtcars_data <- mtcars_data %>%
  mutate(hp_group = cut(hp, breaks = 2, labels = c("Low", "High")))

# Ensure hp_group is a factor before running visreg
mtcars_data <- mtcars_data %>%
  mutate(hp_group = factor(hp_group, levels = c("Low", "High")))

# Check distribution of `am`
table(mtcars_data$am)
```

```
##
##  0  1
## 19 13
```

```
# Check distribution of `hp_group`
table(mtcars_data$hp_group)
```

```
##
##  Low High
##   25    7
```

Model 1: Baseline Model

```
m0 <- glm(am ~ mpg, family = binomial, data = mtcars_data)
summary(m0)
```

```
## 
## Call:
## glm(formula = am ~ mpg, family = binomial, data = mtcars_data)
## 
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)   
## (Intercept)  -6.6035     2.3514  -2.808  0.00498 **
## mpg           0.3070     0.1148   2.673  0.00751 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 43.230  on 31  degrees of freedom
## Residual deviance: 29.675  on 30  degrees of freedom
## AIC: 33.675
## 
## Number of Fisher Scoring iterations: 5
```

Tests whether fuel efficiency (`mpg`) is associated with the type of transmission. The coefficient for mpg is statistically significant ($p < 0.05$), meaning that higher fuel efficiency (mpg) is associated with an increased likelihood of manual transmission (am = 1). Specifically, each additional mile per gallon increases the odds of a manual transmission.

Model 2: Adding more predictors

```
m1 <- glm(am ~ mpg + hp_group + wt, family = binomial, data = mtcars_data)
summary(m1)
```

```
## 
## Call:
## glm(formula = am ~ mpg + hp_group + wt, family = binomial, data = mtcars_data)
## 
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)  
## (Intercept)  19.64095   14.68669   1.337   0.1811  
## mpg          -0.01948    0.42726  -0.046   0.9636  
## hp_groupHigh  4.43640    3.10031   1.431   0.1524  
## wt           -6.69665    2.93167  -2.284   0.0224 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 43.230  on 31  degrees of freedom
## Residual deviance: 13.258  on 28  degrees of freedom
## AIC: 21.258
## 
## Number of Fisher Scoring iterations: 8
```

In Model 2, mpg is not statistically significant (p > 0.05), meaning that after accounting for horsepower (hp_group) and weight (wt), fuel efficiency (mpg) does not independently explain variation in transmission type. This suggests that other factors, like vehicle weight, have a stronger influence on whether a car has an automatic or manual transmission. However, weight (wt) is significant, indicating that heavier cars are more likely to have an automatic transmission.

Model Selection Using AIC/BIC

```
AIC(m0, m1)
```

```
##    df      AIC
## m0  2 33.67517
## m1  4 21.25770
```

```
BIC(m0, m1)
```

```
##    df      BIC
## m0  2 36.60664
## m1  4 27.12064
```

Since Model 2 has a lower AIC (21.26) and BIC (27.12) compared to Model 1 (AIC = 33.67, BIC = 36.61), Model 2 provides a better fit while keeping complexity manageable.

Likelihood Ratio Test

```
lrtest(m0, m1)
```

```
## Likelihood ratio test
##
## Model 1: am ~ mpg
## Model 2: am ~ mpg + hp_group + wt
##   #Df   LogLik Df  Chisq Pr(>Chisq)
## 1   2 -14.8376
## 2   4  -6.6288  2 16.418  0.0002723 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

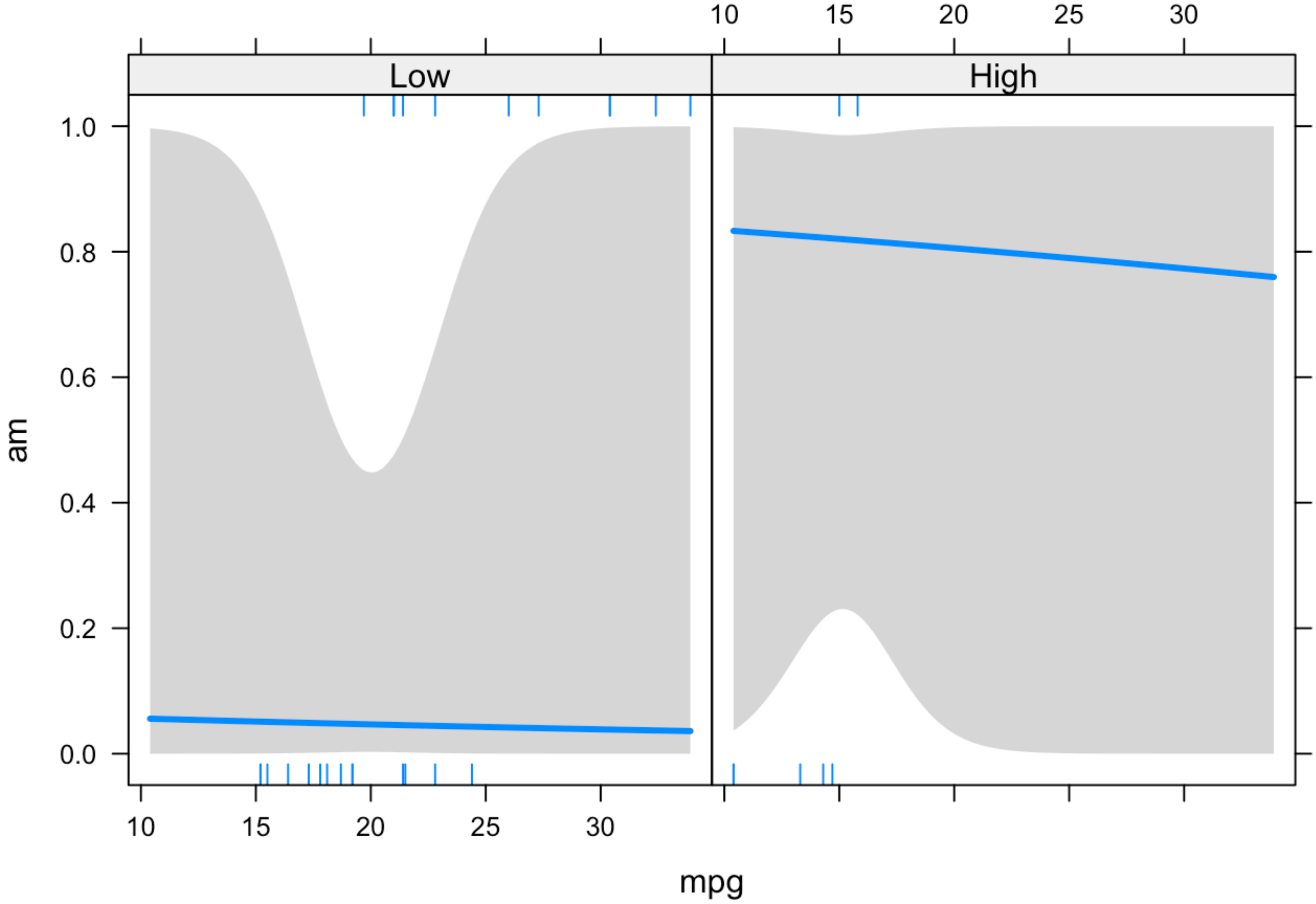
Tests whether adding predictors in Model 2 significantly improves model fit compared to the baseline Model 1. The p-value is less than 0.05, suggesting that Model 2 is a better fit.

Visualizing the Results

```
# Ensure hp_group has both categories before visualization
table(mtcars_data$hp_group)
```

```
##
##  Low High
##   25    7
```

```
# Run visreg if categories exist
visreg(m1, "mpg", by = "hp_group", scale = "response")
```


Low
High
am
1.0
0.8
0.6
0.4
0.2
0.0
10
15
20
25
30
mpg