W271 Summer 2022 Lecture Video Question Solutions Week 3

Contents

Week 3 Discrete Response Model Part 3

3.3 Variable Transformation Part 1: Interactions Among Explanatory Variables—An Example

Q: If we use a strict $\alpha = 0.05$ type I error rate, what would be the conclusions under the two types of tests?

```r
# data can be downloaded from https://www.chrisbilder.com/categorical/programs_and_data.html
df <- read.csv("~/Documents/Berkeley W271/Week 3 Discrete Response Model Part 3/Placekick.csv",
               stringsAsFactors = F)

glm.model.HA <- glm(formula = good ~ distance + wind + distance : wind, family = binomial(link = logit)
                    data = df)

glm.model.H0 <- glm(formula = good ~ distance + wind, family = binomial(link = logit),
                    data = df)

summary(glm.model.HA)
```

Solution:

```
##
## Call:
## glm(formula = good ~ distance + wind + distance:wind, family = binomial(link = logit),
##     data = df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7291   0.2465   0.2465   0.3791   1.8647
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)    5.684181   0.335962  16.919   <2e-16 ***
## distance      -0.110253   0.008603 -12.816   <2e-16 ***
## wind           2.469975   1.662144   1.486   0.1373
## distance:wind -0.083735   0.043301  -1.934   0.0531 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1013.43  on 1424  degrees of freedom
## Residual deviance:  767.42  on 1421  degrees of freedom
## AIC: 775.42
##
## Number of Fisher Scoring iterations: 6
```

```
anova(glm.model.H0, glm.model.HA, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: good ~ distance + wind
## Model 2: good ~ distance + wind + distance:wind
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1      1422     772.53
## 2      1421     767.42  1   5.1097  0.02379 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#install.packages("car")

library(car)
```

```
## Warning: package 'car' was built under R version 4.1.1
```

```
## Loading required package: carData
```

```
## Warning: package 'carData' was built under R version 4.1.1
```

```
Anova(glm.model.HA, test = "LR")
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: good
##               LR Chisq Df Pr(>Chisq)
## distance       238.053  1    < 2e-16 ***
## wind             3.212  1    0.07312 .
## distance:wind    5.110  1    0.02379 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We see that under the likelihood ratio test and a significance threshold of 5% we would reject the null hypothesis of the interaction term coefficient being equal to zero. **This means we have statistical evidence that the model with the interaction term is better than the model without it in terms of predicting field goal makes.**

Note that if we decide to test the null hypothesis of the interaction term coefficient using the t-test within the glm model fit, it has a p-value slightly higher than 5%. In that test we would fail to reject the null hypothesis and conclude there is not statistical evidence of the interaction term being significant.

The difference has to do with the two tests analyzing different things. The likelihood ratio test focuses on model output and whether one model is significantly better than another by looking at the residiuals while the coefficient test within a model tries to estimate the precision to which we are able to estimate the coefficient. This slight difference leads to the conflicting results here.

The fact that the two tests do not align also highlights the arbitrariness of significance thresholds and p-values. Why do we choose 5%? It is due to convention more than anything else and is a nice rule of thumb.

Q: What is the p-value for wind testing?

```r
# data can be downloaded from https://www.chrisbilder.com/categorical/programs_and_data.html
df <- read.csv("~/Documents/Berkeley W271/Week 3 Discrete Response Model Part 3/Placekick.csv",
               stringsAsFactors = F)

glm.model.HA <- glm(formula = good ~ distance + wind + distance : wind, family = binomial(link = logit)
                    data = df)

Anova(glm.model.HA, test = "LR")
```

Solution:

```
## Analysis of Deviance Table (Type II tests)
##
## Response: good
##               LR Chisq Df Pr(>Chisq)
## distance       238.053  1    < 2e-16 ***
## wind             3.212  1    0.07312 .
## distance:wind    5.110  1    0.02379 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The p-value for wind testing can be obtained from the Anova table above using the car package. Note this function tests iterative likelihood ratio tests of including a term in the model against a model that includes previous terms. The reference order is how you write the formula.

What this means is the p-value for distance above references comparing a model with distance to the null model with only an intercept. Likewise the p-value for wind tests including wind and distance together against a model with just distance. **Therefore the p-value is approximately 0.07, and we fail to reject the null hypothesis that the better model is the one with just distance and not wind.**

However, the model with the interaction term as seen above is significantly better than the model without it, suggesting there are mediating effects to the relationship between distance and probability of making a field goal related to wind conditions.

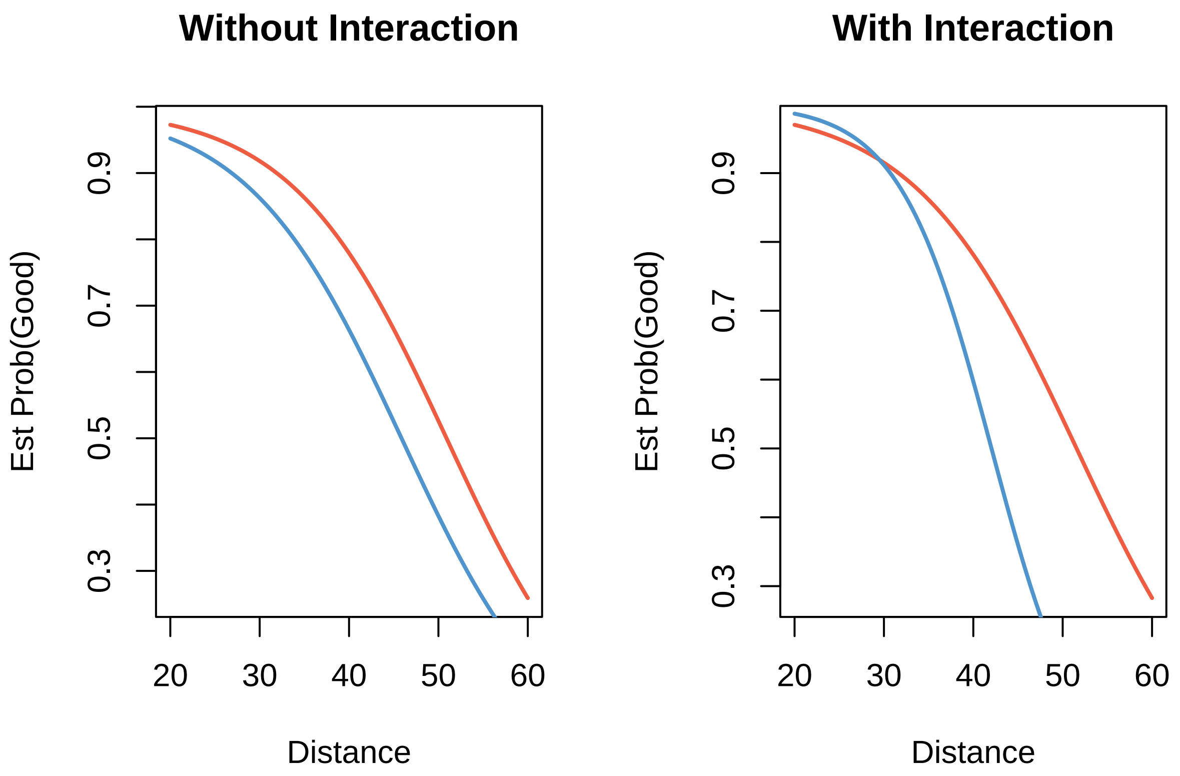
Q: Use the curve () function, which has been used a few times in the last two lectures, and reproduce the two plots from the lecture.

Solution: Note it is actually easier to not use the curve function here, so feel free to code your solution using something else.

```
# data can be downloaded from https://www.chrisbilder.com/categorical/programs_and_data.html
df <- read.csv("~/Documents/Berkeley W271/Week 3 Discrete Response Model Part 3/Placekick.csv",
               stringsAsFactors = F)

glm.model.HA <- glm(formula = good ~ distance + wind + distance : wind, family = binomial(link = logit)
                    data = df)

glm.model.H0 <- glm(formula = good ~ distance + wind, family = binomial(link = logit),
                    data = df)
```

```
with.interaction<-function(dmin = 20, dmax = 60) {

  wind0<-function(d) {
    new.data <- data.frame("distance" = d, "wind" = 0)
    pred.prob <- predict(glm.model.HA, newdata = new.data, type = "response")
    return(pred.prob)
  }

  wind1<-function(d) {
    new.data <- data.frame("distance" = d, "wind" = 1)
    pred.prob <- predict(glm.model.HA, newdata = new.data, type = "response")
    return(pred.prob)
  }

  curve(wind0, from = dmin, to = dmax, xlab = "Distance", ylab = "Est Prob(Good)", lwd = 2,
        col = "tomato2", main = "With Interaction")
  curve(wind1, from = dmin, to = dmax, xlab = "Distance", ylab = "Est Prob(Good)", lwd = 2,
        col = "steelblue3", add = T)

}

without.interaction<-function(dmin = 20, dmax = 60) {

  wind0<-function(d) {
    new.data <- data.frame("distance" = d, "wind" = 0)
    pred.prob <- predict(glm.model.H0, newdata = new.data, type = "response")
    return(pred.prob)
  }

  wind1<-function(d) {
    new.data <- data.frame("distance" = d, "wind" = 1)
    pred.prob <- predict(glm.model.H0, newdata = new.data, type = "response")
    return(pred.prob)
  }

  curve(wind0, from = dmin, to = dmax, xlab = "Distance", ylab = "Est Prob(Good)", lwd = 2,
        col = "tomato2", main = "Without Interaction")
  curve(wind1, from = dmin, to = dmax, xlab = "Distance", ylab = "Est Prob(Good)", lwd = 2,
        col = "steelblue3", add = T)

}
```

```
par(mfrow=c(1,2))
without.interaction()
with.interaction()
```



Note how the left plot without the interaction term is just a shift down when wind equal one due to the fact that the only difference is the inclusion of the marginal intercept term when wind is one. By contrast the right plot shows both a shift down when wind is one but also a changing slope due to the inclusion of the interaction between distance and wind.

3.5 Quadratic Term: An Introduction

Q: How would you code the model below? Please type your answer in the text box.

Solution: We want to code the following model:

$logit(\pi) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1^2 + \beta_4 x_2^2 + \beta_5 x_1 x_2$

Remember that we can specify interaction terms using ":" (just include interaction term), "*" (include main effects and interaction term), or use parentheses along with a square exponent (include main effects and interaction term). The first two options are more common, and it is suggest to use the first option of ":" to make it more clear what model is being fit.

We can include transformations of particular variables using the $I()$ function which performs the transformation on the variable before fitting the model.

Using those rules the code below will fit the desired model.

```
# create data set to show the code works
n <- 500
df <- data.frame("x1" = rnorm(n), "x2" = rnorm(n))
e <- rnorm(n)
df$y <- exp(df$x1 + df$x2^2 + df$x1 * df$x2 + e) /(1 + exp(df$x1 + df$x2^2 + df$x1 * df$x2 + e)) > 0.5

glm.model <- glm(y ~ x1 + x2 + I(x1^2) + I(x2^2) + x1:x2, data = df, family = binomial(link = "logit"))
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
summary(glm.model)
```

```
##
## Call:
## glm(formula = y ~ x1 + x2 + I(x1^2) + I(x2^2) + x1:x2, family = binomial(link = "logit"),
##     data = df)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.06993  -0.61821   0.09269   0.68276   2.38136
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.1883     0.1725  -1.092   0.2748
## x1            1.8003     0.2179   8.264  < 2e-16 ***
## x2            0.4426     0.2229   1.985   0.0471 *
## I(x1^2)       0.1474     0.1599   0.922   0.3565
## I(x2^2)       1.8931     0.2546   7.435 1.05e-13 ***
## x1:x2         2.0414     0.3389   6.024 1.70e-09 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 656.79  on 499  degrees of freedom
## Residual deviance: 401.74  on 494  degrees of freedom
## AIC: 413.74
##
## Number of Fisher Scoring iterations: 7
```

3.6 Quadratic Term: An Example

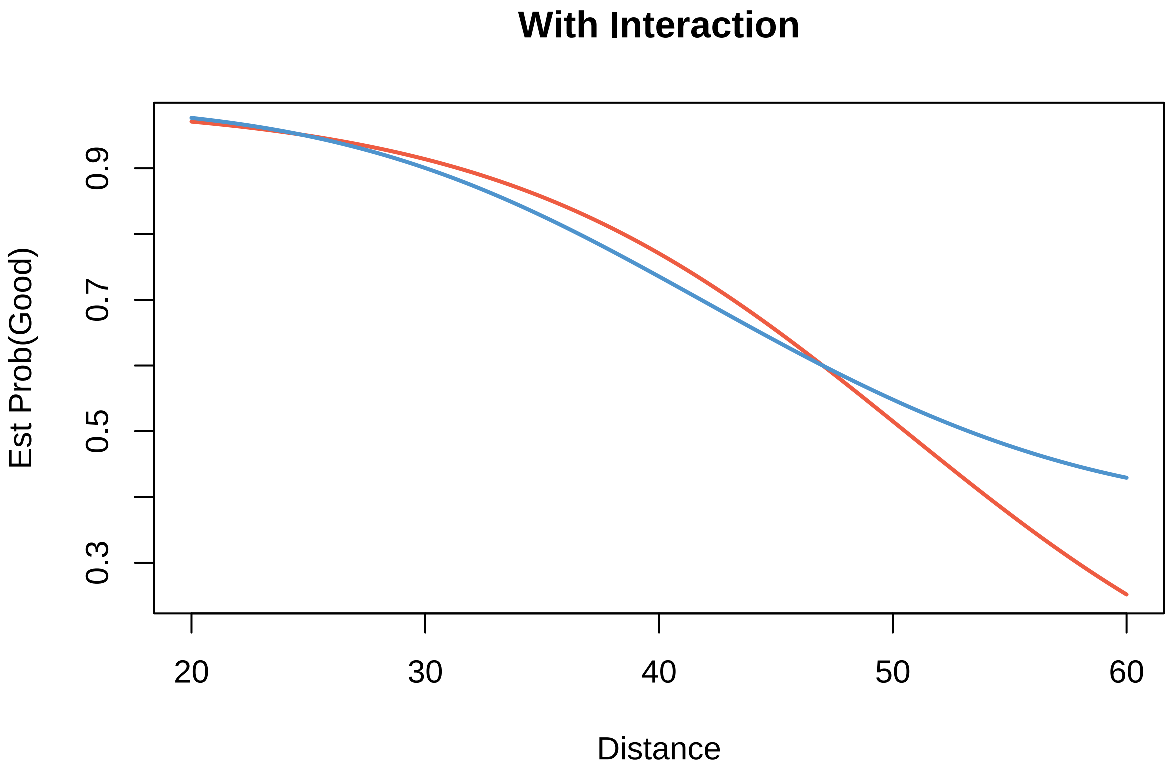
Q: Reproduce the plot and submit the R script.

```
# data can be downloaded from https://www.chrisbilder.com/categorical/programs_and_data.html
df <- read.csv("~/Documents/Berkeley W271/Week 3 Discrete Response Model Part 3/Placekick.csv",
```

```r
                stringsAsFactors = F)

glm.model.linear <- glm(formula = good ~ distance, family = binomial(link = logit), data = df)
glm.model.quad <- glm(formula = good ~ distance + I(distance^2), family = binomial(link = logit),
                      data = df)
```

```r
quad.comp<-function(dmin = 20, dmax = 60) {

  without.quad<-function(d) {
    new.data <- data.frame("distance" = d)
    pred.prob <- predict(glm.model.linear, newdata = new.data, type = "response")
    return(pred.prob)
  }

  with.quad<-function(d) {
    new.data <- data.frame("distance" = d)
    pred.prob <- predict(glm.model.quad, newdata = new.data, type = "response")
    return(pred.prob)
  }

  curve(without.quad, from = dmin, to = dmax, xlab = "Distance", ylab = "Est Prob(Good)", lwd = 2,
        col = "tomato2", main = "With Interaction")
  curve(with.quad, from = dmin, to = dmax, xlab = "Distance", ylab = "Est Prob(Good)", lwd = 2,
        col = "steelblue3", add = T)

}
```

```r
par(mfrow=c(1,1))
quad.comp()
```



Solution:

3.7 Categorical Explanatory Variables

Q: How would you represent an interaction between a categorical explanatory variable with four levels and one continuous explanatory variable?

Solution: We can model an interaction between a categorical and continuous variable in R using the normal interaction methods i.e. using ":". In the backend R will convert a character variable in a regression model to a factor variable which has different levels for each value of the character (or keep it unchanged if it is already a factor variable). R chooses a reference level based on alphabetical ordering of unique values / levels.

This reference level is the base group in the regression and is left out. Everything is measured relative to the reference level in terms of the impact of other levels of the categorical variable.

See the code below for an example of how this works.

```r
# create data set to show the code works
n <- 5000
df <- data.frame("x" = rnorm(n), "f" = c("A", "B", "C", "D"))
df$f <- relevel(factor(df$f), ref = "A")
df$y <- exp(df$x + as.numeric(df$f) + df$x * as.numeric(df$f) + rnorm(n)) /
  (1 + exp(df$x + as.numeric(df$f) + df$x * as.numeric(df$f) + rnorm(n))) > 0.5

glm.model <- glm(y ~ x + f + x:f, data = df, family = binomial(link = "logit"))

summary(glm.model)
```

```
##
## Call:
## glm(formula = y ~ x + f + x:f, family = binomial(link = "logit"),
##     data = df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5307  -1.0384   0.6239   1.0466   1.7705
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.03739    0.06038   0.619   0.5358
## x            0.79526    0.06881  11.558   <2e-16 ***
## fB           0.16498    0.08633   1.911   0.0560 .
## fC           0.17572    0.08583   2.047   0.0406 *
## fD           0.19389    0.08588   2.258   0.0240 *
## x:fB         0.11595    0.09999   1.160   0.2462
## x:fC         0.04101    0.09883   0.415   0.6782
## x:fD         0.04017    0.09876   0.407   0.6842
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 6906.1  on 4999  degrees of freedom
## Residual deviance: 6203.3  on 4992  degrees of freedom
## AIC: 6219.3
##
## Number of Fisher Scoring iterations: 4
```

Q: How would you interpret the interaction in the previous example question?

Solution: We can write the model above as follows:

$$logit(\pi_i) = \beta_0 + \beta_1 I(f = B) + \beta_2 I(f = C) + \beta_3 I(f = D) + \beta_4 x + \beta_5 x I(f = B) + \beta_6 x I(f = C) + \beta_7 x I(f = D)$$

Note what happens when we have f equal to each value:

$f = A \implies logit(\pi_i) = \beta_0 + \beta_4 x$

$f = B \implies logit(\pi_i) = \beta_0 + \beta_1 + \beta_4 x + \beta_5 x$

$f = C \implies logit(\pi_i) = \beta_0 + \beta_2 + \beta_4 x + \beta_6 x$

$f = D \implies logit(\pi_i) = \beta_0 + \beta_3 + \beta_4 x + \beta_7 x$

Comparing the equations we can see that each value of the categorical variable f results in a different regression equation with a different intercept and slope term for each value of f. **Therefore, the interaction term means that the relationship between the continuous variable and outcome differs for each value of the categorical variable. The interaction term controls the degree to which the slope and relationship changes.**

Q: How would you represent an interaction between a categorical explanatory variable with four levels and another categorical explanatory variable with three levels?

Solution: We can model an interaction between two categorical variables in R using the normal interaction methods i.e. using “:”. In the backend R will convert each character variable in a regression model to a factor variable which has different levels for each value of the character (or keep it unchanged if it is already a factor variable). R chooses a reference level based on alphabetical ordering of unique values / levels.

This reference level is the base group in the regression and is left out. Everything is measured relative to the reference level in terms of the impact of other levels of the categorical variables.

See the code below for an example of how this works. Note that the selected reference groups for each factor are missing from the model output because they are absorbed into the intercept.

```
# create data set to show the code works
n.group <- 100
df <- expand.grid("f1" = c("A", "B", "C", "D"), "f2" = c("E", "F", "G"))
df <- df[rep(1:nrow(df), each = n.group),]
df$x <- rnorm(nrow(df))
df$f1 <- relevel(factor(df$f1), ref = "A")
df$f2 <- relevel(factor(df$f2), ref = "E")
df$y <- exp(df$x + as.numeric(df$f1) + as.numeric(df$f2) + as.numeric(df$f1) * as.numeric(df$f2) + rnorm
  (1 + exp(df$x + as.numeric(df$f1) + as.numeric(df$f2) + as.numeric(df$f1) * as.numeric(df$f2) + rnorm

glm.model <- glm(y ~ f1 + f2 + f1:f2, data = df, family = binomial(link = "logit"))

summary(glm.model)
```

```
##
## Call:
## glm(formula = y ~ f1 + f2 + f1:f2, family = binomial(link = "logit"),
##     data = df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6894  -1.4689   0.8446   0.8782   0.9943
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  7.538e-01  2.144e-01   3.516 0.000438 ***
## f1B          7.892e-15  3.032e-01   0.000 1.000000
## f1C          3.989e-01  3.175e-01   1.257 0.208912
## f1D          4.635e-02  3.045e-01   0.152 0.879014
## f2F          3.448e-01  3.151e-01   1.094 0.273789
## f2G         -4.559e-02  3.020e-01  -0.151 0.880003
## f1B:f2F     -6.513e-01  4.328e-01  -1.505 0.132324
## f1C:f2F     -8.342e-01  4.457e-01  -1.872 0.061264 .
## f1D:f2F     -2.977e-01  4.401e-01  -0.676 0.498786
## f1B:f2G      9.193e-02  4.288e-01   0.214 0.830246
## f1C:f2G     -2.598e-01  4.400e-01  -0.590 0.554922
## f1D:f2G      9.277e-02  4.308e-01   0.215 0.829492
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1484.2  on 1199  degrees of freedom
## Residual deviance: 1476.4  on 1188  degrees of freedom
## AIC: 1500.4
##
```

```
## Number of Fisher Scoring iterations: 4
```

Mathematically, we can write the model above as follows:

$$logit(\pi_i) = \beta_0 + \beta_1 I(f_1 = B) + \beta_2 I(f_1 = C) + \beta_3 I(f_1 = D) + \beta_4 (f_2 = F) + \beta_5 I(f_2 = G)$$
$$+\beta_6 I(f_1 = B) * I(f_2 = F) + \beta_7 I(f_1 = B) * I(f_2 = G) + \beta_8 I(f_1 = C) * I(f_2 = F)$$
$$+\beta_9 I(f_1 = C) * I(f_2 = G) + \beta_{10} I(f_1 = D) * I(f_2 = F) + \beta_{11} I(f_1 = D) * I(f_2 = G)$$

Note what happens when we have f1 equal to each value:

$f1 = A, f2 = E \implies logit(\pi_i) = \beta_0$

$f1 = A, f2 = G \implies logit(\pi_i) = \beta_0 + \beta_5$

$f1 = C, f2 = E \implies logit(\pi_i) = \beta_0 + \beta_2$

$f1 = D, f2 = G \implies logit(\pi_i) = \beta_0 + \beta_3 + \beta_5 + \beta_{11}$

And so on.

Comparing the equations we can see that each set of values of the categorical variables f1 and f2 results in a different regression equation with a different intercept. **Therefore, the interaction terms here are effectively selecting different intercepts for each group in the data i.e. combination of f1 and f2 and measuring the marginal impact to y of each group**. This model is effectively return the group average in the data for each combination of the factor variables.

3.8 Odds Ratio in the Context of Categorical Explanatory Variables

Q: In the example above, which method (B or C) reduces the estimated odds more?

Solution: We have estimated the following model: $logit(\pi_i) = -0.67 + 0.22I(Infest2) - 0.79I(C) + 0.52I(N)$

As stated the odds ratio for comparing level N to level B is $e^{0.52} \approx 1.68$. This is the increase in likelihood that a plant shows symptoms for level N compared to level B. We can flip it to find the chances that a plant in level B has symptoms compared to level N, which is $1/e^{0.52} \approx 0.59$.

Using a biological control (the reference group) takes the odds from 1 to 0.59, which is a $(0.59-1)/1 = -41\%$ reduction in the odds of showing symptoms.

Comparing level N to level C, the odds are $e^{0.52-(-0.79)} \approx 3.71$. Following the same logic as above, we can flip the odds ratio and say that using chemical control takes the odds from 1 to 0.27 compared to using no control, which is a $(0.27-1)/1 = -73\%$ reduction in the odds of showing symptoms.

Note that the odds ratio of level N to level B is 1.69, and the odds ratio of level C to level N is 0.27.

The odds ratio of level C to level B is $e^{-0.79} = 0.45$, which is the product of 1.69 and 0.27. Hence, in logistic regression you can multiply odds ratios together to cancel things out i.e. $\frac{Odds_C}{Odds_B} = \frac{Odds_C}{Odds_N} * \frac{Odds_N}{Odds_B}$ due to the properties of exponents.

This means that using a control method reduces the odds of a plant showing symptoms relative to a biological method. We can also see this comparing the level to which C reduces the odds when N is the reference group i.e. in the denominator of the odds ratio. Compared to no control, using a chemical control reduced the odds by 73% while a biological control only reduced the odds by 41%.