Interaction vs. Non-Interaction in models with one continuous and one categorical predictor

Scott Nelson

9/16/2020

The main difference between interaction effects and non-interaction effects has to do with the slopes. When creating a model with a continuous and categorical predictor that *do not* interact, you are essentially trying to come up with the best slope for the continous predictor that can account for each of the different levels of the categorical predictor. The coeffecient for the continous predictor is the slope and the coeffecients for each level of the categorical predictor are summed with the reference level to get different intercepts.

When creating a model with a continous and categorical predictor that *do* interact, you are trying to come up with the unique best slope for each level of the categorical predictor. So in this case, you not only have varying intercepts, but you also have varying slopes! That's the main difference between interaction and non-interaction.

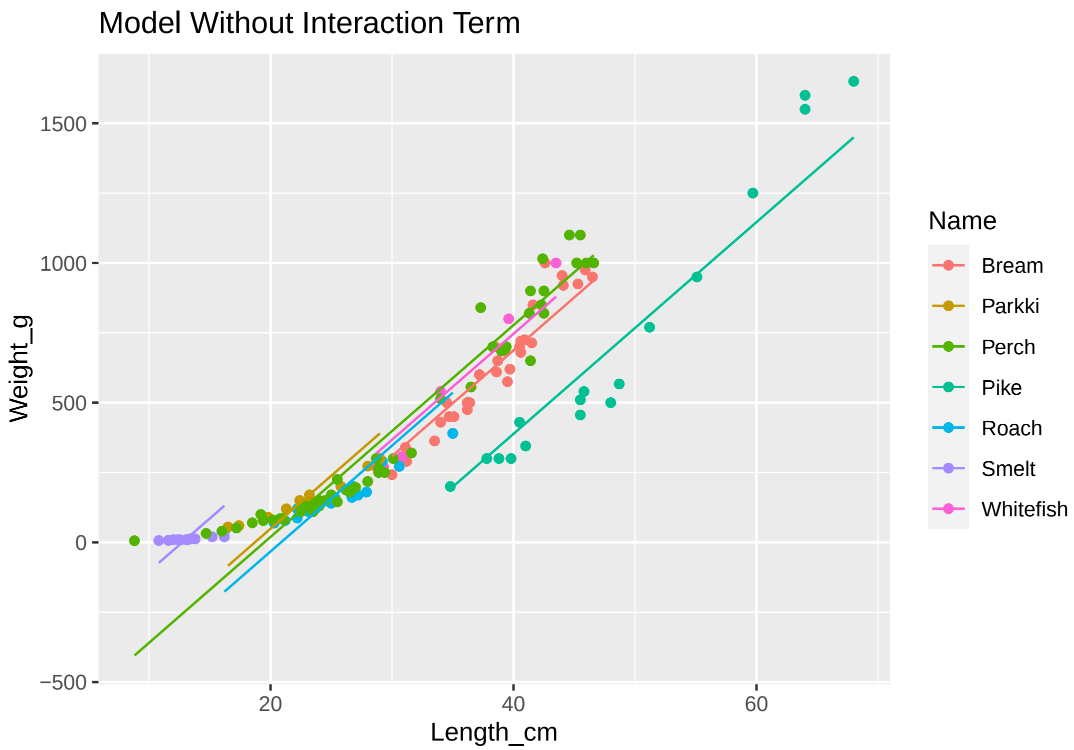
Let's look at the Fish data as an example.

```
library(tidyverse) # call the tidyverse package
fish <- read.csv("http://jeffreyheinz.net/classes/20F/materials/fishCatchData.csv") # data
fish <- fish %>%
  filter(!is.na(Weight_g)) %>% # this line removes the NA values in the Weight_g column
  mutate(LengthCentered = Length_cm - mean(Length_cm), # add centered length values
         Name = factor(Name)) # and make sure Name column type is Factor
```

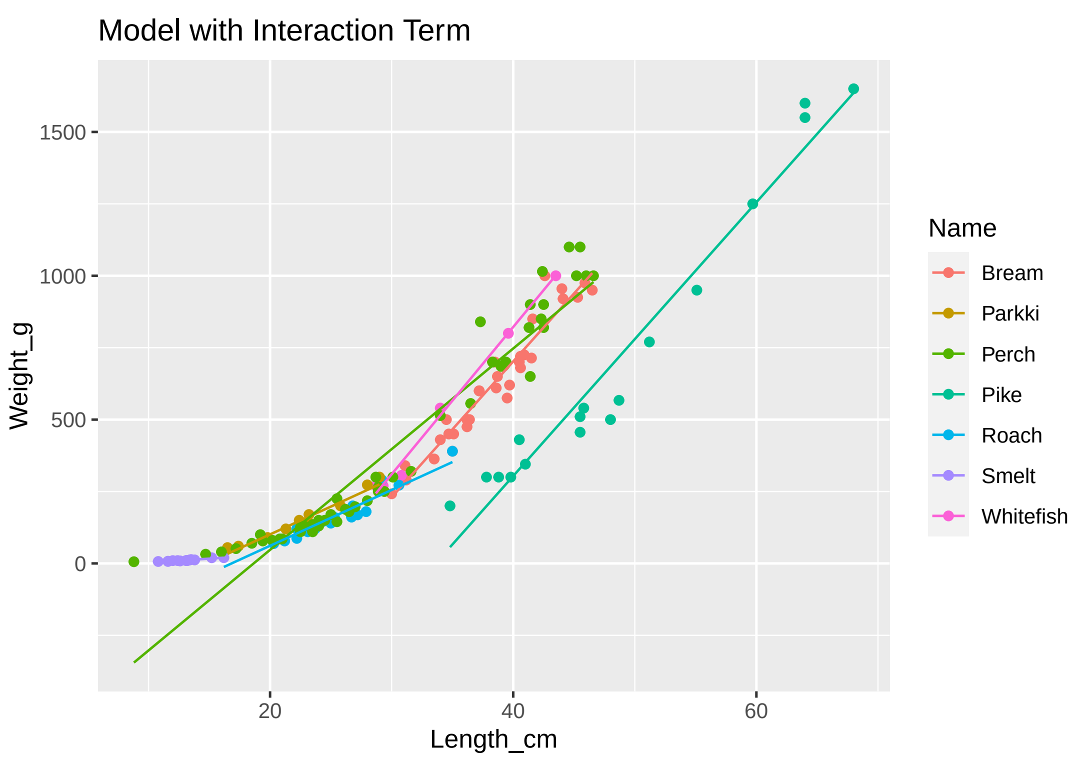
Let's make two models now. m1 will be our non-interactive model and m2 will be our interactive model. We should first look at the models and the corresponding graphs to get an intuitive idea of what is going on/what they look like. This first graph shows the non-interactive model. Here, we are using the predict() function in order to generate the predicted weight values based on the model and then using this to generate lines for each level of the categorical predictor. What you should notice is that each line has the same slope, but if you extended the lines to 0, they would have different intercepts.[1] The second graph shows the interactive model. We once again use the predict() function in order to generate the predicted weight and using this to generate lines. The important thing to notice here is that there is a different intercept **and** slope for each level of the categorical predictor.

```
m1 <- lm(Weight_g~Length_cm+Name, data = fish) # non-interactive model
NonInteractiveFish <- cbind(fish,pred = predict(m1)) # add prediction column based on m1
ggplot(data=NonInteractiveFish, aes(x=Length_cm, y=Weight_g, color=Name)) +
  geom_point() +
  geom_line(aes(y=pred))+
  ggtitle("Model Without Interaction Term")
```

[1] The reason that the line segments don't extend to the intercept is because the range is based off of the observed values. This is a pseudo-hack, but is useful for getting the basic point across. The actual model results in a line and not a line segment. It is possible to graph this with ggplot but is much more labor inducing. If you want to see how this works, you can get the coefficients for the model and then use geom_abline() with arguments for slope and intercept. The slope will always be the coefficient for the continuous variable (length in our case) while the intercept will be the model intercept coefficent + the coefficient value for the level you are interested in (0 if interested in the base level).




```r
m2 <- lm(Weight_g~Length_cm*Name, data = fish) # interactive model
InteractiveFish <- cbind(fish,pred = predict(m2)) # add prediction column based on m2
ggplot(data=InteractiveFish, aes(x=Length_cm, y=Weight_g, color=Name)) +
  geom_point() +
  geom_line(aes(y=pred)) +
  ggtitle("Model with Interaction Term")
```



How to derive these lines from the coefficients.

Non-Interactive model

Let's ignore everything except for the coefficients. These are shown below.

```r
coef(m1)
```

```
##   (Intercept)     Length_cm    NameParkki     NamePerch      NamePike
##    -828.61219      37.89504     119.76800      90.24106    -298.83903
##     NameRoach     NameSmelt NameWhitefish
##      38.09770     345.80187      59.18079
```

Notice that the Length_cm coefficient is approximately 37.9. This is the slope **for each level** of the categorical predictor. What this means in this model is that the weight of a fish increases by 37.9 grams for every increase of 1cm in length. All of the other coefficients are how we determine the intercepts for each level. This is done by adding the base level with the level of interest. So if we wanted to know the intercept for Smelt we would add -828.6 with 345.8 and get -482.8. Therefore, we could isolate the model to just Smelt by looking at the line y = 37.9(x) - 482.8 where y is the predicted weight (g) given a length (cm) value x. You can play the same game for every level of the categorical variable. The slope will always be 37.9, but the intercept will change.

Interactive model

Again, let's look at the coefficients. These are shown below.

```r
coef(m2)
```

```
##            (Intercept)              Length_cm             NameParkki
##          -1187.1185430             47.2347180            901.8069216
##              NamePerch               NamePike              NameRoach
##            534.3314057           -412.1963524            860.3864491
##              NameSmelt          NameWhitefish   Length_cm:NameParkki
##           1162.5093566            -39.4186021            -27.9230792
##    Length_cm:NamePerch     Length_cm:NamePike    Length_cm:NameRoach
##            -12.2338245              0.3460031            -27.8286966
##    Length_cm:NameSmelt Length_cm:NameWhitefish
##            -44.4893558              3.9805461
```

Due to the interactions in this model, we have more coefficients to deal with. Unlike last time, the coefficient for Length_cm is not the slope for every level of the categorical predictor but instead is **the slope for the base level only**. Like last time, the interecpt coefficient is equal to the intercept for the base level. What this means is that for Bream, we can predict their weight given the equation y = 47.2(x) - 1187.1. Now, say we wanted to figure out how to predict the values for Smelt again. We would calculate the intercept the same we did last time by adding the model intercept with the NameSmelt coefficent value. This gets us -1187.1 + 1162.5 = -24.6. To calculate the slope value for Smelt we have to add the base level slope with Length_cm:NameSmelt coefficient value. This gives us 47.2 + (-44.5) or 2.7. So the equation for calculating the weight for Smelt is y = 2.7(x) -24.6. Notice that we now have a different slope value and a different intercept value!

The important takeaway from this is that non-interaction models with one categorical and one continuous predictor give us individual intercepts by level while interaction models with one categorical and one continuous predictor give us individual intercepts and individual slopes by level.