Lab Assignment #5

Nick Noel & Liz VIlla

Due March 8, 2023

Instructions

The purpose of this lab is to introduce more advanced regression strategies that were probably not covered in Math 338.

In this lab, we will be working with four datasets. Three (`Boston`, `Carseats`, and `Wage`) are contained in the `ISLR2` package. Information about these datasets can be found by searching R help for them.

The fourth dataset, `RateMyProfessor`, needs to be downloaded from Canvas. This dataset contains the overall average rating from https://www.ratemyprofessors.com/ for over 22,000 professors, as collected by Murray et al. (2020). A data dictionary for the dataset can be found at https://github.com/murrayds/aa_rmp/tree/master/data (note that I removed a bunch of variables so that you're downloading a 2 MB dataset instead of a much larger one).

```
library(ISLR2)
library(ggplot2)
library(dplyr)
library(broom) # See Problem 3b

RateMyProfessor <- read.csv("RateMyProfessor.csv")
```

This lab assignment is worth a total of **15 points**.

Problem 1: Indicator Variables

Part a (Code: 0.5 pts)

Run the code in ISLR Lab 3.6.6. Put each chunk from the textbook in its own chunk.

```
lm.fit <- lm(Sales ~ . + Income:Advertising + Price:Age, data = Carseats)

summary(lm.fit)
```

```
##
## Call:
## lm(formula = Sales ~ . + Income:Advertising + Price:Age, data = Carseats)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.9208 -0.7503  0.0177  0.6754  3.3413
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)           6.5755654  1.0087470   6.519 2.22e-10 ***
```

```
## CompPrice           0.0929371  0.0041183  22.567  < 2e-16 ***
## Income              0.0108940  0.0026044   4.183 3.57e-05 ***
## Advertising         0.0702462  0.0226091   3.107 0.002030 **
## Population          0.0001592  0.0003679   0.433 0.665330
## Price              -0.1008064  0.0074399 -13.549  < 2e-16 ***
## ShelveLocGood       4.8486762  0.1528378  31.724  < 2e-16 ***
## ShelveLocMedium     1.9532620  0.1257682  15.531  < 2e-16 ***
## Age                -0.0579466  0.0159506  -3.633 0.000318 ***
## Education          -0.0208525  0.0196131  -1.063 0.288361
## UrbanYes            0.1401597  0.1124019   1.247 0.213171
## USYes              -0.1575571  0.1489234  -1.058 0.290729
## Income:Advertising  0.0007510  0.0002784   2.698 0.007290 **
## Price:Age           0.0001068  0.0001333   0.801 0.423812
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.011 on 386 degrees of freedom
## Multiple R-squared:  0.8761, Adjusted R-squared:  0.8719
## F-statistic:   210 on 13 and 386 DF,  p-value: < 2.2e-16
```

```r
attach(Carseats)

contrasts(ShelveLoc)
```

```
##        Good Medium
## Bad       0      0
## Good      1      0
## Medium    0      1
```

Part b (Explanation: 1 pt)

Interpret the slope estimate corresponding to `ShelveLocGood` in the model fit in part (a).

When compared to a bad shelf location, placing a child car seat in a good shelf location increases the expected sales by about 5 car seats.

Part c (Code: 1 pt; Explanation: 1.5 pts)

Using the RateMyProfessor dataset, fit a linear model predicting the overall rating of a professor (`overall`) from the difficulty rating (`difficulty`), chili pepper rating (`hotness`), and rank (`rank`). What are the reference levels for each categorical variable? How do you know?

```r
RateMyProfessor <- read.csv("RateMyProfessor.csv")

lm.fitrmp <- lm(overall ~ difficulty + hotness + rank, data = RateMyProfessor)

summary(lm.fitrmp)
```

```
##
## Call:
## lm(formula = overall ~ difficulty + hotness + rank, data = RateMyProfessor)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.3163 -0.5038  0.0385  0.5274  2.4384
##
```

```
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)              5.301027   0.025082 211.351  < 2e-16 ***
## difficulty              -0.587803   0.006605 -88.997  < 2e-16 ***
## hotnesshot               0.636429   0.012819  49.649  < 2e-16 ***
## rankAssociate Professor -0.050371   0.015088  -3.338 0.000844 ***
## rankProfessor           -0.047455   0.014747  -3.218 0.001293 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7475 on 22033 degrees of freedom
## Multiple R-squared:  0.3581, Adjusted R-squared:  0.3579
## F-statistic:  3072 on 4 and 22033 DF,  p-value: < 2.2e-16
```

```r
unique(RateMyProfessor[c("hotness", "rank")])
```

```
##      hotness                rank
## 1       cold Associate Professor
## 2        hot Associate Professor
## 4       cold           Professor
## 24       hot           Professor
## 33      cold Assistant Professor
## 1298     hot Assistant Professor
```

The reference level for hotness is cold and for rank it is assistant professor. We know this because these categories do not appear in the coefficients, indicating that they were selected as the reference levels for the remaining categories to be compared to.

Part d (Explanation: 1.5 pts)

Holding difficulty constant, which of the following instructors would be predicted to have the highest overall rating? Which would be predicted to have the lowest overall rating? Explain your reasoning.

- Attractive Assistant Professor
- Attractive Associate Professor
- Attractive Professor
- Less-attractive Assistant Professor
- Less-attractive Associate Professor
- Less-attractive Professor

Attractive Assistant Professors would be predicted to have the highest overall rating since holding all else constant, being an associate professor or professor decreases the average overall rating, and being hot increases the average overall rating. A less-attractive associate professor would be predicted to have the lowest overall rating since being rated as hot increases the average overall rating compared to being less attractive and compared to being an assistant professor, associate professors on average have lower overall ratings than professors.

Problem 2: Interaction Terms

Part a (Code: 0.5 pts)

Run the single line of code in ISLR Lab 3.6.4.

```r
summary(lm(medv~lstat * age , data = Boston))
```

```
##
## Call:
```

```
## lm(formula = medv ~ lstat * age, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.806  -4.045  -1.333   2.085  27.552
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 36.0885359  1.4698355  24.553  < 2e-16 ***
## lstat       -1.3921168  0.1674555  -8.313 8.78e-16 ***
## age         -0.0007209  0.0198792  -0.036   0.9711
## lstat:age    0.0041560  0.0018518   2.244   0.0252 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.149 on 502 degrees of freedom
## Multiple R-squared:  0.5557, Adjusted R-squared:  0.5531
## F-statistic: 209.3 on 3 and 502 DF,  p-value: < 2.2e-16
```

```
summary(lm(medv~lstat + age , data = Boston))
```

```
##
## Call:
## lm(formula = medv ~ lstat + age, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.981  -3.978  -1.283   1.968  23.158
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 33.22276    0.73085  45.458  < 2e-16 ***
## lstat       -1.03207    0.04819 -21.416  < 2e-16 ***
## age          0.03454    0.01223   2.826  0.00491 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.173 on 503 degrees of freedom
## Multiple R-squared:  0.5513, Adjusted R-squared:  0.5495
## F-statistic:   309 on 2 and 503 DF,  p-value: < 2.2e-16
```

Part b (Explanation: 2 pts)

Notice that `age` is a significant predictor of `medv` in the model without the interaction term (from ISLR Lab 3.6.3 on Lab 4), but it is no longer a significant predictor of `medv` once we add in the interaction term. The p-value is huge (0.971!). What do you think is happening here? Are we okay to remove the `age` variable from the model with the interaction term? Why or why not?

The result of adding our interaction term, age:lstat, yields us the same amount of significance regarding our model but has the added benefit of accounting for the interaction between the two terms. In all honesty, it seems like the model can go with or without the the interaction term as we are still seeing age be a significant factor, just regarding the interaction in this new model. We cannot take out just the age variable and leave in the interaction term though as a result of the hierarchical principal that states such.

Part c (Code: 1 pt; Explanation: 1.5 pts)

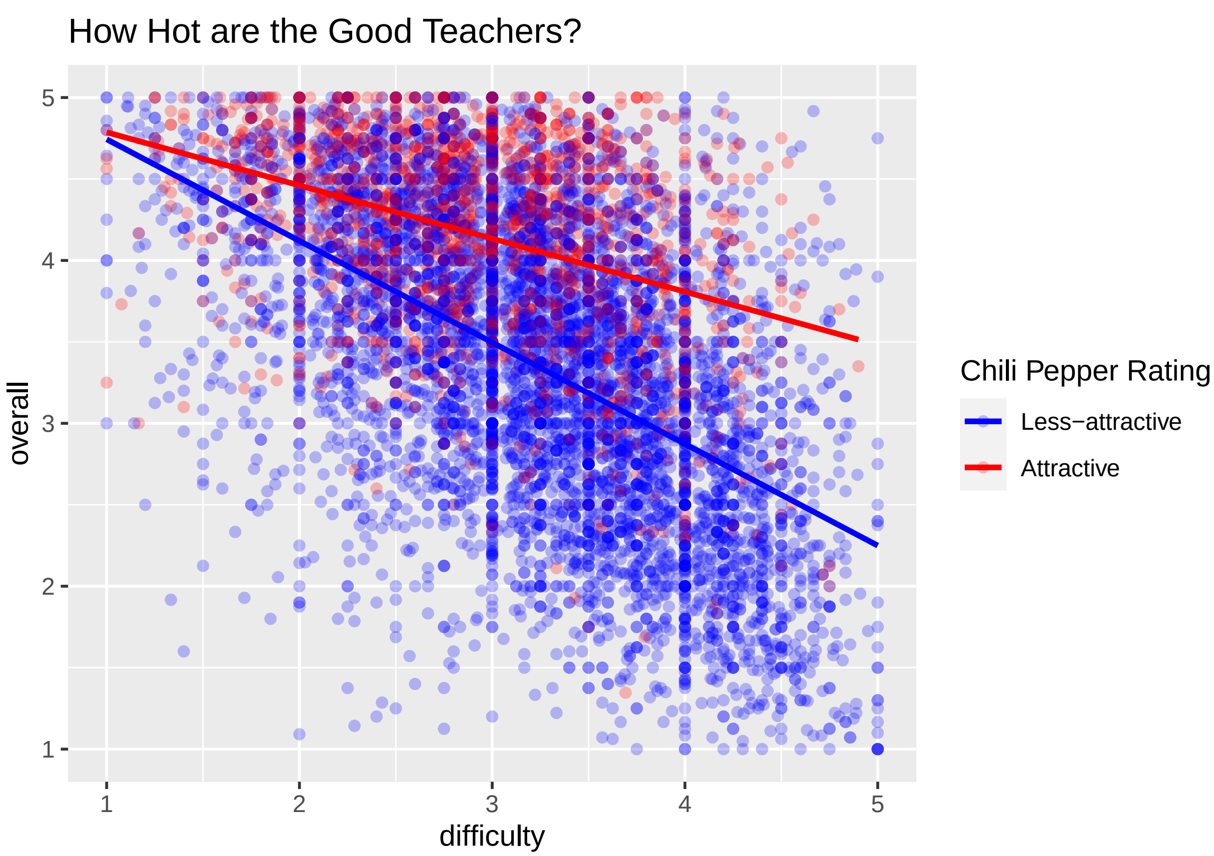
Create a new dataset, `associates`, by `filter`ing the `RateMyProfessor` dataset to include only the Associate Professors.

```
RateMyProfessor <- read.csv("RateMyProfessor.csv")
associates <- RateMyProfessor %>%
  filter(rank == "Associate Professor")
```

Next, complete this code chunk to create a graph of overall rating vs. difficulty rating for the associate professors, with "hot" professors shown in red and "cold" professors shown in blue. Remember to delete `eval = FALSE` once you get the code to run!

```
ggplot(data = associates, aes(x = difficulty, y = overall)) +
  geom_point(alpha = .25, aes(color = hotness)) +
  geom_smooth(method = "lm", se = FALSE,aes(color = hotness)) +
  scale_color_manual(name = "Chili Pepper Rating",  # make legend nice
                     labels = c(hot = "Attractive",
                                cold ="Less-attractive"),
                     values = c(hot = "red", cold = "blue")) +
  ggtitle("How Hot are the Good Teachers?")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```
  # scale_fill_manual(breaks = c("cold", "hot"),
  #                   values=c("blue", "red"))
```

How does the difficulty of the professor modify the relationship between attractiveness and overall rating?

The difficulty of a professor does not seem to be too dependent on attractiveness but their overall rating definitely seems to show teachers with lower overall ratings having higher difficulties compared to teachers with higher ratings having less difficulty.

As difficulty increases, overall rating also decreases. Attractiveness affects that as more attractive professors will have a slope less steep than a cold professor.

Part d (Code: 1 pt; Computation and Explanation: 2 pts)

Using the `RateMyProfessor` dataset, fit a linear model predicting overall rating from the difficulty rating (`difficulty`), chili pepper rating (`hotness`), rank (`rank`), and an interaction term between `difficulty` and `hotness`.

```
# hist(RateMyProfessor$overall)

lm_for_teachers <- lm(overall ~ difficulty + hotness + rank + difficulty:hotness , data = RateMyProfess
summary(lm_for_teachers)
```

```
##
## Call:
## lm(formula = overall ~ difficulty + hotness + rank + difficulty:hotness,
##     data = RateMyProfessor)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4198 -0.4943  0.0486  0.5109  2.5324
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)             5.470868   0.026768 204.378  < 2e-16 ***
## difficulty             -0.640874   0.007239 -88.532  < 2e-16 ***
## hotnesshot             -0.252382   0.052806  -4.779 1.77e-06 ***
## rankAssociate Professor -0.048940   0.014987  -3.266  0.00109 **
## rankProfessor          -0.044282   0.014648  -3.023  0.00251 **
## difficulty:hotnesshot    0.297055   0.017128  17.343  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7425 on 22032 degrees of freedom
## Multiple R-squared:  0.3667, Adjusted R-squared:  0.3666
## F-statistic:  2552 on 5 and 22032 DF,  p-value: < 2.2e-16
```

Using your results, write out the least-squares regression equation predicting overall rating from difficulty for an attractive associate professor. Also, write out the least-squares regression equation predicting overall rating from difficulty for a less-attractive associate professor. Explain how you obtained each equation.

Total formula :

overall = 5.4708 - .6408(difficulty) - .2524(hot?) -.0489(Associate Professor?) - .0443(Professor) +.2971(difficulty)(hot?)

For an attractive associate professor:

overall = 5.4708 - .6408(difficulty) -.2524(1) - .0489(1) - 0 +.2971(difficulty)(1)

overall =5.1695- 0.3437(difficulty)

Do your equations support your conclusions from part (c)? Explain why or why not.

Our equations do in fact support our conclusion from part c for the most part. Being attractive changes our slope by increasing (making less steep of a decline) which can be seen in our plot.

Problem 3: Regression with Nonlinear Transformations of the Predictors

Part a (Code: 0.5 pts)

Run the first four code chunks in ISLR Lab 7.8.1 (up through the point where `fit2b` is created). Put each chunk from the textbook in its own chunk.

```
library(ISLR2)

fit <- lm(wage~poly (age, 4), data = Wage)
coef(summary(fit))
```

```
##                  Estimate Std. Error    t value     Pr(>|t|)
## (Intercept)    111.70361  0.7287409 153.283015 0.000000e+00
## poly(age, 4)1  447.06785 39.9147851  11.200558 1.484604e-28
## poly(age, 4)2 -478.31581 39.9147851 -11.983424 2.355831e-32
## poly(age, 4)3  125.52169 39.9147851   3.144742 1.678622e-03
## poly(age, 4)4  -77.91118 39.9147851  -1.951938 5.103865e-02
```

```
fit2 <- lm(wage~poly (age, 4, raw =T), data = Wage)
coef(summary(fit2))
```

```
##                              Estimate   Std. Error   t value     Pr(>|t|)
## (Intercept)             -1.841542e+02 6.004038e+01 -3.067172 0.0021802539
## poly(age, 4, raw = T)1   2.124552e+01 5.886748e+00  3.609042 0.0003123618
## poly(age, 4, raw = T)2  -5.638593e-01 2.061083e-01 -2.735743 0.0062606446
## poly(age, 4, raw = T)3   6.810688e-03 3.065931e-03  2.221409 0.0263977518
## poly(age, 4, raw = T)4  -3.203830e-05 1.641359e-05 -1.951938 0.0510386498
```

```
fit2a <- lm(wage~age+I(age^2)+I(age^3)+age^4, data = Wage)
coef(fit2a)
```

```
##   (Intercept)           age      I(age^2)      I(age^3)
## -7.524391e+01  1.018999e+01 -1.680286e-01  8.494522e-04
```

```
fit2b <- lm(wage~cbind(age, age^2, age^3, age^4), data = Wage)
```

Part b (Code: 1 pt)

In the code chunk below, create a data frame with a single variable, `age`, ranging from 18 to 80, then use the `augment` function (in the `broom` package) to obtain the predicted wage, standard error of the mean wage, and the lower and upper bounds of a 95% confidence interval for the population mean wage at each age. (You can use any of `fit`, `fit2`, `fit2a`, or `fit2b` - they should all give the same predictions.)

```
attach(Wage)
agelims <- range(Wage$age)

#create a data frame with a single variable, `age`, ranging from 18 to 80
age.grid <-data.frame(age = seq(from = agelims[1], to = agelims[2]))

#Use the `augment` function (in the `broom` package) to obtain the predicted wage
preds <- augment(fit2b, new_data = age.grid)
```

What is the 95% confidence interval for the population mean wage of 25-year-olds? 50-year-olds?

```r
broom::augment(fit2b, newdata = age.grid, interval = "confidence")
```

```
## # A tibble: 63 x 4
##      age .fitted .lower .upper
##    <int>   <dbl>  <dbl>  <dbl>
##  1    18    51.9   41.5   62.3
##  2    19    58.5   49.9   67.1
##  3    20    64.6   57.5   71.6
##  4    21    70.2   64.4   76.0
##  5    22    75.4   70.5   80.2
##  6    23    80.1   76.0   84.2
##  7    24    84.5   80.9   88.1
##  8    25    88.5   85.2   91.7
##  9    26    92.1   89.1   95.2
## 10    27    95.4   92.5   98.4
## # ... with 53 more rows
```

For age 25: 88.47380 85.21437 91.73322

For age 50: 119.57013 117.35377 121.78650