Multiple linear regression + Model selection

Dr. Maria Tackett

10.22.19

STA 199

[Click for PDF of slides]

Announcements

- Lab 06 **due Wednesday at 11:59p**
- Complete Reading 06 for Thursday
- Project proposal **due Friday at 11:59p**

The linear model with multiple predictors

Data: Riders in Florence, MA

The Pioneer Valley Planning Commission collected data in Florence, MA for 90 days from April 5 to November 15, 2005 using a laser sensor, with breaks in the laser beam recording when a rail-trail user passed the data collection station.

- **`hightemp`**: daily high temperature (in degrees Fahrenheit)
- **`volume`**: estimated number of trail users that day (number of breaks recorded)
- **`dayType`**: weekday or weekend

```
library(mosaicData)
data(RailTrail)
```

Main effects, numerical and categorical predictors

term	estimate
(Intercept)	-8.747
hightemp	5.348
dayTypeweekend	51.553

- For each additional degree Fahrenheit in the day's high temperature, there are predicted to be, on average, 5.3478168 (about 5) additional riders on the trail, holding all else constant.

- Days on the weekend are predicted to have, on average, 51.553496 (about 52) more riders on the trail than days that are weekdays, holding all else constant.

- Weekdays that have a high temperature of 0 degrees Fahrenheit are predicted to have -8.7469229 (about -9) riders, on average.

Modeling with interaction effects

```
m_int <- lm(volume ~ hightemp + dayType + hightemp*dayType,
            data = RailTrail)
kable(tidy(m_int) %>% select(term, estimate), format = "html", digits = 3)
```

term	estimate
(Intercept)	-51.224
hightemp	5.980
dayTypeweekend	186.377
hightemp:dayTypeweekend	-1.906

$$\widehat{volume} = -51.224 + 5.980\ hightemp + 186.377\ dayTypeweekend - 1.906\ hightemp \times day$$

Practice

Suppose you wish to fit a model using **`hightemp`** and **`summer`** to predict the number of riders on a trail. **`summer`** is 1 if the day is during the summer, 0 otherwise.

term	estimate
(Intercept)	-232.432
hightemp	9.294
summer1	576.081
hightemp:summer1	-8.349

1. Interpret the coefficient of **`summer1`**.
2. Write the model equation for days that are <u>not</u> during the summer.
3. Write the model equation for days that are during the summer.
4. Interpret the coefficient of **`highTemp`** for days during the summer.

Quality of fit in MLR

R^2

- R^2 is the percentage of variability in the response variable explained by the regression model.

```
glance(m_main)$r.squared
```

```
## [1] 0.3735356
```

```
glance(m_int)$r.squared
```

```
## [1] 0.3816309
```

- Clearly the model with interactions has a higher R^2.
- However using R^2 for model selection in models with multiple explanatory variables is not a good idea as R^2 increases when **any** variable is added to the model.

R^2 - first principles

$$R^2 = \frac{SS_{Reg}}{SS_{Total}} = 1 - \left(\frac{SS_{Error}}{SS_{Total}} \right)$$

Calculate R^2 based on the output below.

```
anova(m_main)
```

```
## Analysis of Variance Table
##
## Response: volume
##           Df Sum Sq Mean Sq F value    Pr(>F)
## hightemp   1 490744  490744  47.133 9.349e-10 ***
## dayType    1  49373   49373   4.742   0.03214 *
## Residuals 87 905841   10412
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Adjusted R^2

$$R^2_{adj} = 1 - \left(\frac{SS_{Error}}{SS_{Total}} \times \frac{n-1}{n-k-1} \right),$$

where n is the number of cases and k is the number of predictors in the model

- Adjusted R^2 doesn't increase if the new variable does not provide any new informaton or is completely unrelated.
- This makes adjusted R^2 a preferable metric for model selection in multiple regression models.

In pursuit of Occam's Razor

- Occam's Razor states that among competing hypotheses that predict equally well, the one with the fewest assumptions should be selected.
- Model selection follows this principle.
- We only want to add another variable to the model if the addition of that variable brings something valuable in terms of predictive power to the model.
- In other words, we prefer the simplest best model, i.e. parsimonious model.

Comparing models

It appears that adding the interaction actually increased adjusted R^2, so for now we'll use the model with the interactions

```
glance(m_main)$adj.r.squared
```

```
## [1] 0.3591341
```

```
glance(m_int)$adj.r.squared
```

```
## [1] 0.3600599
```

Model selection

Backwards elimination

- Start with **full** model (including all candidate explanatory variables and all candidate interactions)
- Remove one variable at a time, and select the model with the highest adjusted R^2
- Continue until adjusted R^2 does not increase

Forward selection

- Start with **empty** model
- Add one variable (or interaction effect) at a time, and select the model with the highest adjusted R^2
- Continue until adjusted R^2 does not increase

Model selection and interaction effects

If an interaction is included in the model, the main effects of both of those variables must also be in the model

If a main effect is not in the model, then its interaction should not be in the model.

Other model selection criteria

- Adjusted R^2 is one model selection criterion

- There are others out there (many many others!), we'll discuss some later in the course, and you may see some in future courses

Your turn

What's the ultimate Halloween candy?

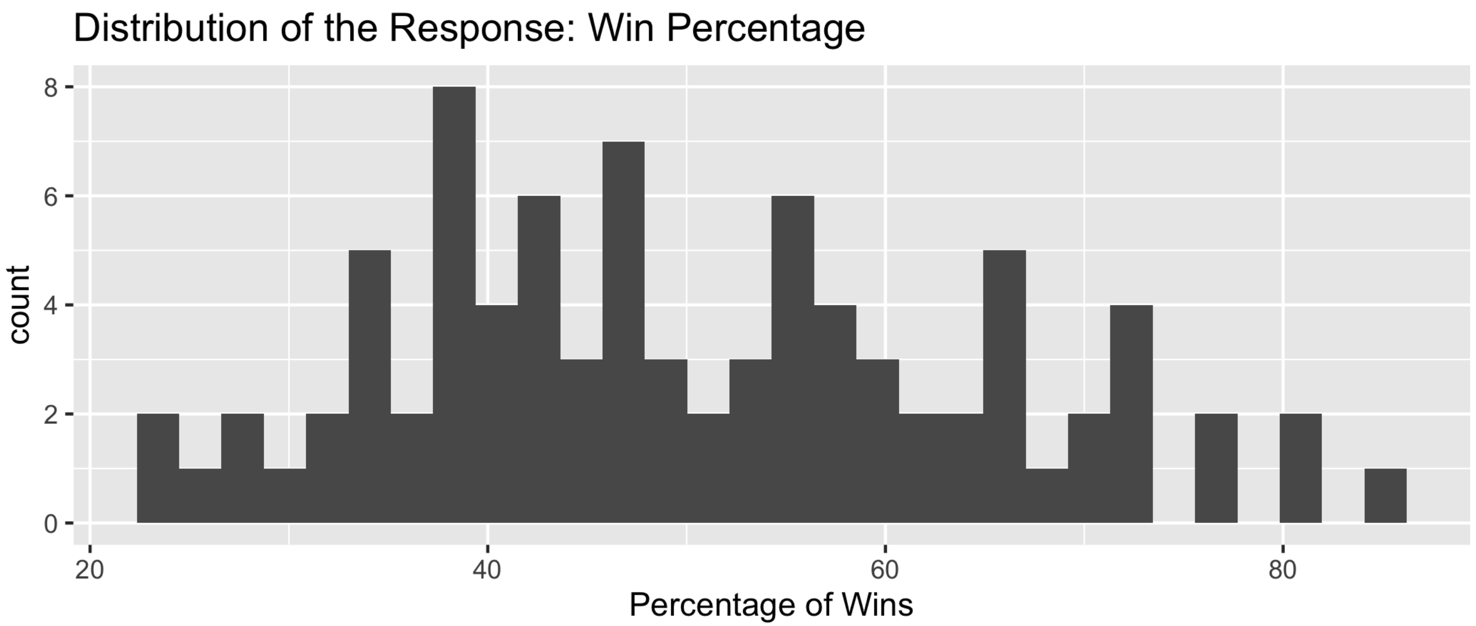
- In the 2017 article, The Ultimate Halloween Candy Power Ranking, Walt Hickey from FiveThirtyEight sought to find the best Halloween candy.
- To collect data, random candy matchups were generated and users selected their favorite of the two candies
  - There were about 296,000 matchups voted on by users from 8,371 different IP addresses

The Dataset

- We will use the **`candy_rankings`** dataset in the fivethirtyeight package
- Each row contains the characteristics and win percentage for a certain candy
- The response variable is **`winpercent`**, the overall percentage of times a candy won according to the 296,000 matchups
- type **`??candy_rankings`** in the console to see the other variables in the dataset

Distribution of response: `winpercent`

```r
ggplot(data = candy_rankings, aes(x = winpercent)) +
  geom_histogram() +
  labs(x = "Percentage of Wins",
       title = "Distribution of the Response: Win Percentage")
```



Your turn

- Work with your lab group in Rstudio Cloud
- **Project**: Ultimate Candy Rankings - Model Selection
- **Task**:
    - Use backwards elimination to do model selection. Make sure to show each step of decision (though you don't have to interpret the models at each stage).
    - Provide interpretations for the slopes for your final model and create at least one visualization that supports your narrative.
- We'll have two groups share their results in the beginning of next class

Planning

- You want to consider at least two interactions in the model
  - The interactions should be between a categorical variable and a numeric variable
- Remember if an interaction term is in the model, the main effects should also be in the model
- Consider 7 - 10 variables (including interactions) for the model