Week 5 Summary

Leo Soccio

Table of contents

Tuesday, Jan 17

> **!** TIL
>
> Include a *very brief* summary of what you learnt in this class here.
> Today, I learnt the following concepts in class:
>
> 1. Interpreting Regression Coefficients
> 2. Categorical Covariates
> 3. Reordering Factors + Setting a Baseline

Packages:

```r
library(tidyverse)
```

```
-- Attaching packages --------------------------------------- tidyverse 1.3.2 --
v ggplot2 3.4.0     v purrr   1.0.1
```

```
v tibble  3.1.8     v dplyr   1.1.0
v tidyr   1.3.0     v stringr 1.5.0
v readr   2.1.3     v forcats 1.0.0
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
```

```r
library(ISLR2)
library(cowplot)
library(kableExtra)
```

```
Attaching package: 'kableExtra'

The following object is masked from 'package:dplyr':

    group_rows
```

Provide more concrete details here. You can also use footenotes[1] if you like

Interpreting regression coefficients

Recall that the regression model is $y_i = \beta_0 + \beta_1 x_i + \epsilon_i$

y_i is the response, x_i is the covariate, ϵ_i is the error, β_0 and β_1 are the regression coefficients, and $i = 1, 2, ..., n$ are the indices for the observations.

Example using mtcars:

```r
library(ggplot2)
attach(mtcars)
```

```
The following object is masked from package:ggplot2:

    mpg
```

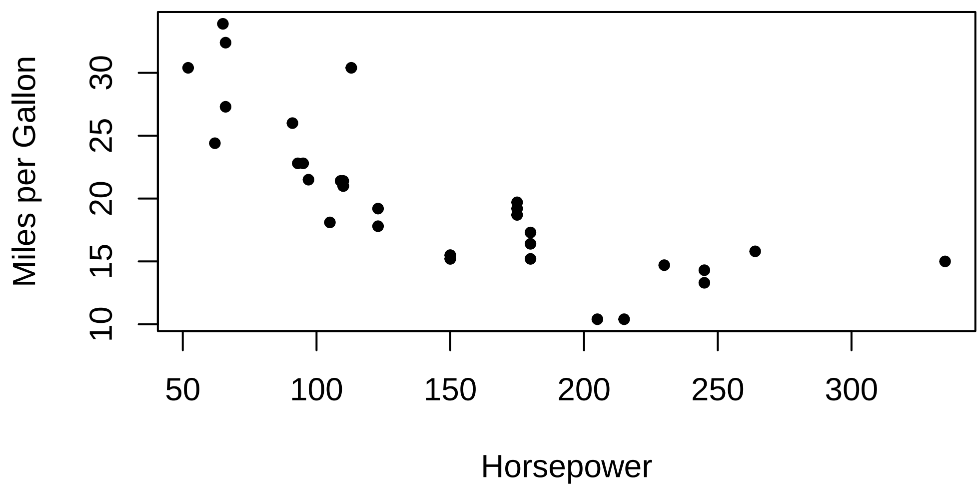
```r
mtcars %>% head()
```

[1] You can include some footnotes here

```
                   mpg cyl disp  hp drat    wt  qsec vs am gear carb
Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

```
x <- mtcars$hp
y<- mtcars$mpg

plot(x,y,pch=20, xlab="Horsepower", ylab="Miles per Gallon")
```




```
model<-lm(y~x)
summary(model)
```

```
Call:
lm(formula = y ~ x)

Residuals:
    Min      1Q  Median      3Q     Max
```

```
-5.7121 -2.1122 -0.8854  1.5819  8.2360

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 30.09886    1.63392  18.421  < 2e-16 ***
x           -0.06823    0.01012  -6.742 1.79e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.863 on 30 degrees of freedom
Multiple R-squared:  0.6024,    Adjusted R-squared:  0.5892
F-statistic: 45.46 on 1 and 30 DF,  p-value: 1.788e-07
```

For the intercept, a hypothetical car with 0 horsepower would have a predicted mpg of 30.099=β_0
For the slope, each increase of 1 horsepower decreases the predicted mpg by 0.068=β_1

Using Categorical Covariates

Return to the mtcars dataset, looking at the cyl variable. Also look at the iris dataset:

```
mtcars$cyl
```

```
 [1] 6 6 4 6 8 6 8 4 4 6 6 8 8 8 8 8 8 4 4 4 4 8 8 8 8 4 4 4 8 6 8 4
```

```
iris%>%head()
```

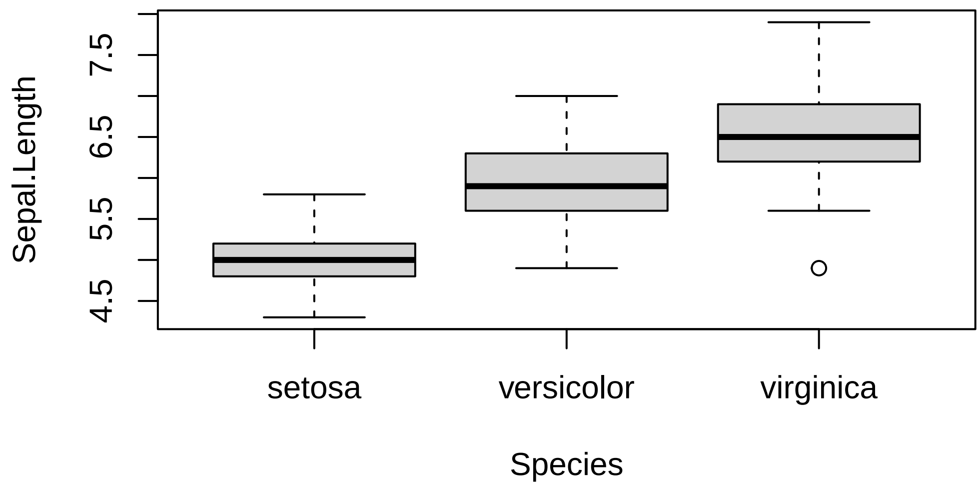
```
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.1         3.5          1.4         0.2  setosa
2          4.9         3.0          1.4         0.2  setosa
3          4.7         3.2          1.3         0.2  setosa
4          4.6         3.1          1.5         0.2  setosa
5          5.0         3.6          1.4         0.2  setosa
6          5.4         3.9          1.7         0.4  setosa
```

```
summary(iris$Species)
```

```
    setosa versicolor  virginica 
        50         50         50 
```

EDA for a potential relationship between Species and Sepal.Length:

```r
boxplot(Sepal.Length~Species, data=iris)
```



Run a linear regression model:

```r
iris_model <- lm(Sepal.Length~Species,iris)
iris_model
```

```
Call:
lm(formula = Sepal.Length ~ Species, data = iris)

Coefficients:
      (Intercept)  Speciesversicolor   Speciesvirginica
            5.006              0.930              1.582
```

With a categorical x, we can write the regression model the same way: $y_i = \beta_0 + \beta_1 x_i$, where $x \in (setosa, versicolor, virginica)$

We essentially have 3 different models:

- $y_i = \beta_0 + \beta_1 x_i =$setosa
- $y_i = \beta_0 + \beta_1 x_i =$versicolor

- $y_i = \beta_0 + \beta_1 x_i$ =virginica
- For the baseline (setosa), $\beta_1 = 0$ such that β_0 is the expected value for the baseline category.
- For the other two beta 1 values, they describe the change from the baseline to its category.(ex. setosa to versicolor and setosa to virginica)

Reordering Factors

Let's say we want to place virginica as the baseline.

```
iris$Species <- relevel(iris$Species,"virginica")
summary(iris$Species)
```

```
 virginica     setosa versicolor
        50         50         50
```

```
new_iris_model<-lm(Sepal.Length~Species,iris)
new_iris_model
```

```
Call:
lm(formula = Sepal.Length ~ Species, data = iris)

Coefficients:
      (Intercept)      Speciessetosa  Speciesversicolor
            6.588             -1.582             -0.652
```

Thursday, Jan 19

! TIL

Include a *very brief* summary of what you learnt in this class here.
Today, I learnt the following concepts in class:

1. Introduction to Multiple Linear Regression
2. Relationship between beta values and R-squared
3. Categorical Covariates in MLR

Provide more concrete details here, e.g.,

```r
# packages for today
library(plotly)
```

```
Attaching package: 'plotly'


The following object is masked from 'package:ggplot2':

    last_plot


The following object is masked from 'package:stats':

    filter


The following object is masked from 'package:graphics':

    layout
```

Multiple Linear Regression

We now have p covariates instead of 1: $X = \{x_1|x_2|...|x_p\}$ such that $y = \beta_0 + \beta_1 x_1 + ... + \beta_p x_p$ and the full description is $y_i = \beta_0 + \beta_1 x_{1i} + ... + \beta_p x_{pi} + \epsilon_i$

Look at the Credit dataset:

```r
attach(ISLR2::Credit)
df<-Credit%>%tibble()
df
```

```
# A tibble: 400 x 11
   Income Limit Rating Cards   Age Educat~1 Own   Student Married Region Balance
    <dbl> <dbl>  <dbl> <dbl> <dbl>    <dbl> <fct> <fct>   <fct>   <fct>    <dbl>
 1   14.9  3606    283     2    34       11 No    No      Yes     South      333
 2  106.   6645    483     3    82       15 Yes   Yes     Yes     West       903
 3  105.   7075    514     4    71       11 No    No      No      West       580
 4  149.   9504    681     3    36       11 Yes   No      No      West       964
 5   55.9  4897    357     2    68       16 No    No      Yes     South      331
 6   80.2  8047    569     4    77       10 No    No      No      South     1151
 7   21.0  3388    259     2    37       12 Yes   No      No      East       203
 8   71.4  7114    512     2    87        9 No    No      No      West       872
 9   15.1  3300    266     5    66       13 Yes   No      No      South      279
```

```
10   71.1  6819    491     3    41        19 Yes    Yes      Yes      East       1350
# ... with 390 more rows, and abbreviated variable name 1: Education
```

Focus on income, rating, and limit:

```
df3 <- df %>%select(Income,Limit,Rating)
df3
```

```
# A tibble: 400 x 3
   Income Limit Rating
    <dbl> <dbl>  <dbl>
 1   14.9  3606    283
 2  106.   6645    483
 3  105.   7075    514
 4  149.   9504    681
 5   55.9  4897    357
 6   80.2  8047    569
 7   21.0  3388    259
 8   71.4  7114    512
 9   15.1  3300    266
10   71.1  6819    491
# ... with 390 more rows
```

To see how credit limit relates to income and rating, use the following EDA:

```
# fig <- plot_ly(df3, x=~Income,y=~Rating,z=~Limit)
# fig%>%add_markers()
# these interactive plots break the pdf rendering so they will not be included in the pdf
```

And a model:

```
model<- lm(Limit~Income+Rating,df3)
model
```

```
Call:
lm(formula = Limit ~ Income + Rating, data = df3)

Coefficients:
(Intercept)       Income       Rating
  -532.4711       0.5573      14.7711
```

The model looks like a hyperplane when using 2 covariates:

```
# ranges <- df3 %>%
 # select(Income,Rating) %>%
#  colnames()%>%
 # map(\(x) seq(0.1*min(df3[x]),1.1*max(df3[x]),length.out=50))

#b<-model$coefficients
#z<-outer(
 # ranges[[1]],
  #ranges[[2]],
  #Vectorize(function(x2,x3) {
   # b[1]+b[2]*x2+b[3]*x3
    #})
#)
#fig%>%
 # add_surface(x=ranges[[1]],y=ranges[[2]],z=t(z),alpha=0.3)%>%
#  add_markers()
#once again, the interactive plots cannot be included in the pdf submission.
```

Interpretation:

- $\beta_0 = -532.47$ is the expected value of y when $income = 0$ and $rating = 0$
- If Rating is held constant and Income changes by 1 unit, the corresponding change in Limit is $\beta_1 = 0.553$ units
- If Income is held constant and Rating changes by 1 unit, the corresponding change in Limit is $\beta_2 = 14.77$ units

Significance:

```
summary(model)
```

```
Call:
lm(formula = Limit ~ Income + Rating, data = df3)

Residuals:
    Min      1Q  Median      3Q     Max 
-420.97 -121.77   14.97  126.72  485.48 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) -532.47115   24.17283 -22.028   <2e-16 ***
```

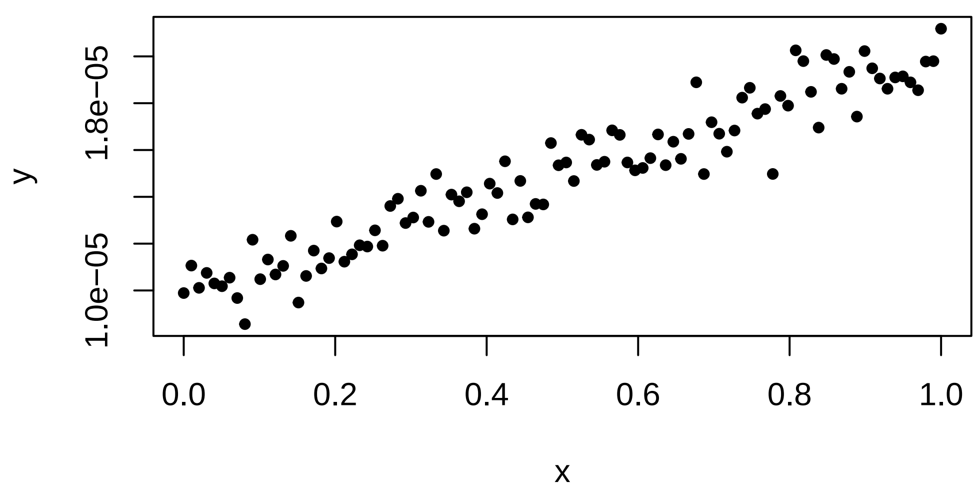
```
Income          0.55727    0.42349   1.316    0.189
Rating         14.77115    0.09647 153.124   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 182.3 on 397 degrees of freedom
Multiple R-squared:  0.9938,    Adjusted R-squared:  0.9938
F-statistic: 3.18e+04 on 2 and 397 DF,  p-value: < 2.2e-16
```

Clear case of multicolinearity, Income and Rating are related, so that is why Income shows up as completely insignificant.

Relating betas and R-squared:

```r
x<-seq(0,1,length.out=100)
b0<-0.00001
b1<-0.00001
y<-b0+b1*x+rnorm(100)*0.000001
plot(x,y,pch=20)
```



```r
modelex <- lm(y~x)
summary(modelex)
```

```

Call:
lm(formula = y ~ x)

Residuals:
       Min         1Q     Median         3Q        Max 
-2.834e-06 -6.108e-07 -3.994e-08  5.960e-07  2.136e-06 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 9.827e-06  1.899e-07   51.76   <2e-16 ***
x           1.026e-05  3.281e-07   31.29   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 9.565e-07 on 98 degrees of freedom
Multiple R-squared:  0.909, Adjusted R-squared:  0.9081 
F-statistic: 978.9 on 1 and 98 DF,  p-value: < 2.2e-16
```

You can have a significant p-value without a high R-squared, but not vice versa. To have a high R-squared, you *NEED* a significant p-value.

Multiple Regression with Categorical Covariates

Very similarly to simple linear regression, a categorical covariate changes the intercept.

```r
attach(Credit)
```

```
The following objects are masked from ISLR2::Credit:

    Age, Balance, Cards, Education, Income, Limit, Married, Own,
    Rating, Region, Student
```

```r
df<-Credit%>%tibble()

model <- lm(Limit~Rating+Married,df)
model
```

```
Call:
lm(formula = Limit ~ Rating + Married, data = df)

Coefficients:
(Intercept)       Rating   MarriedYes
    -528.09        14.87       -25.97
```

```r
ggplot(df)+
  geom_point(aes(x=Rating,y=Limit,color=Married))+
  geom_smooth(aes(x=Rating,y=Limit, fill=Married))
```

```
`geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```