Weekly Summary Template

Author Name

Table of contents

Tuesday, Feb 7

❗ TIL

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

1. Interpretation of regression coefficients
2. Categorical Covariates
3. Multiple Regression
    1. Extension from SLR
    2. Other Topics

Libraries Used

```
library(tidyverse)
```

```
-- Attaching packages --------------------------------------- tidyverse 1.3.2 --
v ggplot2 3.4.1     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   2.1.4     v forcats 1.0.0
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
```

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

```
Attaching package: 'kableExtra'

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

    group_rows
```

```
library(htmlwidgets)
```

What is the interpretation of β_0 and β_1?

The regression model is given as follows:

$$y_i = \beta_0 + \beta_1 * x_i + \epsilon_1$$

where:

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

Interpretations for the regression coefficients are that β_0 is the intercept and β_1 is the slope.

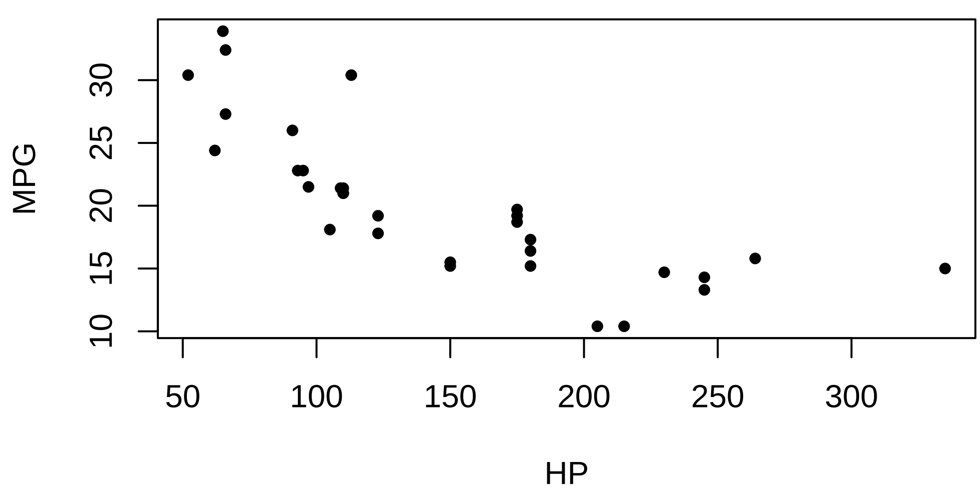
Lets consider the following example using 'mtcars'

```
library(ggplot2)

mtcars %>% head() %>% kable()
```

	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

Consider the following relationships:

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

plot(x, y, pch=20, xlab="HP", ylab="MPG")
```



```r
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 this means that:

> A "hypothetical" car with 'hp = 0' will have 'mpg = 30.09' = β_0

Its more instructive to consider the interpretation of the slope:

For example, if we have a covariate x_0 then the expected value for $y(x_0)$ is given by

$$y(x_0) = \beta_0 + \beta_1 x_0$$

What is the expected value for $x_0 + 1$

$y(x_0+1) = \beta_0 + \beta_1 \times (x_0+1) = \beta_0 + \beta_1 x_0 + \beta_1 = y(x_0) + \beta_1 above \implies \beta_1 = y(x_0+1) - y(x_0)$

Categorical Covariates

So far we looked at *simple* linear regression models where both x and y are quantitative.

Lets confirm that 'cyl' is indeed categorical:

```r
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
```

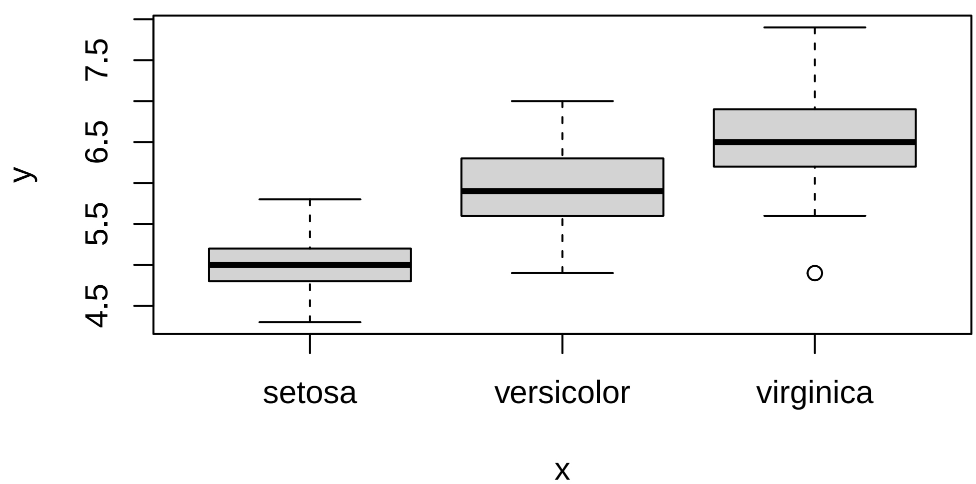
Another example we have is with the iris dataset:

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

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa

Example: We want to see if there is a relationship between 'species' and 'sepal.length'. How would we start the EDA?

```
y <- iris$Sepal.Length
x <- iris$Species
boxplot(y ~ x, iris)
```



Lets run a linear regression model and see what the model output is going to look like:

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

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

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

Even if x is categorical we can still write down the regression model as follows:

$$y_i = \beta_0 + \beta_1 * x_i$$

where $x_i \in \{setosa, versicolor, virginica\}$. This means that we end up with, (fundamentally) three different models

1. $y_i = \beta_0 + \beta_1 * (x_i ==)'setosa'$
2. $y_i = \beta_0 + \beta_1 * (x_i ==)'versicolor'$
3. $y_i = \beta_0 + \beta_1 * (x_i ==)'virginica'$

Now, the interpretation for the coefficients are as follows:

Intercept

β_0 is the expected y value when x belongs to th base category. This is what the intercept is capturing.

Slopes

β_1 with the name 'Species.versicolor' represents the following:

'(Intercept)' $= y(x =$ `setosa`$)$

'Species.versicolor' $= y(x =$ `versicolor`$) - y(x =$ `setosa`$)$

'Species.virginca' $= y(x =$ `virginca`$) - y(x =$ `setosa`$)$

Reordering the factors

Lets say that we didn't want 'setosa' to be the baseline level, and, instead, we wanted 'virginica' to be the baseline level. How would we do this?

First, we're going to reorder/relevel the categorical covariate

```
iris$Species # Before
```

```
  [1] setosa     setosa     setosa     setosa     setosa     setosa    
  [7] setosa     setosa     setosa     setosa     setosa     setosa    
 [13] setosa     setosa     setosa     setosa     setosa     setosa    
 [19] setosa     setosa     setosa     setosa     setosa     setosa    
 [25] setosa     setosa     setosa     setosa     setosa     setosa    
 [31] setosa     setosa     setosa     setosa     setosa     setosa    
 [37] setosa     setosa     setosa     setosa     setosa     setosa    
 [43] setosa     setosa     setosa     setosa     setosa     setosa    
 [49] setosa     setosa     versicolor versicolor versicolor versicolor
 [55] versicolor versicolor versicolor versicolor versicolor versicolor
 [61] versicolor versicolor versicolor versicolor versicolor versicolor
 [67] versicolor versicolor versicolor versicolor versicolor versicolor
 [73] versicolor versicolor versicolor versicolor versicolor versicolor
 [79] versicolor versicolor versicolor versicolor versicolor versicolor
 [85] versicolor versicolor versicolor versicolor versicolor versicolor
 [91] versicolor versicolor versicolor versicolor versicolor versicolor
 [97] versicolor versicolor versicolor versicolor virginica  virginica 
[103] virginica  virginica  virginica  virginica  virginica  virginica 
[109] virginica  virginica  virginica  virginica  virginica  virginica 
[115] virginica  virginica  virginica  virginica  virginica  virginica 
[121] virginica  virginica  virginica  virginica  virginica  virginica 
[127] virginica  virginica  virginica  virginica  virginica  virginica 
[133] virginica  virginica  virginica  virginica  virginica  virginica 
[139] virginica  virginica  virginica  virginica  virginica  virginica 
[145] virginica  virginica  virginica  virginica  virginica  virginica 
Levels: setosa versicolor virginica
```

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

```
iris$Species # After
```

```
  [1] setosa     setosa     setosa     setosa     setosa     setosa    
```

```
  [7] setosa     setosa     setosa     setosa     setosa     setosa    
 [13] setosa     setosa     setosa     setosa     setosa     setosa    
 [19] setosa     setosa     setosa     setosa     setosa     setosa    
 [25] setosa     setosa     setosa     setosa     setosa     setosa    
 [31] setosa     setosa     setosa     setosa     setosa     setosa    
 [37] setosa     setosa     setosa     setosa     setosa     setosa    
 [43] setosa     setosa     setosa     setosa     setosa     setosa    
 [49] setosa     setosa     versicolor versicolor versicolor versicolor
 [55] versicolor versicolor versicolor versicolor versicolor versicolor
 [61] versicolor versicolor versicolor versicolor versicolor versicolor
 [67] versicolor versicolor versicolor versicolor versicolor versicolor
 [73] versicolor versicolor versicolor versicolor versicolor versicolor
 [79] versicolor versicolor versicolor versicolor versicolor versicolor
 [85] versicolor versicolor versicolor versicolor versicolor versicolor
 [91] versicolor versicolor versicolor versicolor versicolor versicolor
 [97] versicolor versicolor versicolor versicolor virginica  virginica 
[103] virginica  virginica  virginica  virginica  virginica  virginica 
[109] virginica  virginica  virginica  virginica  virginica  virginica 
[115] virginica  virginica  virginica  virginica  virginica  virginica 
[121] virginica  virginica  virginica  virginica  virginica  virginica 
[127] virginica  virginica  virginica  virginica  virginica  virginica 
[133] virginica  virginica  virginica  virginica  virginica  virginica 
[139] virginica  virginica  virginica  virginica  virginica  virginica 
[145] virginica  virginica  virginica  virginica  virginica  virginica 
Levels: virginica setosa versicolor
```

Once we do the re-leveling, we can now run the regression model:

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

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

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

Thursday, Feb 9

> **!** TIL
>
> Include a *very brief* summary of what you learnt in this class here.
> Today, I learnt the following concepts in class:
>
> 1. Multiple Regression

Libraries

```
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 Regression

This is the extension of simple linear regression to multiple covariates $X = [x_1|x_2|\ldots|x_p]$, i.e.,

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots \beta_p x_p + \epsilon$$

In particular, the data looks like the following:

$\mathbf{y}$	$\mathbf{x}_1$	$\mathbf{x}_2$	...	$\mathbf{x}_p$
y_1	$x_{1,1}$	$x_{2,1}$	...	$x_{3,1}$

$\mathbf{y}$	$\mathbf{x}_1$	$\mathbf{x}_2$	$\cdots$	$\mathbf{x}_p$
y_2	$x_{1,2}$	$x_{2,2}$	$\cdots$	$x_{3,2}$
y_3	$x_{1,3}$	$x_{2,3}$	$\cdots$	$x_{3,3}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
y_n	$x_{1,n}$	$x_{2,n}$	$\cdots$	$x_{3,n}$

and, the full description of the model is as follows:

$$y_i = \beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i} + \cdots + \beta_p x_{p,i} + \epsilon$$

Consider the 'Credit' dataset:

```
library(ISLR2)
attach(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
```

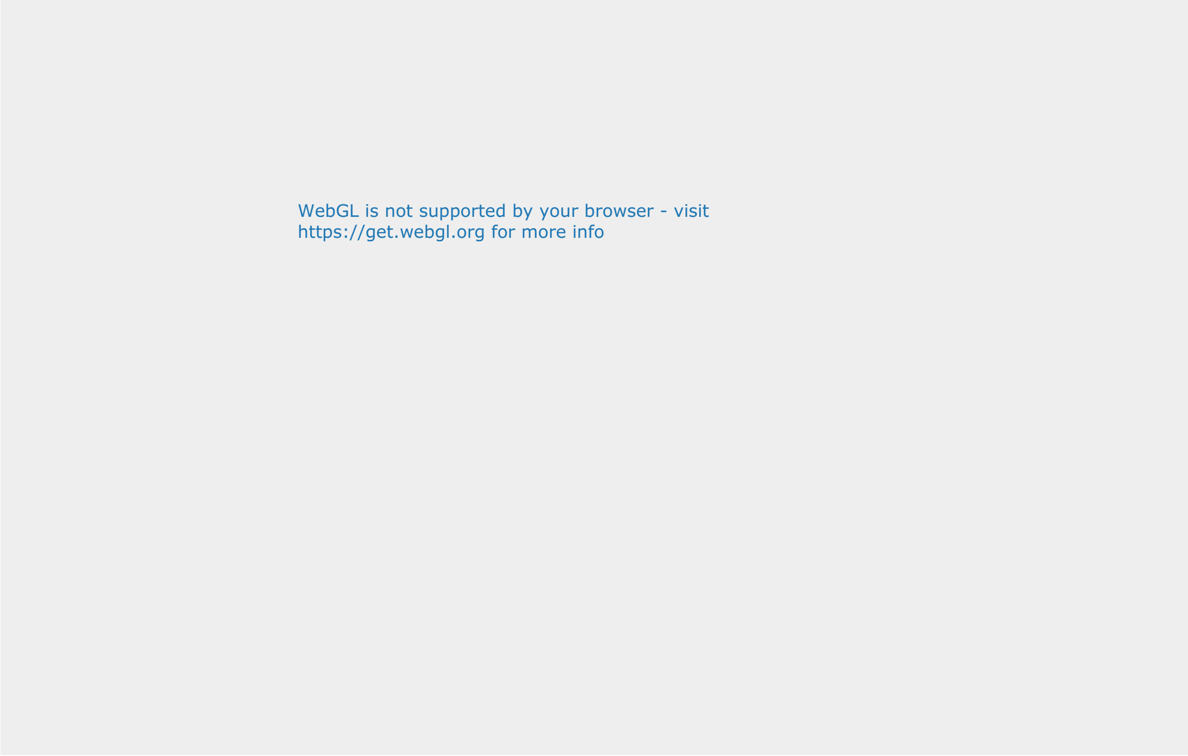
and, we'll look at the following three columns: 'income, rating, limit'

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

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

If we want to see how the credit limit is related too income and credit rating, we can visualize the following plot:

```
fig <- plot_ly(df3, x = ~Income, y = ~Rating, z = ~Limit)
fig %>% add_markers()
```



The regression model is as follows:

```
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
```

Q. What does the regression model look like here?

```
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()
```

WebGL is not supported by your browser - visit
https://get.webgl.org for more info

Q. What is the interpretation for the coefficients?

1. β_0 is the expected value of y when $income = 0$ and $rating = 0$
2. β_1 is saying that if $rating$ is held constant and $income$ changes by 1 unit, then the corresponding change in the 'limit' is 0.5573
3. β_2 is saying that if 'income' is held constant and 'rating' changes by 1 unit, then the corresponding change in 'limit' is 14.771.

Q. What about the significance?

```r
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
```