Module 1 - Peer reviewed

Outline:

In this homework assignment, there are four objectives.

1. To assess your knowledge of ANOVA/ANCOVA models
2. To apply your understanding of these models to a real-world datasets

General tips:

1. Read the questions carefully to understand what is being asked.
2. This work will be reviewed by another human, so make sure that you are clear and concise in what you are attempting to explain or answer.

Problem #1: Simulate ANCOVA Interactions

In this problem, we will work up to analyzing the following model to show how interaction terms work in an ANCOVA model.

$$Y_i = \beta_0 + \beta_1 x_i + \beta_2 z_i + \beta_3 x_i z_i + \varepsilon_i$$

This question is designed to enrich understanding of interactions in ANCOVA models. There is no additional coding required for this question, however we recommend messing around with the coefficents and plot as you see fit. Ultimately, this problem is graded based on written responses to questions asked in part **(a)** and **(b)**.

To demonstrate how interaction terms work in an ANCOVA model, let's generate some data. First, we consider the model

$$Y_i = \beta_0 + \beta_1 x_i + \beta_2 z_i + \varepsilon_i$$

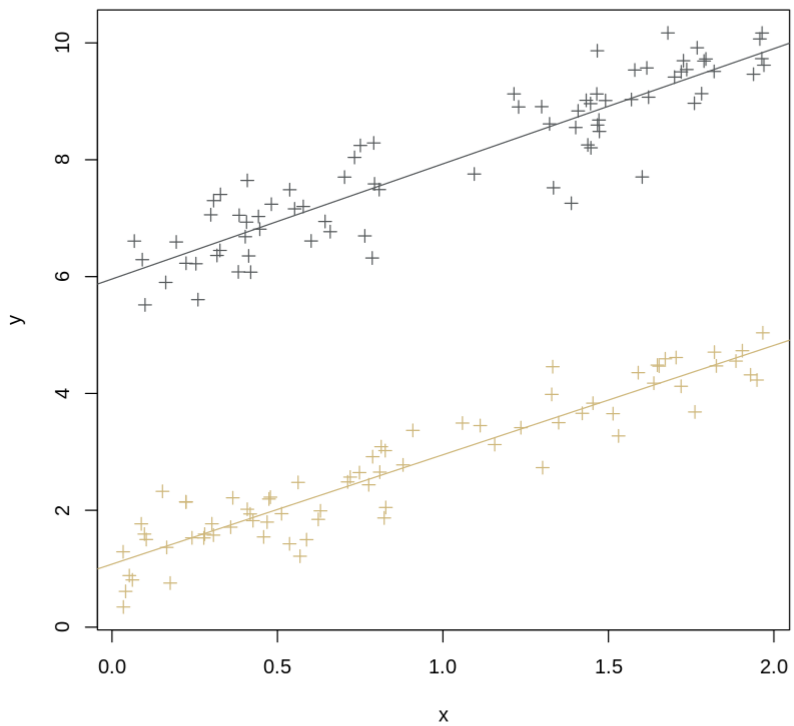
where X is a continuous covariate, Z is a dummy variable coding the levels of a two level factor, and $\varepsilon_i \overset{iid}{\sim} N(0, \sigma^2)$. We choose values for the parameters below (b0,...,b2).

In [1]:

```r
rm(list = ls())
set.seed(99)

#simulate data
n = 150
# choose these betas
b0 = 1; b1 = 2; b2 = 5; eps = rnorm(n, 0, 0.5);
x = runif(n,0,2); z = runif(n,-2,2);
z = ifelse(z > 0,1,0);
# create the model:
y = b0 + b1*x + b2*z + eps
df = data.frame(x = x,z = as.factor(z),y = y)
head(df)

#plot separate regression lines
with(df, plot(x,y, pch = 3, col = c("#CFB87C","#565A5C")[z]))
abline(coef(lm(y[z == 0] ~ x[z == 0], data = df)), col = "#CFB87C")
abline(coef(lm(y[z == 1] ~ x[z == 1], data = df)), col = "#565A5C")
```

A data.frame: 6 × 3

	x	z	y
	<dbl>	<fct>	<dbl>
1	0.09159879	1	6.290179
2	1.96439135	1	10.168612
3	0.57805656	1	7.200027
4	0.03370108	0	1.289331
5	1.82614045	0	4.470862
6	0.71220319	0	2.485743



1. (a) What happens with the slope and intercept of each of these lines?

In this case, we can think about having two separate regression lines--one for Y against x when the unit is in group $z = 0$ and another for Y against x when the unit is in group $z = 1$. What do we notice about the slope of each of these lines?

For the model above, the slopes of these lines will be exactly the same. This is because, when $z_i = 0$:

$$Y_i = \beta_0 + \beta_1 x_i + \varepsilon_i,$$

and when $z_i = 1$:

$$Y_i = \beta_0 + \beta_2 + \beta_1 x_i + \varepsilon_i.$$

Note that these lines have the same slope, but difference intercepts.

1. (b) Now, let's add the interaction term (let $\beta_3 = 3$). What happens to the slopes of each line now?

The model now is of the form:

$$Y_i = \beta_0 + \beta_1 x_i + \beta_2 z_i + \beta_3 x_i z_i + \varepsilon_i$$

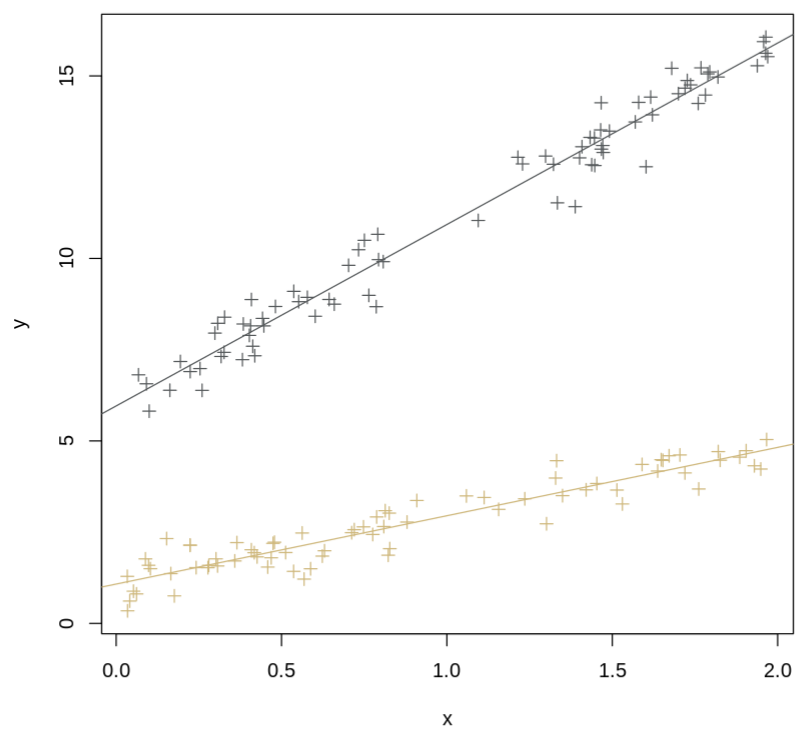
where x is a continuous covariate, z is a dummy variable coding the levels of a two level factor, and $\varepsilon_i \overset{iid}{\sim} N(0, \sigma^2)$. We choose values for the parameters below (b0,...,b3).

In [2]:

```r
#simulate data
set.seed(99)
n = 150
# pick the betas
b0 = 1; b1 = 2; b2 = 5; b3 = 3; eps = rnorm(n, 0, 0.5);

#create the model
y = b0 + b1*x + b2*z + b3*(x*z) + eps
df = data.frame(x = x,z = as.factor(z),y = y)
head(df)

lmod = lm(y ~ x + z, data = df)
lmodz0 = lm(y[z == 0] ~ x[z == 0], data = df)
lmodz1 = lm(y[z == 1] ~ x[z == 1], data = df)
# summary(lmod)
# summary(lmodz0)
# summary(lmodz1)

# lmodInt = lm(y ~ x + z + x*z, data = df)
# summary(lmodInt)

#plot separate regression lines
with(df, plot(x,y, pch = 3, col = c("#CFB87C","#565A5C")[z]))
abline(coef(lm(y[z == 0] ~ x[z == 0], data = df)), col = "#CFB87C")
abline(coef(lm(y[z == 1] ~ x[z == 1], data = df)), col = "#565A5C")
```

A data.frame: 6 × 3

	x	z	y
	<dbl>	<fct>	<dbl>
1	0.09159879	1	6.564975
2	1.96439135	1	16.061786
3	0.57805656	1	8.934197
4	0.03370108	0	1.289331
5	1.82614045	0	4.470862
6	0.71220319	0	2.485743



In this case, we can think about having two separate regression lines--one for Y against x when the unit is in group $z = 0$ and another for Y against x when the unit is in group $z = 1$. **What do you notice about the slope of each of these lines?**

Now, with this new model, the slopes are different (and so will the intercepts). Note that when $z_i = 0$:

$$Y_i = \beta_0 + \beta_1 x_i + \varepsilon_i,$$

but when $z_i = 1$:

$$Y_i = \beta_0 + \beta_1 x_i + \beta_2 + \beta_3 x_i + \varepsilon_i = (\beta_0 + \beta_2) + (\beta_1 + \beta_3) x_i + \varepsilon_i.$$

We can interpret β_2 as the mean change in the mean of Y when x is zero. We can interpret β_3 as the mean change in the mean change in y for a one-unit increase in x.

Problem #2

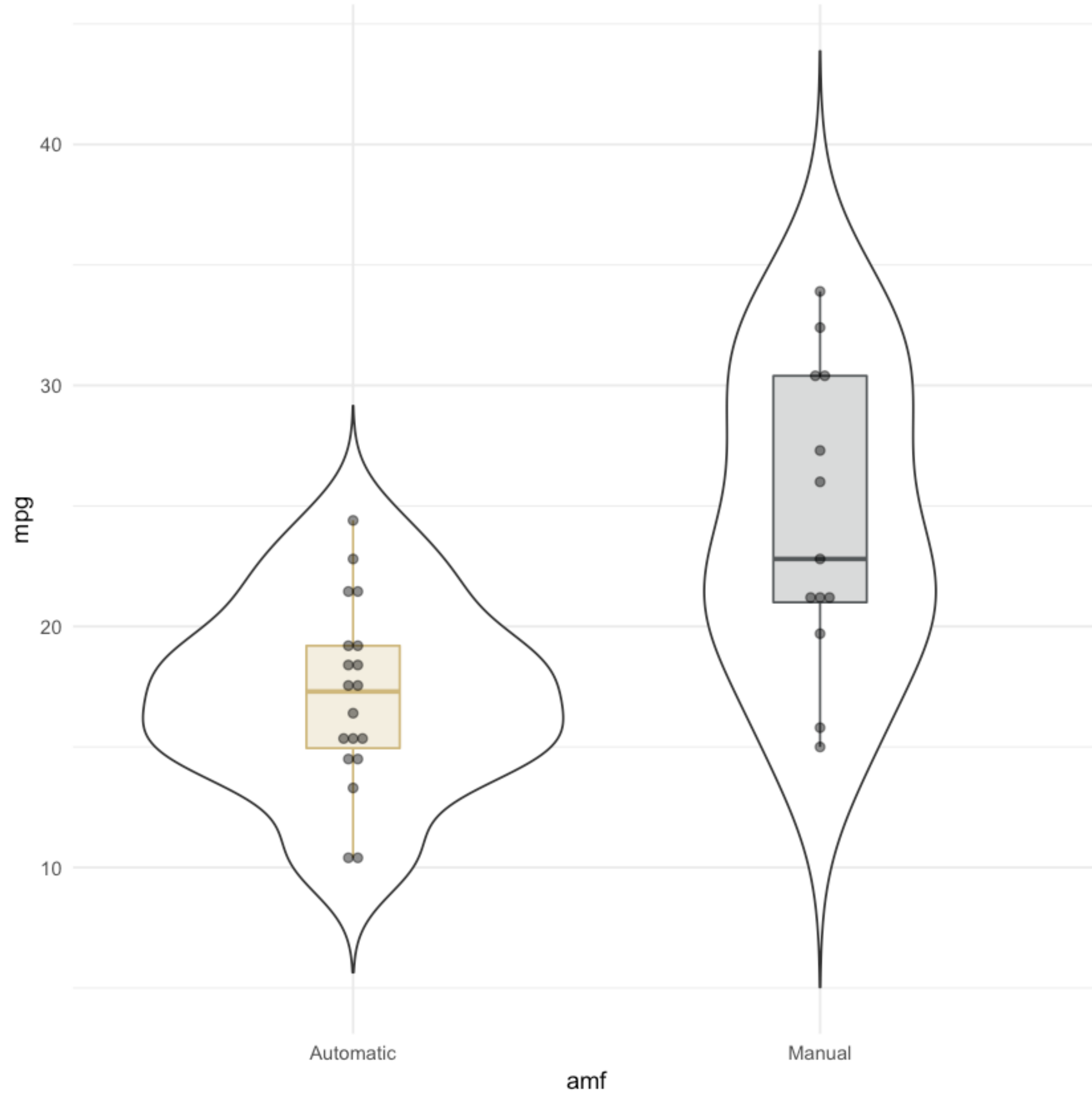
In this question, we ask you to analyze the `mtcars` dataset. The goal if this question will be to try to explain the variability in miles per gallon (`mpg`) using transmission type (`am`), while adjusting for horsepower (`hp`).

To load the data, use `data(mtcars)`

2. (a) Rename the levels of am from 0 and 1 to "Automatic" and "Manual" (one option for this is to use the revalue() function in the plyr package). Then, create a boxplot (or violin plot) of mpg against am. What do you notice? Comment on the plot.

In [4]:
```
library(tidyverse)
library(ggplot2)
library(plyr)
data(mtcars)

mtcars$amf = as.factor(mtcars$am)
mtcars$amf = with(mtcars, revalue(amf, c("0" = "Automatic", "1" = "
Manual")))
#with(mtcars, boxplot(mpg ~ amf), xlab = "Transmission Type", ylab
= "MPG")
summary(mtcars)

p = ggplot(mtcars, aes(x=amf, y=mpg));
p = p + geom_violin(trim=FALSE)#
p = p + geom_boxplot(width=0.2, col = c("#CFB87C", "#565A5C"), fill
= c("#CFB87C", "#565A5C"), alpha = 0.25)
p = p + theme_minimal()
p = p + geom_dotplot(binaxis='y', stackdir='center', dotsize=0.5, a
lpha = 0.5)
p
```

```
      mpg             cyl             disp             hp       
 Min.   :10.40   Min.   :4.000   Min.   : 71.1   Min.   : 52.0  
 1st Qu.:15.43   1st Qu.:4.000   1st Qu.:120.8   1st Qu.: 96.5  
 Median :19.20   Median :6.000   Median :196.3   Median :123.0  
 Mean   :20.09   Mean   :6.188   Mean   :230.7   Mean   :146.7  
 3rd Qu.:22.80   3rd Qu.:8.000   3rd Qu.:326.0   3rd Qu.:180.0  
 Max.   :33.90   Max.   :8.000   Max.   :472.0   Max.   :335.0  
      drat             wt             qsec             vs        
 Min.   :2.760   Min.   :1.513   Min.   :14.50   Min.   :0.0000  
 1st Qu.:3.080   1st Qu.:2.581   1st Qu.:16.89   1st Qu.:0.0000  
 Median :3.695   Median :3.325   Median :17.71   Median :0.0000  
 Mean   :3.597   Mean   :3.217   Mean   :17.85   Mean   :0.4375  
 3rd Qu.:3.920   3rd Qu.:3.610   3rd Qu.:18.90   3rd Qu.:1.0000  
 Max.   :4.930   Max.   :5.424   Max.   :22.90   Max.   :1.0000  
       am              gear            carb             amf    
 Min.   :0.0000   Min.   :3.000   Min.   :1.000   Automatic:19  
 1st Qu.:0.0000   1st Qu.:3.000   1st Qu.:2.000   Manual   :13  
 Median :0.0000   Median :4.000   Median :2.000                 
 Mean   :0.4062   Mean   :3.688   Mean   :2.812                 
 3rd Qu.:1.0000   3rd Qu.:4.000   3rd Qu.:4.000                 
 Max.   :1.0000   Max.   :5.000   Max.   :8.000                 

`stat_bindot()` using `bins = 30`. Pick better value with `binwidth
`.
```



The difference in the mean of mpg for cars in the Automatic group vs the Manual group is -7.2449 (i.e., manual is higher, on average).

2. (b) Calculate the mean difference in `mpg` for the Automatic group compared to the Manual group.

In [5]:
```
diff = with(mtcars, mean(mpg[amf == "Manual"]) - mean(mpg[amf == "A
utomatic"]))
diff
```

7.24493927125506

The difference in the mean of mpg for cars in the Automatic group vs the Manual group is -7.2449 (i.e., manual is higher, on average).

2. (c) Construct three models:

1. An ANOVA model that checks for differences in mean `mpg` across different transmission types.
2. An ANCOVA model that checks for differences in mean `mpg` across different transmission types, adjusting for horsepower.
3. An ANCOVA model that checks for differences in mean `mpg` across different transmission types, adjusting for horsepower and for interaction effects between horsepower and transmission type.

Using these three models, determine whether or not the interaction term between transmission type and horsepower is significant.

In [6]:

```
anov = lm(mpg ~ amf, data = mtcars)
summary(anov)

anc = lm(mpg ~ hp + amf, data = mtcars)
summary(anc)

ancInt = lm(mpg ~  hp + amf + amf:hp, data = mtcars)
summary(ancInt)
```

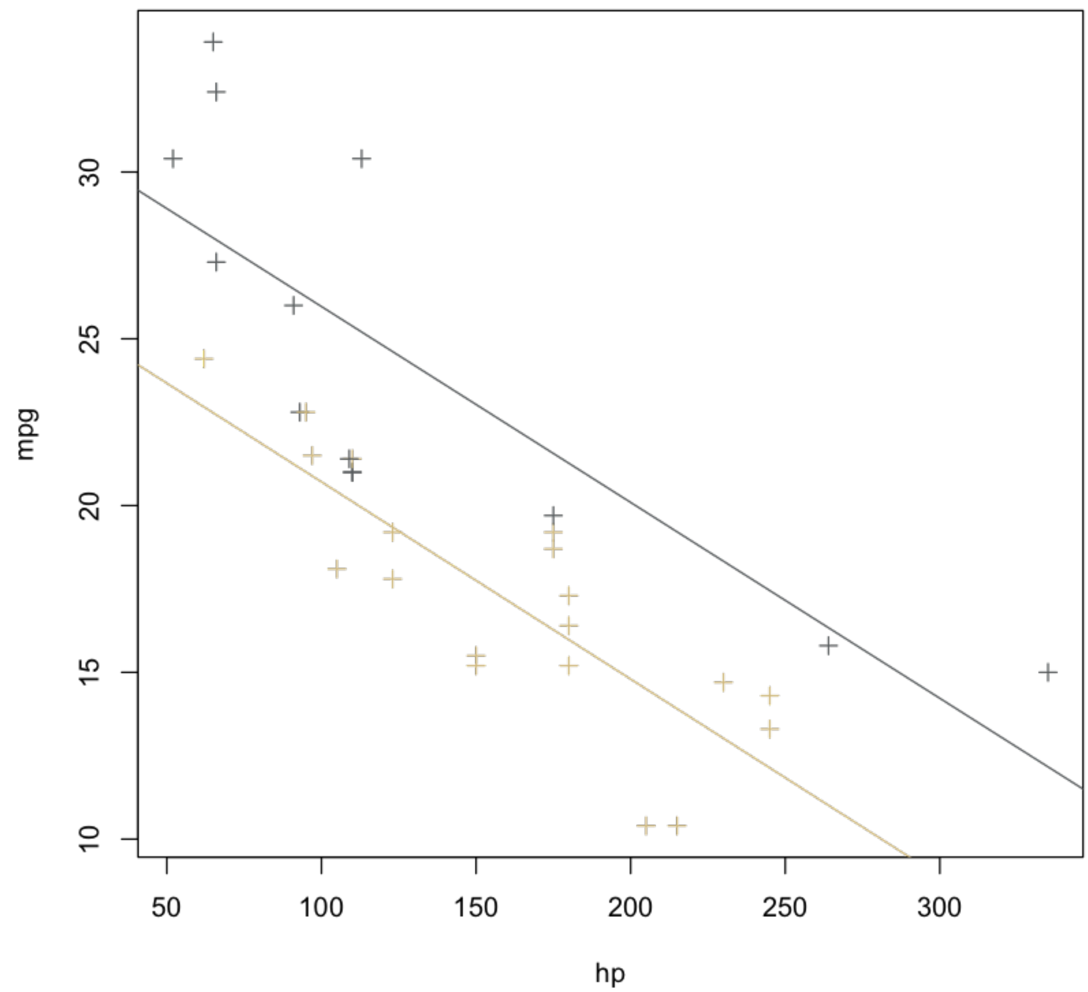
```
Call:
lm(formula = mpg ~ amf, data = mtcars)

Residuals:
    Min      1Q  Median      3Q     Max 
-9.3923 -3.0923 -0.2974  3.2439  9.5077 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   17.147      1.125  15.247 1.13e-15 ***
amfManual      7.245      1.764   4.106 0.000285 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.902 on 30 degrees of freedom
Multiple R-squared:  0.3598,	Adjusted R-squared:  0.3385 
F-statistic: 16.86 on 1 and 30 DF,  p-value: 0.000285

Call:
lm(formula = mpg ~ hp + amf, data = mtcars)

Residuals:
    Min      1Q  Median      3Q     Max 
-4.3843 -2.2642  0.1366  1.6968  5.8657 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 26.584914   1.425094  18.655  < 2e-16 ***
hp          -0.058888   0.007857  -7.495 2.92e-08 ***
amfManual    5.277085   1.079541   4.888 3.46e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.909 on 29 degrees of freedom
Multiple R-squared:  0.782,	Adjusted R-squared:  0.767 
F-statistic: 52.02 on 2 and 29 DF,  p-value: 2.55e-10

Call:
lm(formula = mpg ~ hp + amf + amf:hp, data = mtcars)

Residuals:
    Min      1Q  Median      3Q     Max 
-4.3818 -2.2696  0.1344  1.7058  5.8752 

Coefficients:
               Estimate Std. Error t value Pr(>|t|)    
(Intercept)  26.6248479  2.1829432  12.197 1.01e-12 ***
hp           -0.0591370  0.0129449  -4.568 9.02e-05 ***
amfManual     5.2176534  2.6650931   1.958   0.0603 .  
hp:amfManual  0.0004029  0.0164602   0.024   0.9806    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.961 on 28 degrees of freedom
Multiple R-squared:  0.782,	Adjusted R-squared:  0.7587 
F-statistic: 33.49 on 3 and 28 DF,  p-value: 2.112e-09
```

The p-value associated with the interaction parameter is large, providing no evidence of an interaction. So, we can use the ANCOVA model without an interaction.

2. (d) Construct a plot of mpg against horsepower, and color points based in transmission type. Then, overlay the regression lines with the interaction term, and the lines without. How are these lines consistent with your answer in (b)?

In [8]:
```
#plot separate regression lines


with(mtcars, plot(hp,mpg, pch = 3, col = c("#CFB87C","#565A5C")[am
f]))
abline(coef(lm(mpg[am == 0] ~ hp[am == 0], data = mtcars)), col = "
#CFB87C")
with(mtcars, abline(coef(ancInt)[1] + coef(ancInt)[3],
                    coef(ancInt)[2] + coef(ancInt)[4], col = "#565A
5C"))
coef(ancInt)[2] + coef(ancInt)[4]
```

hp: -0.0587340910946415



Each line has the sample plot!

In [ ]: