m1-peer-reviewed

March 12, 2023

1 Module 1 - Peer reviewed

1.0.1 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.

[1]:
```
# Load Required Packages
library(tidyverse)
library(ggplot2)
library(dplyr)
```

```
  Attaching packages                                         tidyverse
1.3.0

 ggplot2 3.3.0       purrr   0.3.4
 tibble  3.0.1       dplyr   0.8.5
 tidyr   1.0.2       stringr 1.4.0
 readr   1.3.1       forcats 0.5.0

  Conflicts
tidyverse_conflicts()
 dplyr::filter() masks stats::filter()
 dplyr::lag()    masks stats::lag()
```

1.0.2 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.

$$Y_i = \beta_0 + \beta_1 X + \beta_2 Z + \beta_3 XZ + \varepsilon_i$$

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

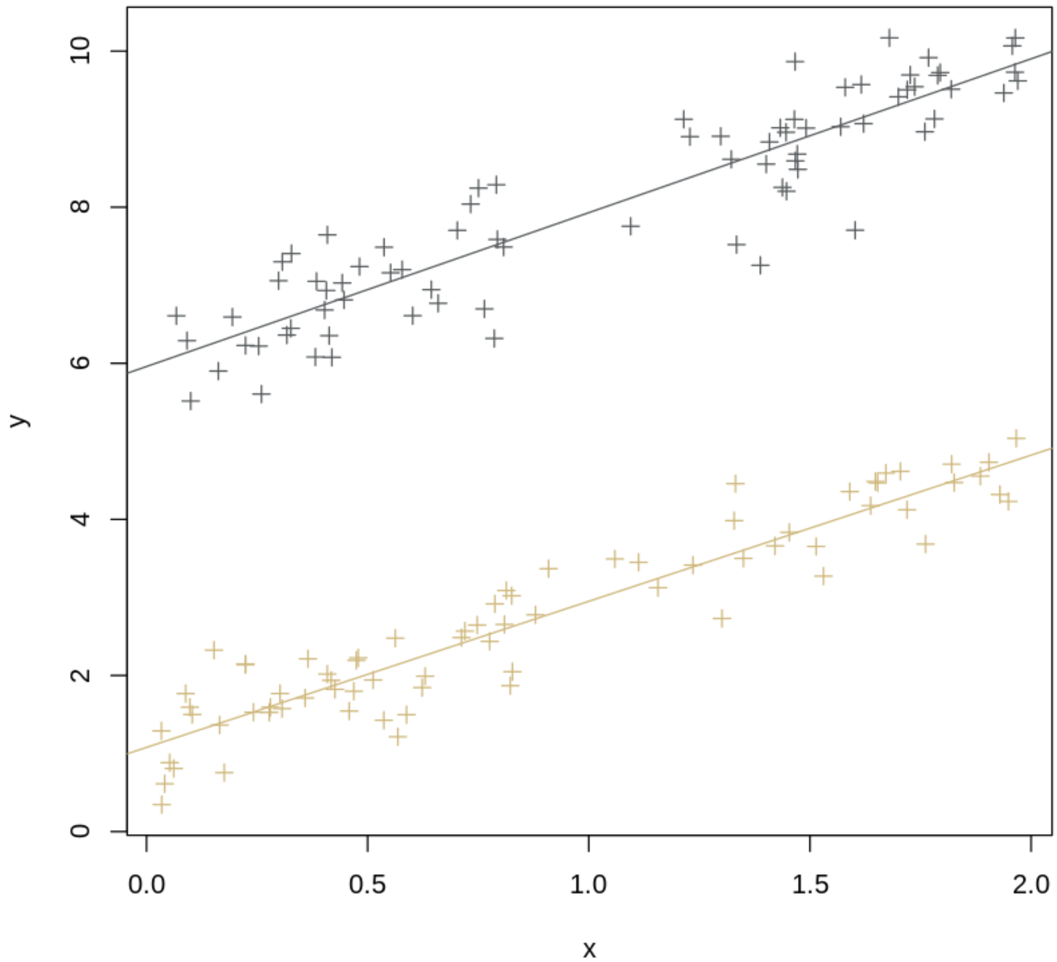
$$Y_i = \beta_0 + \beta_1 X + \beta_2 Z + \varepsilon_i$$

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).

[2]:
```
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



[3]:
```
lm(y ~ x + z ,data=df)
```

```
Call:
lm(formula = y ~ x + z, data = df)

Coefficients:
(Intercept)            x           z1
      1.035        1.923        4.974
```

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?

From regression coefficients, we can write down formulas for each regression model .Slope is same. $y_{0}= 1.035 + 1.923x $ $y_{1}= 1.035 + 4.974 + 1.923x $

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:

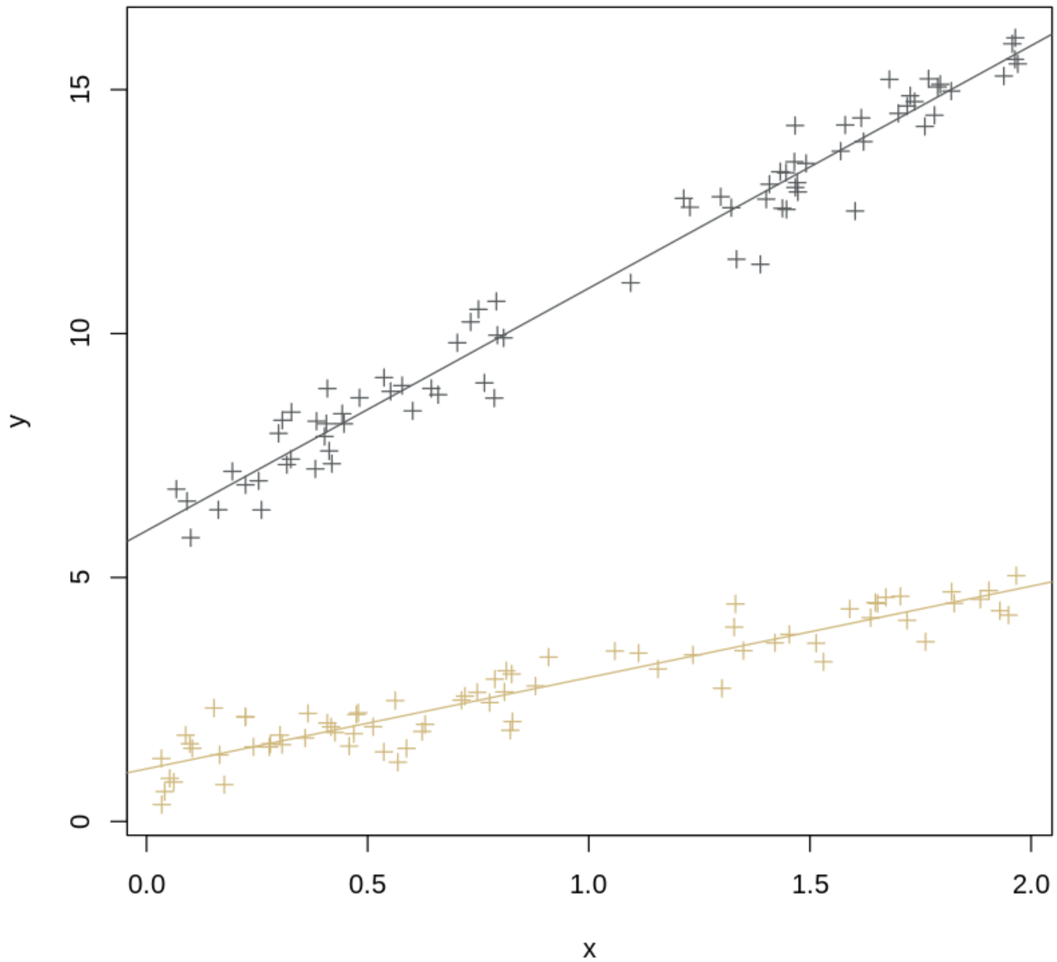
$$Y_i = \beta_0 + \beta_1 X + \beta_2 Z + \beta_3 XZ + \varepsilon_i$$

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).

[4]:
```
#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



[5]:
```
lm(y ~ x + z + x:z ,data=df)
```

```
Call:
lm(formula = y ~ x + z + x:z, data = df)

Coefficients:
(Intercept)            x           z1          x:z1
      1.079        1.872        4.880         3.099
```

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?**

From regression coefficients, we can write down formulas for each regression model.When we assumed interaction term in our regresion model, slope is different. $y_{0}= 1.079 + 1.872x $ $y_{1}= 1.079 + 4.880 + 1.872x + 3.099x $

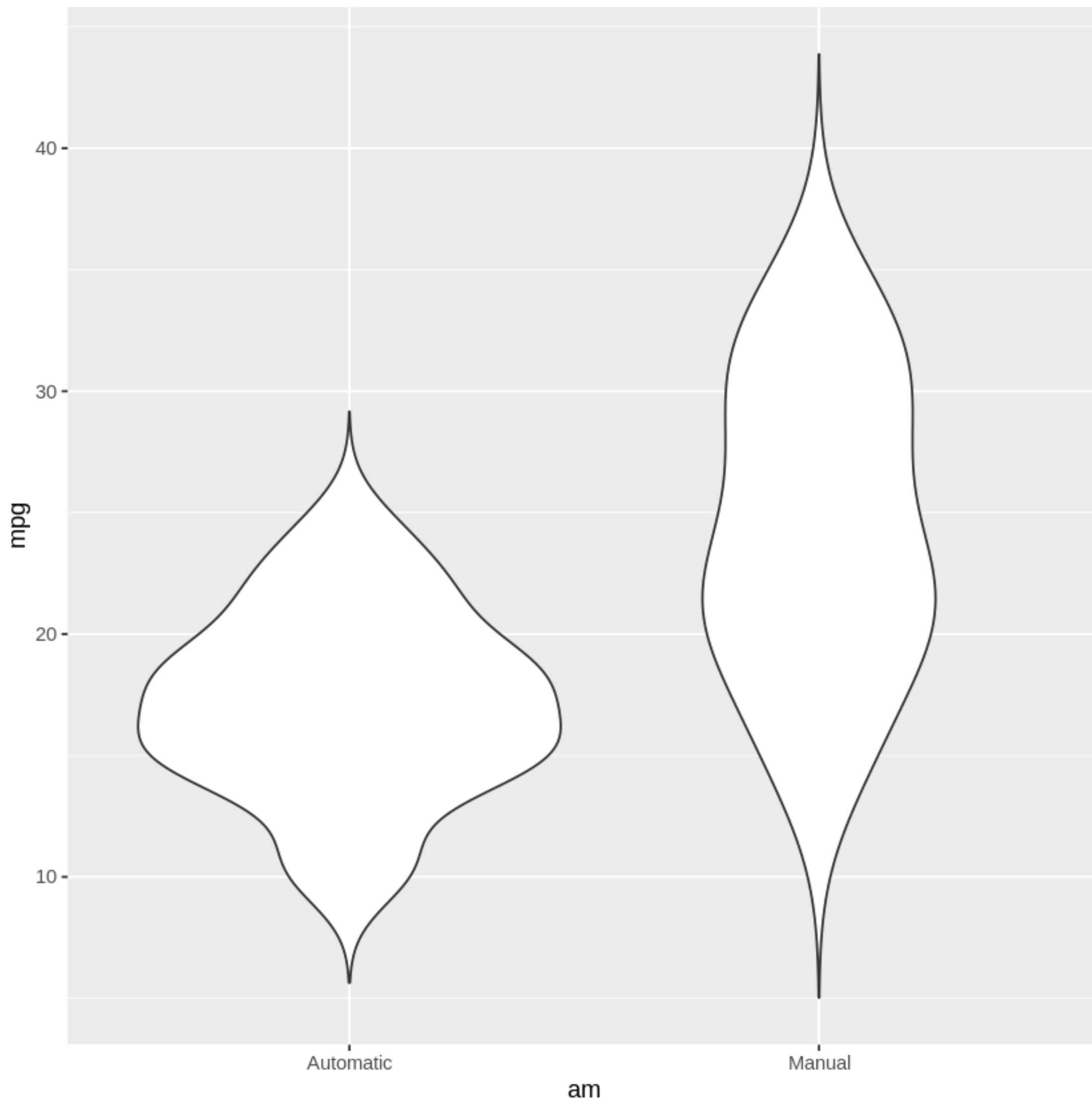
1.1 Problem #2

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

[28]:
```
data(mtcars)

# your code here
library(plyr)
mtcars$am <- revalue(as.character(mtcars$am),c('0' = "Automatic",'1'= "Manual"))

ggplot(data=mtcars,aes(x=am,y=mpg)) +
    geom_violin(trim=FALSE)
```



[59]:
```
summary(filter(mtcars,am=='Automatic')$mpg)
summary(filter(mtcars,am=='Manual')$mpg)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  10.40   14.95   17.30   17.15   19.20   24.40 


   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  15.00   21.00   22.80   24.39   30.40   33.90 
```

From violin plot, we can notice distribution of mpg is different by transmission type. From statistical summary, Automatic type's Mean mile/galon is lower than Manual type's Mean mile/galon.

** Statistical Summary ** * Automatic transmission type's mile/galon: * Min:10.4 * Median:17.3

* Mean:17.15 * Max:24.4 * Manual transmission type's mile/galon: * Min:15.0 * Median:22.80 * Mean:22.39 * Max:33.9

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

[48]:
```
# your code here
Y_mean <- mean(mtcars$mpg)
Y_mean_manual <- mean(filter(mtcars,am=='Manual')$mpg)
Y_mean_automatic <- mean(filter(mtcars,am=='Automatic')$mpg)

mean_difference <-(Y_mean_automatic - Y_mean)^2 - (Y_mean_manual - Y_mean)^2
mean_difference
```

-9.84171469578261

Mean difference in mpg for the Automatic group compared to the Manual group is about -9.84 (mpg).

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.

[34]:
```
# your code here
aov(mpg~am,data=mtcars)
aov(mpg~am + hp,data=mtcars)
aov(mpg~am + hp + hp:am,data=mtcars)
```

```
Call:
   aov(formula = mpg ~ am, data = mtcars)

Terms:
                      am Residuals
Sum of Squares  405.1506  720.8966
Deg. of Freedom        1        30

Residual standard error: 4.902029
Estimated effects may be unbalanced


Call:
```

```
   aov(formula = mpg ~ am + hp, data = mtcars)

Terms:
                      am       hp Residuals
Sum of Squares  405.1506 475.4573  245.4393
Deg. of Freedom        1        1        29

Residual standard error: 2.909196
Estimated effects may be unbalanced


Call:
   aov(formula = mpg ~ am + hp + hp:am, data = mtcars)

Terms:
                      am       hp    am:hp Residuals
Sum of Squares  405.1506 475.4573   0.0053  245.4340
Deg. of Freedom        1        1        1        28

Residual standard error: 2.960659
Estimated effects may be unbalanced
```

[56]:
```
anova(aov(mpg~am,data=mtcars))
```

A anova: 2 × 5

	Df <int>	Sum Sq <dbl>	Mean Sq <dbl>	F value <dbl>	Pr(>F) <dbl>
am	1	405.1506	405.15059	16.86028	0.0002850207
Residuals	30	720.8966	24.02989	NA	NA

[55]:
```
anova(aov(mpg~am,data=mtcars),aov(mpg~am + hp,data=mtcars))
```

A anova: 2 × 6

	Res.Df <dbl>	RSS <dbl>	Df <dbl>	Sum of Sq <dbl>	F <dbl>	Pr(>F) <dbl>
1	30	720.8966	NA	NA	NA	NA
2	29	245.4393	1	475.4573	56.17789	2.920375e-08

[54]:
```
anova(aov(mpg~am + hp,data=mtcars),aov(mpg~am + hp + hp:am,data=mtcars))
```

A anova: 2 × 6

	Res.Df <dbl>	RSS <dbl>	Df <dbl>	Sum of Sq <dbl>	F <dbl>	Pr(>F) <dbl>
1	29	245.4393	NA	NA	NA	NA
2	28	245.4340	1	0.005251456	0.0005991051	0.980646

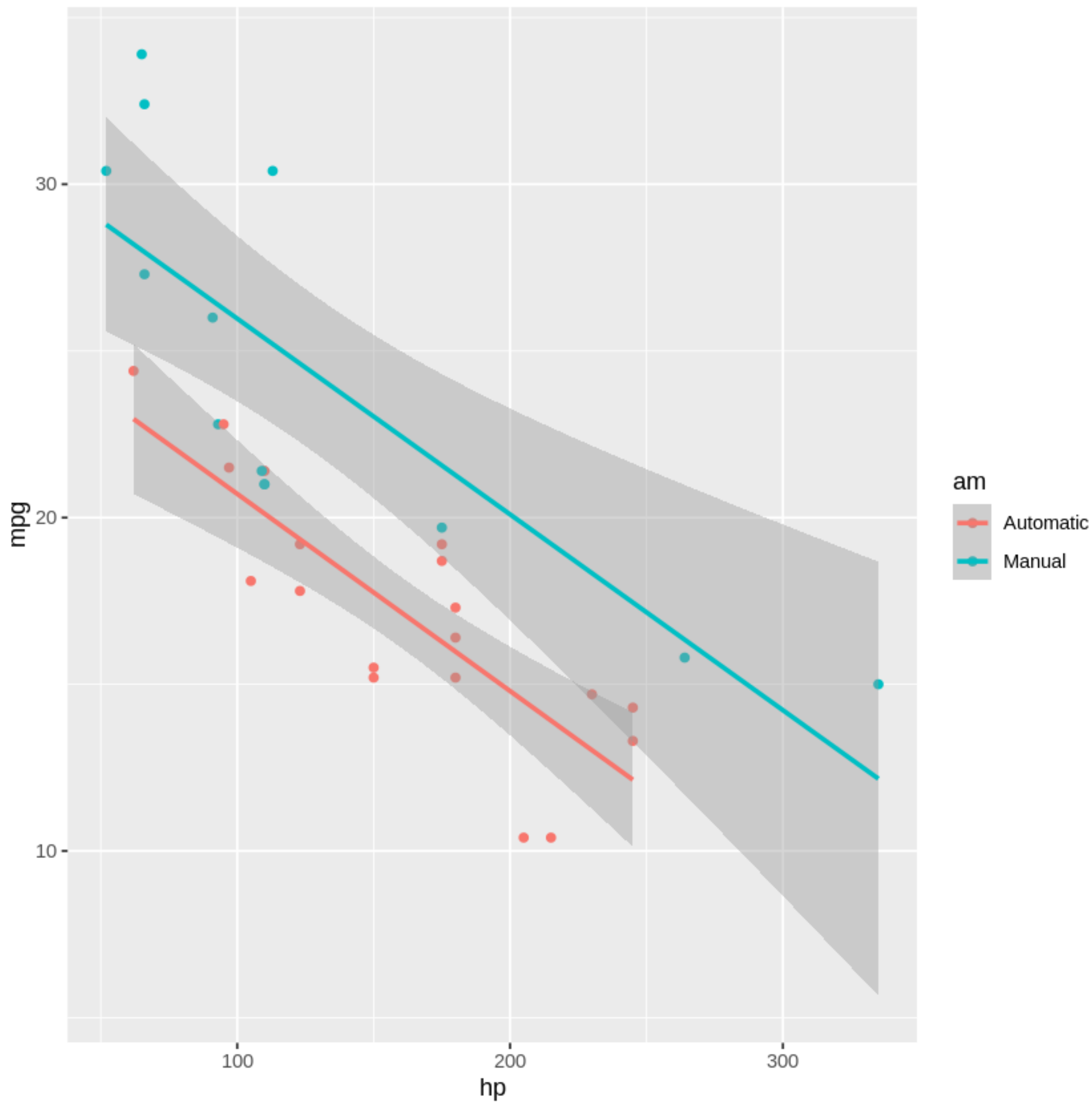
Using 3 anova/acova model, we perform ANOVA analysis for each additional term . We summarize p-value of each test with significant level(0.05).interaction term of horsepoer is not statistically significant in F-test.

- Baseline(Add am term):p-value < 0.05
- 2nd model(Add hp term):p-value < 0.05

- 3rd model(Add hp interaction term):p-value > 0.05

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) and (c)?

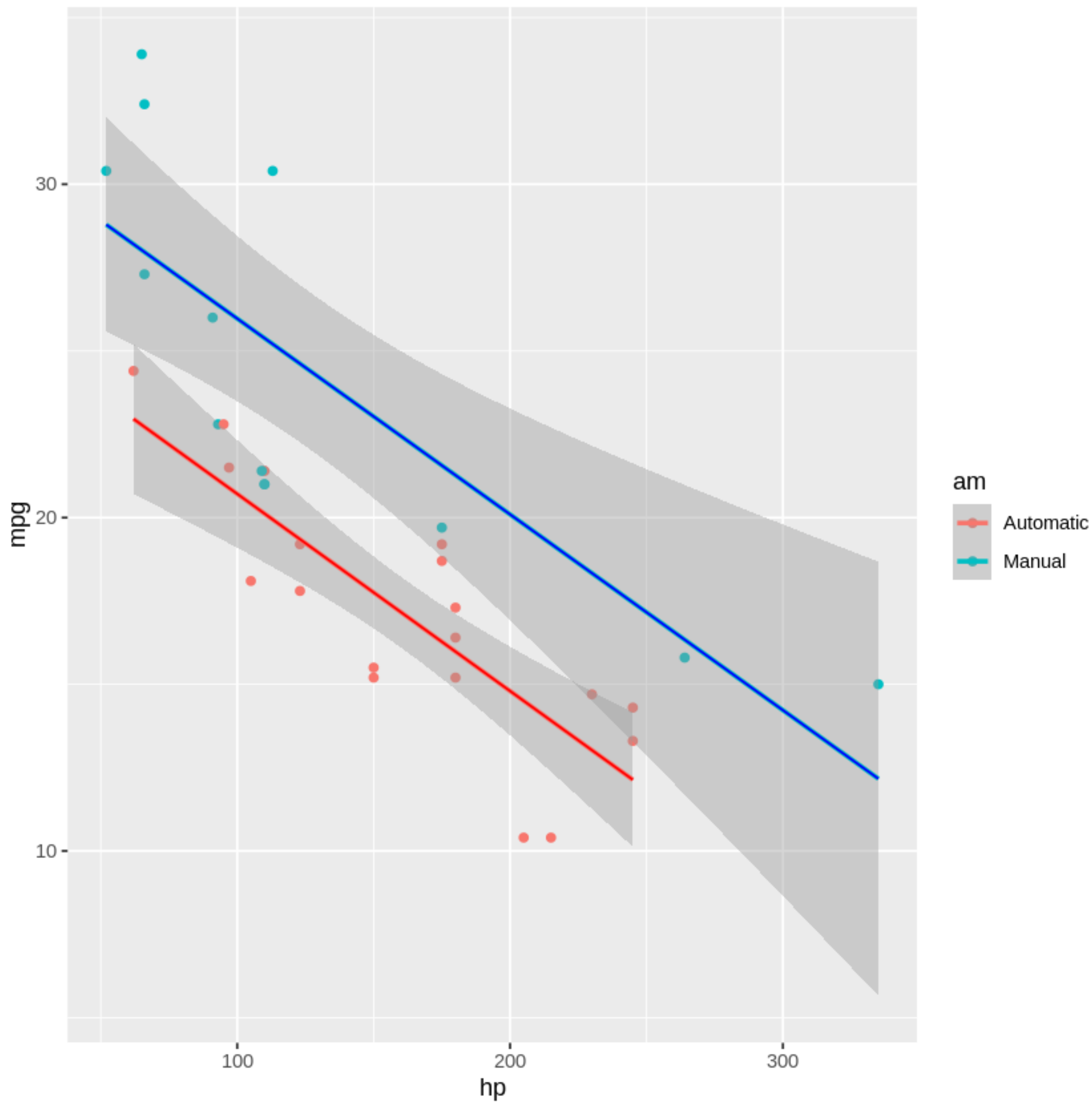
[86]:
```
# your code here
ggplot(data=mtcars,aes(x=hp,y=mpg,color=am)) +
    geom_point() +
    geom_smooth(formula=y~x,method='lm')
```



Construct a plot of mpg against horsepower, and color points based in transmission type. Then, overlay the regression lines withoutthe interaction term.

Then, I overlayed regression line with interaction term.

[105]:
```
lm.interact <- lm(mpg ~ hp*am,mtcars)
mtcars$yhat <- predict(lm.interact)

ggplot(data=mtcars,aes(x=hp,y=mpg,color=am)) +
    geom_point() +
    geom_smooth(formula=y~x,method='lm') +
  ␣
 ↪geom_line(aes(x=hp,y=yhat,color=am),filter(mtcars,am=='Manual'),colour=('blue'))+
  ␣
 ↪geom_line(aes(x=hp,y=yhat,color=am),filter(mtcars,am=='Automatic'),colour=('red'))
```



From above plot, interraction term is not significant so there is consistance question(b,c).

```
[ ]:
```