m1-peer-reviewed

February 25, 2025

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.

[15]:
```
# Load Required Packages
library(tidyverse)
library(ggplot2)
library(dplyr)
```

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

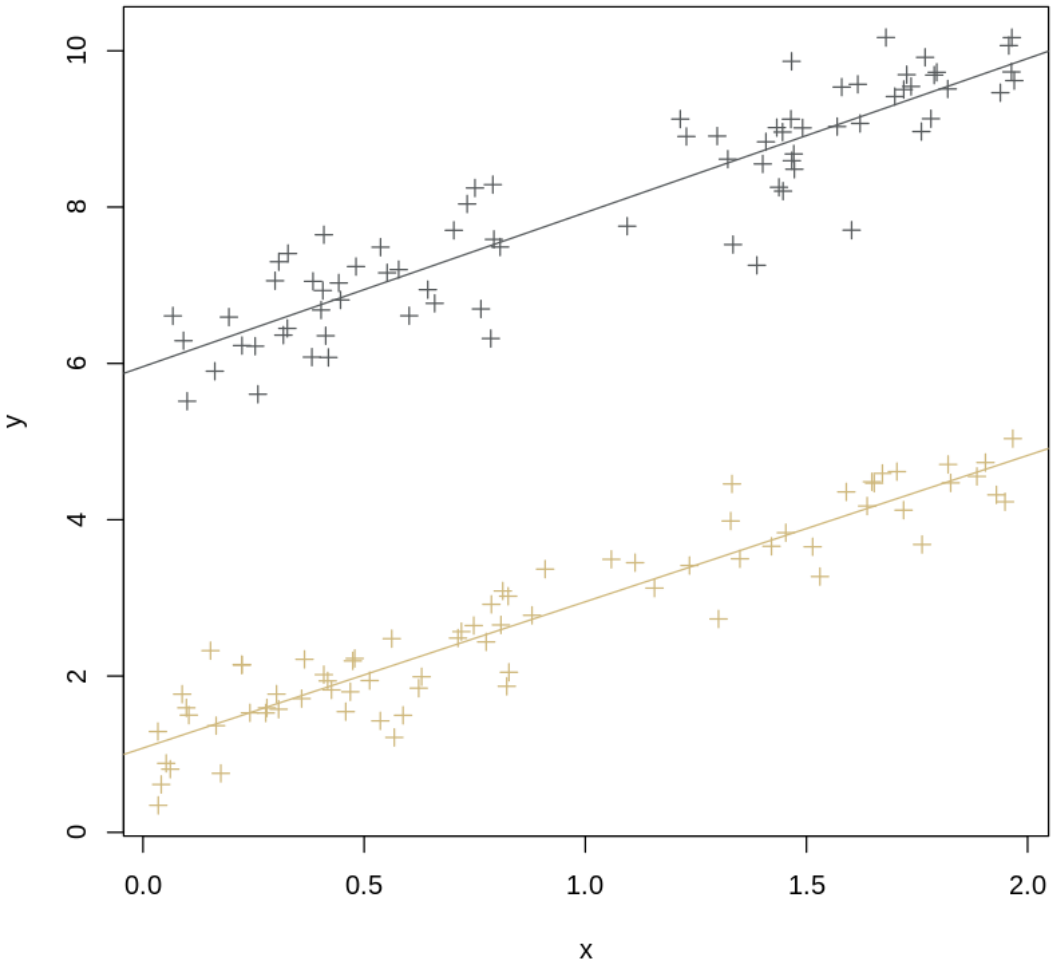
$$Y_i = \beta_0 + \beta_1 X + \beta_2 Z + \varepsilon_i$$

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).

[16]:
```
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 <dbl>	z <fct>	y <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?

[25]:
```
print("The intercepts are different but the slopes are the same for Z=0 and Z=1.
↪")
```

```
[1] "The intercepts are different but the slopes are the same for Z=0 and Z=1."
```

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:

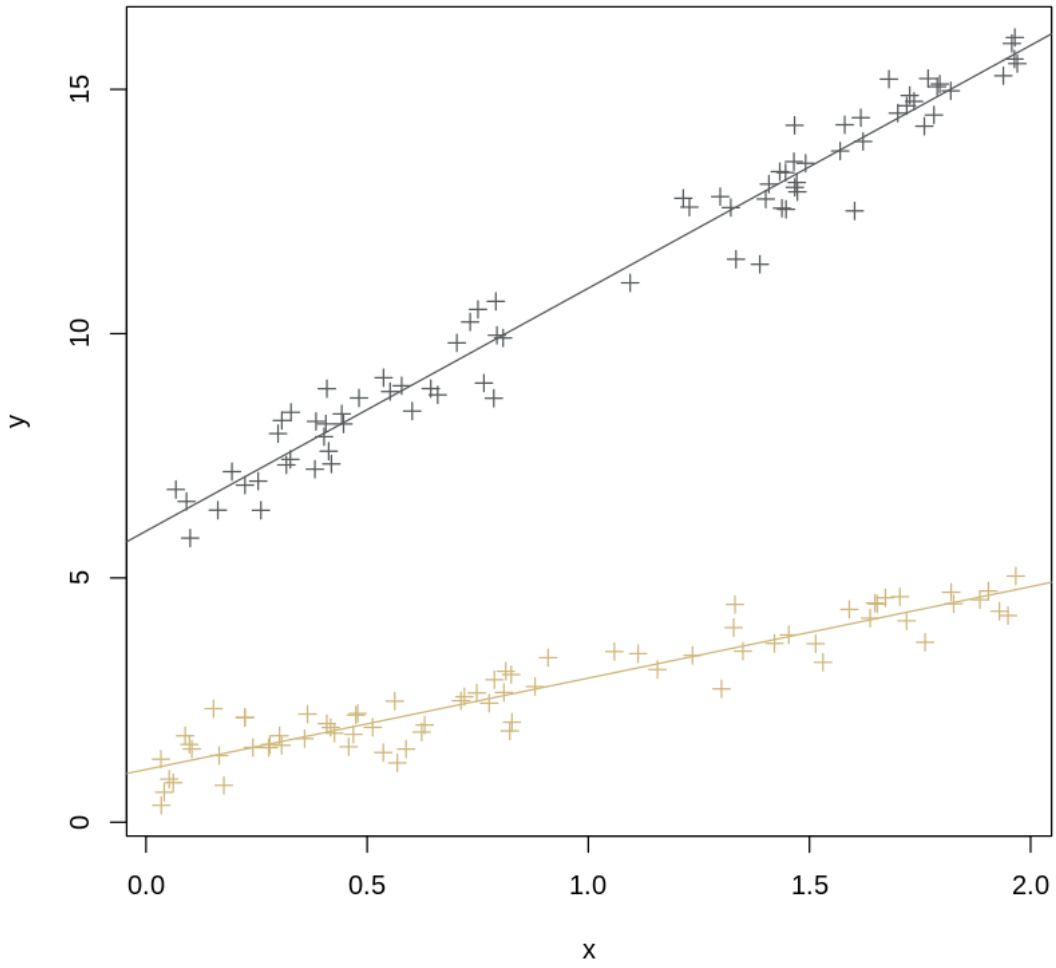
$$Y_i = \beta_0 + \beta_1 X + \beta_2 Z + \beta_3 XZ + \varepsilon_i$$

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).

[18]:
```
#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 <dbl>	z <fct>	y <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?**

[26]:
```
print("The slopes change for Z=1 because of the interaction term.  There are␣
↪different slopes now.")
```

```
[1] "The slopes change for Z=1 because of the interaction term.  There are
different slopes now."
```

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

[38]:
```
library(plyr)
library(dplyr)

data(mtcars)
mtcars$am = revalue(as.factor(mtcars$am), c("0" = "Automatic", "1" = "Manual"))


ggplot(mtcars, aes(x = am, y = mpg)) +
  geom_violin() +
  labs(title = "Boxplot of MPG by Transmission Type",
       x = "Transmission Type",
       y = "Miles Per Gallon (MPG)")

print("There are no outliers on the violin plot.  The median seems to be higher␣
 ↪on cars with manual transmission and the spread on the automatic seems to be␣
 ↪wider than that of the manual transmissions.")

# your code here
```

```
[1] "There are no outliers on the violin plot.  The median seems to be higher on
cars with manual transmission and the spread on the automatic seems to be wider
than that of the manual transmissions."
```

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

[50]:
```r
levels(mtcars$am) = c("Automatic", "Manual")

automatic_mpg <- mean(mtcars$mpg[mtcars$am == "Automatic"])
manual_mpg <- mean(mtcars$mpg[mtcars$am == "Manual"])

mean_difference <- automatic_mpg - manual_mpg
mean_difference


# your code here
```

-7.24493927125506

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.

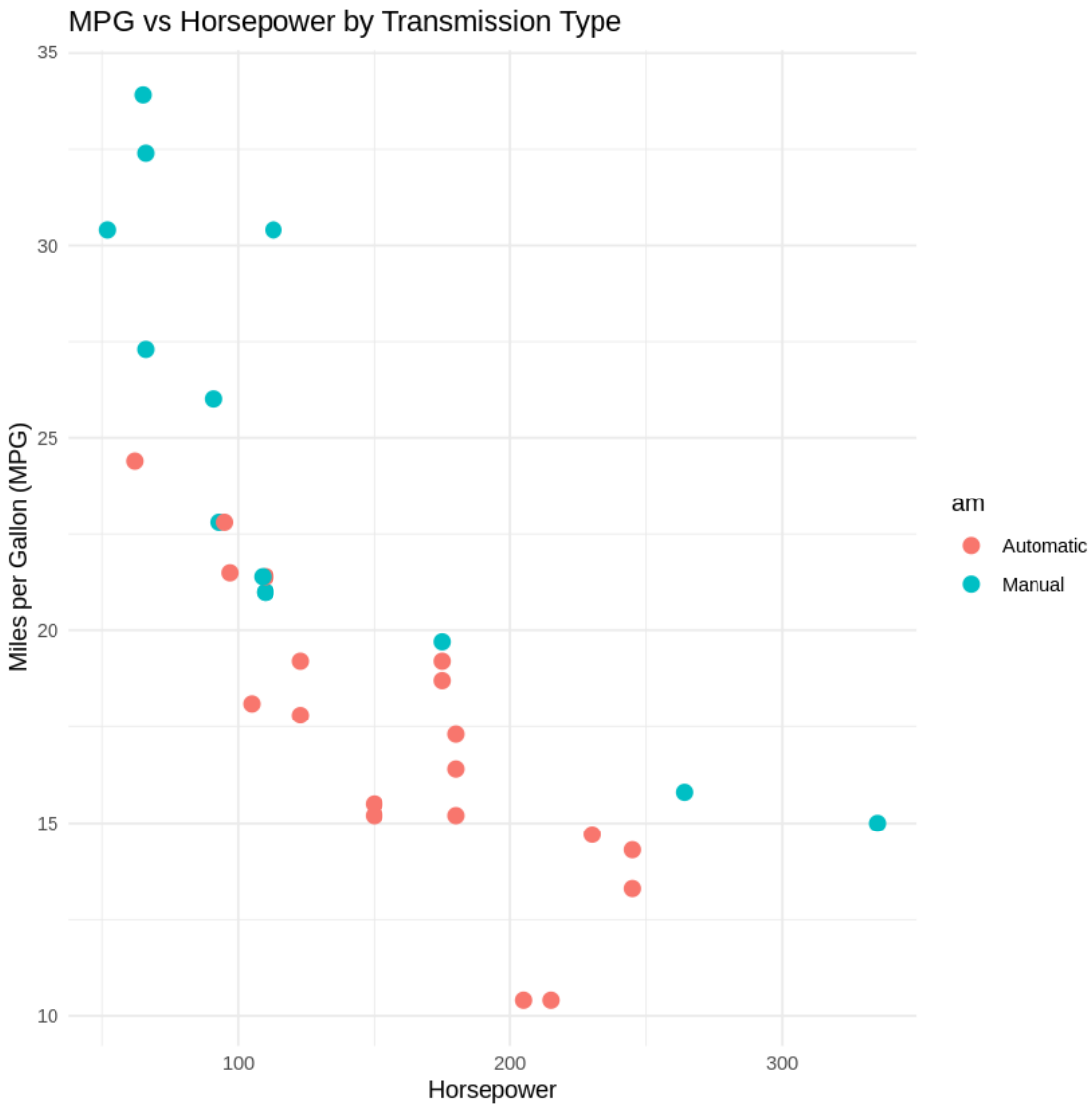
[51]:
```
anova_model <- lm(mpg ~ am, data = mtcars)
ancova_model.1 <- lm(mpg ~ am + hp, data = mtcars)
ancova_model.2 <- lm(mpg ~ am * hp, data = mtcars)

anova_model
ancova_model.1
ancova_model.2

# your code here
```

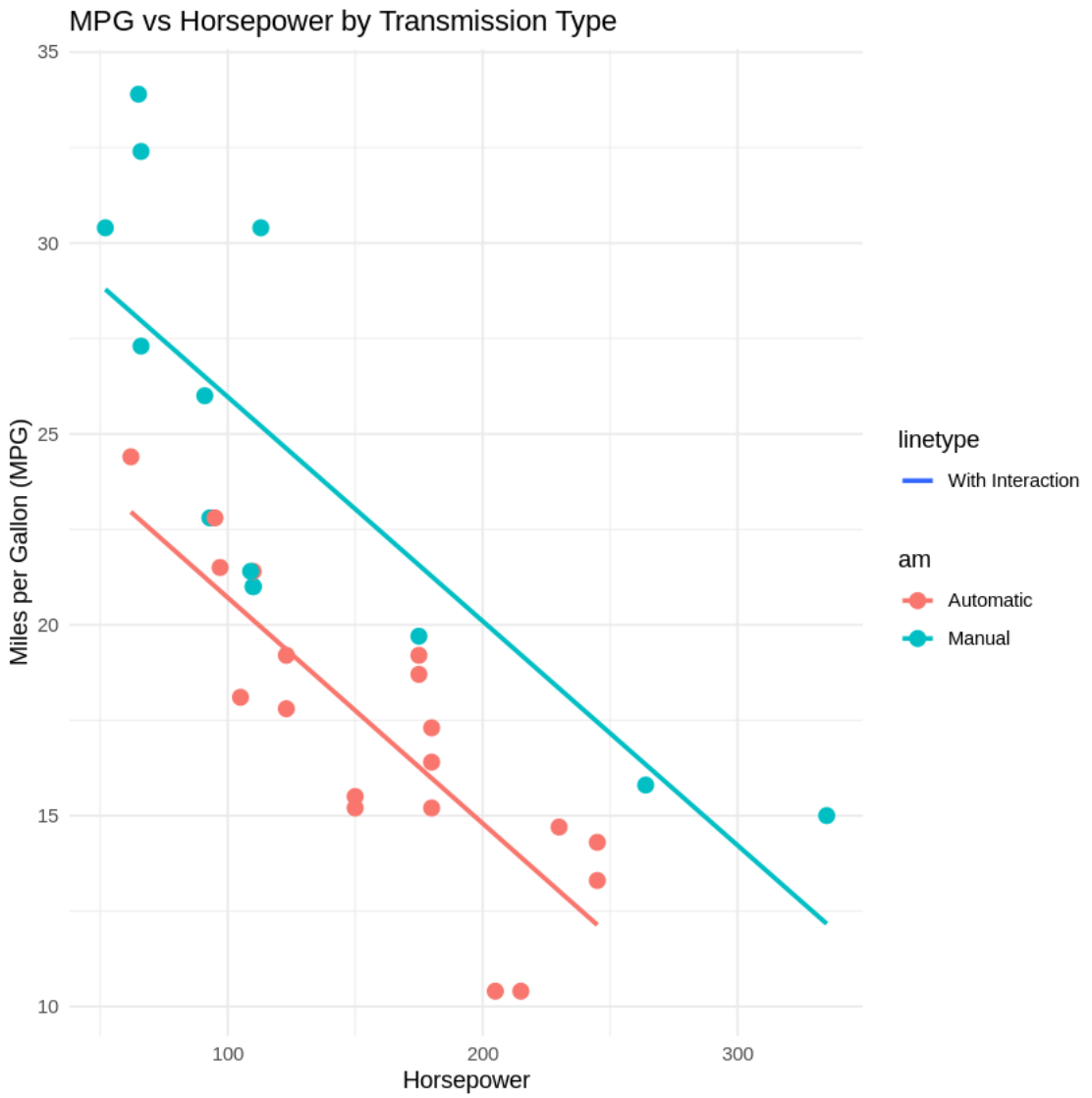
```
Call:
lm(formula = mpg ~ am, data = mtcars)

Coefficients:
(Intercept)      amManual
     17.147         7.245



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

Coefficients:
(Intercept)      amManual           hp
   26.58491       5.27709     -0.05889



Call:
lm(formula = mpg ~ am * hp, data = mtcars)

Coefficients:
(Intercept)      amManual           hp  amManual:hp
 26.6248479    5.2176534   -0.0591370    0.0004029
```

The interaction is not significant.

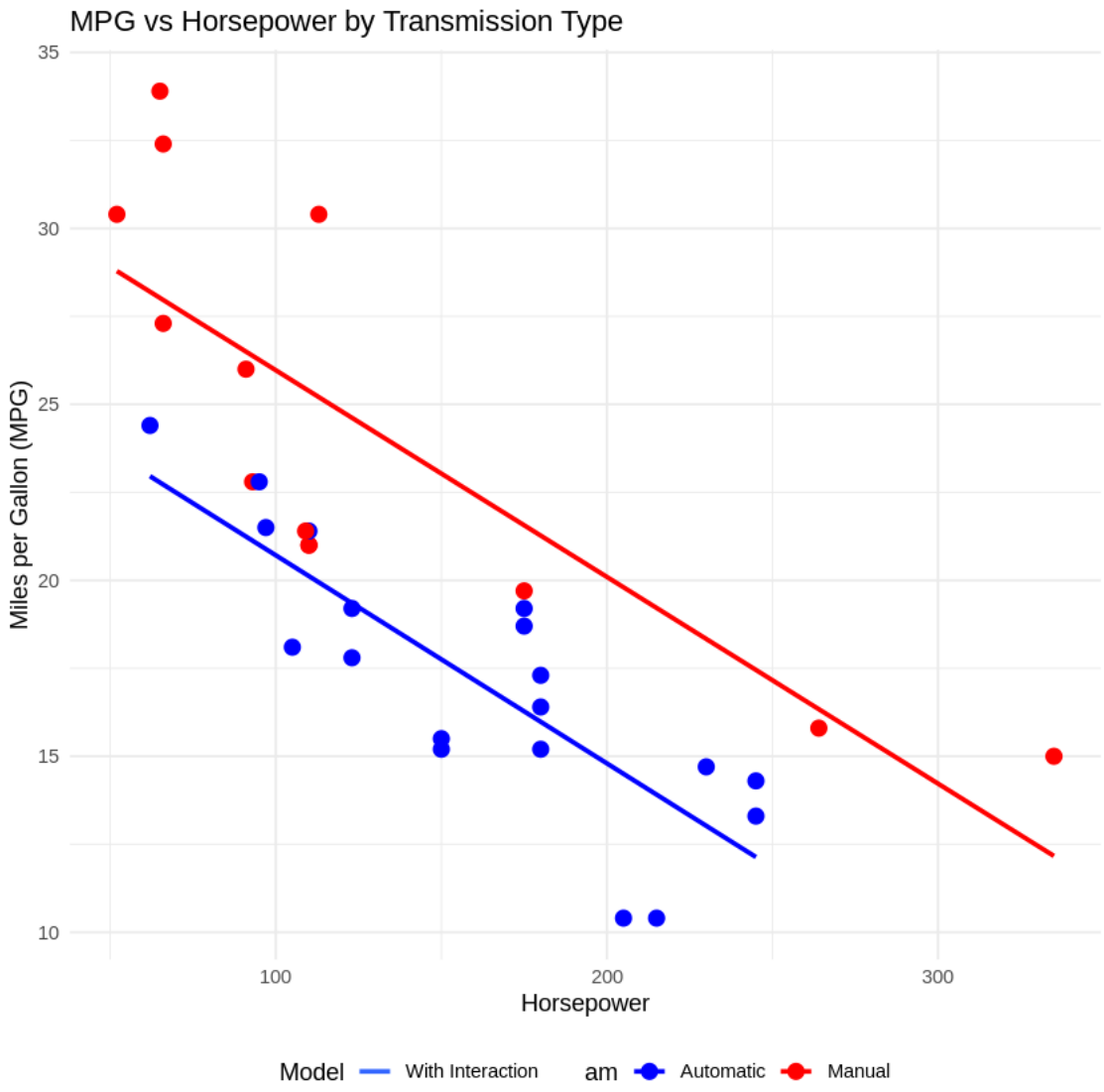
2. (d) Construct a plot of mpg against horsepower, and color points based in transmission type. Then, overlay the regression lines with the interaction term, and the lines without. How are these lines consistent with your answer in (b) and (c)?

[63]:
```
p <- ggplot(mtcars, aes(x = hp, y = mpg, color = am)) +
  geom_point(size = 3) +
  labs(title = "MPG vs Horsepower by Transmission Type",
       x = "Horsepower",
       y = "Miles per Gallon (MPG)") +
  theme_minimal()

p2 <- p +
  geom_smooth(method = "lm", formula = y ~ x, se = FALSE, aes(color = am,␣
 ↪linetype = "Without Interaction"), data = mtcars)

mtcars_interaction <- cbind(mtcars, predict(ancova_model.2, interval =␣
 ↪"confidence"))

p2 <- p +
  geom_smooth(aes(y = fit, color = am, linetype = "With Interaction"), data =␣
 ↪mtcars_interaction, method = "lm", formula = y ~ x, se = FALSE)


p3 <- p2 +
  scale_color_manual(values = c("Automatic" = "blue", "Manual" = "red")) +
  scale_linetype_manual(name = "Model", values = c("Without Interaction" =␣
 ↪"dashed", "With Interaction" = "solid")) +
  theme(legend.position = "bottom")

p
p2
p3

print("The slopes for each transmission type show that the effect of horsepower␣
 ↪on mpg is different between Automatic and Manual transmissions.")

# your code here
```


MPG vs Horsepower by Transmission Type
Miles per Gallon (MPG)
35
30
25
20
15
10
100
200
300
Horsepower
am
Automatic
Manual




```
[1] "The slopes for each transmission type show that the effect of horsepower on
mpg is different between Automatic and Manual transmissions."
```


MPG vs Horsepower by Transmission Type
Miles per Gallon (MPG)
35
30
25
20
15
10
100
200
300
Horsepower
Model
With Interaction
am
Automatic
Manual