STA_445_HW5

Bianca L.

2023-10-03

```
library(readxl)
library(tidyverse)
library(faraway)
library(ggrepel)
```

Problem 1: Chapter 14 Problem 1

The `infmort` data set from the package `faraway` gives the infant mortality rate for a variety of countries. The information is relatively out of date, but will be fun to graph. Visualize the data using by creating scatter plots of mortality vs income while faceting using `region` and setting color by `oil` export status. Utilize a $\log_{10}$ transformation for both `mortality` and `income` axes. This can be done either by doing the transformation inside the `aes()` command or by utilizing the `scale_x_log10()` or `scale_y_log10()` layers. The critical difference is if the scales are on the original vs log transformed scale. Experiment with both and see which you prefer.

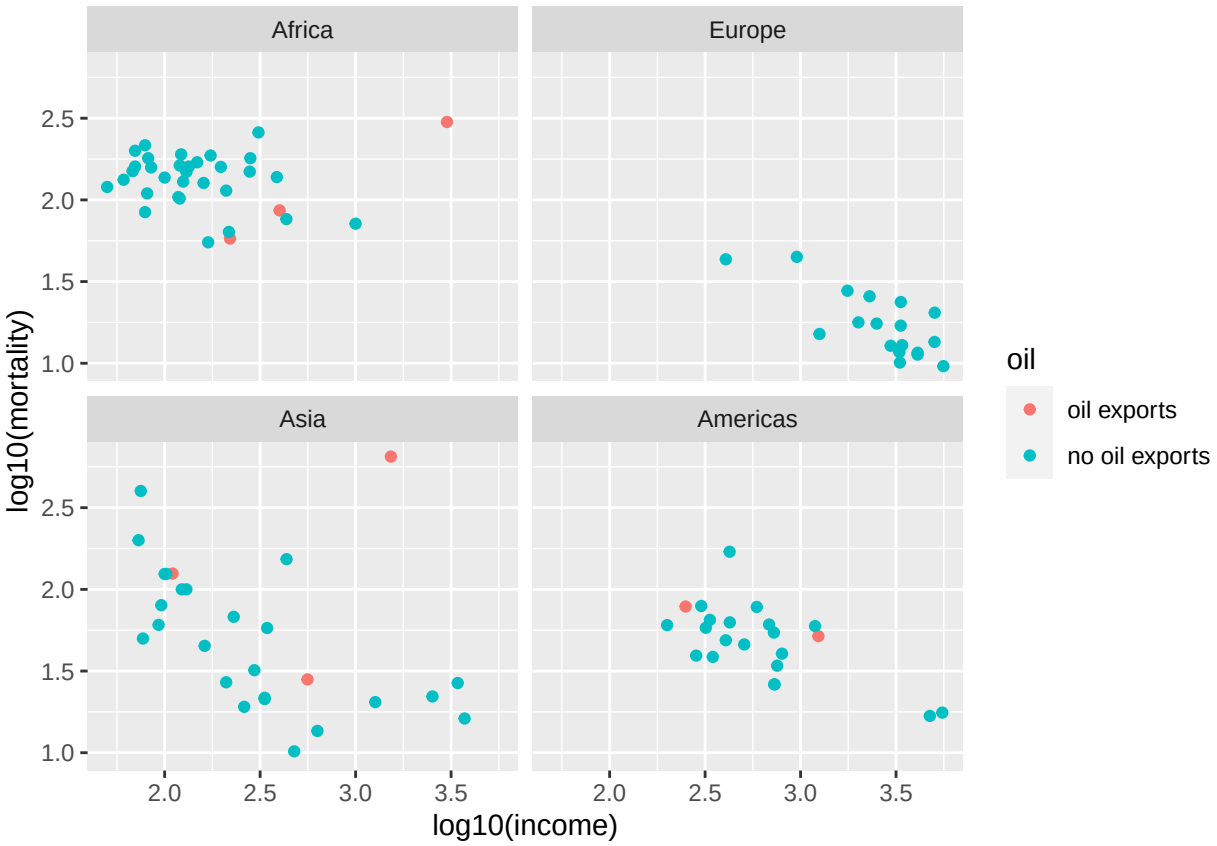
a) The `rownames()` of the table gives the country names and you should create a new column that contains the country names. *`rownames`

```
infmort.a <- infmort %>% mutate(Country=rownames(infmort))
head(infmort.a)
```

```
##                       region income mortality            oil
## Australia               Asia   3426      26.7 no oil exports
## Austria               Europe   3350      23.7 no oil exports
## Belgium               Europe   3346      17.0 no oil exports
## Canada              Americas   4751      16.8 no oil exports
## Denmark               Europe   5029      13.5 no oil exports
## Finland               Europe   3312      10.1 no oil exports
##                                Country
## Australia           Australia
## Austria             Austria
## Belgium             Belgium
## Canada              Canada
## Denmark             Denmark
## Finland             Finland
```

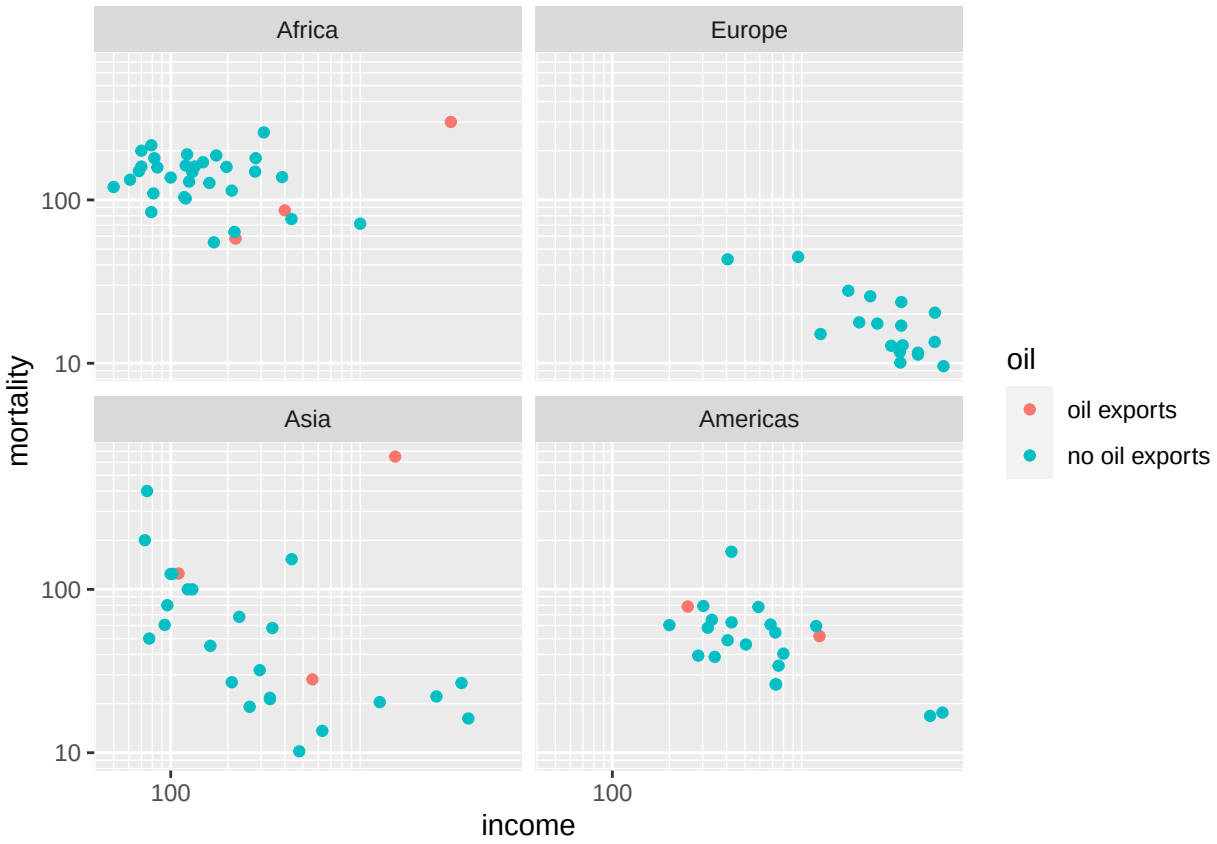
b. Create scatter plots with the `log10()` transformation inside the `aes()`command.

```
ggplot(infmort.a) +
  geom_point(aes(x=log10(income), y= log10(mortality), color=oil)) +
  facet_wrap( ~ region )
```



c. Create the scatter plots using the `scale_x_log10()` and `scale_y_log10()`.Set the major and minor breaks to be useful and aesthetically pleasing.Comment on which version you find easier to read.

```
my.plot <- ggplot(infmort.a) +
  geom_point(aes(x=income, y= mortality, color=oil)) + facet_wrap( ~ region )+
scale_y_log10(breaks=c(1,10,100),
                  minor=c(1:10,
                        seq( 10, 100,by=10 ),
                        seq(100,1000,by=100))) +
scale_x_log10(breaks=c(1,10,100),
                  minor=c(1:10,
                        seq( 10, 100,by=10 ),
                        seq(100,1000,by=100)))
my.plot
```



The second graphs are easier to read since I don't have to exponentiate in my head. The math is done for the reader.

d. The package `ggrepel` contains functions `geom_text_repel()` and `geom_label_repel()` that mimic the basic `geom_text()` and `geom_label()`functions in `ggplot2`, but work to make sure the labels don't overlap. Select 10-15 countries to label and do so using the `geom_text_repel()` function.

```
annotation.data <- sample_n(infmort.a, size=10) %>% select(Country, region, income, mortality)

annotation.data
```

```
##                                Country   region income mortality
## Argentina           Argentina           Americas   1191      59.6
## Cambodia            Cambodia               Asia    123     100.0
## Dominican_Republic  Dominican_Republic  Americas    406      48.8
## Ghana               Ghana                 Africa    217      63.7
## Thailand            Thailand               Asia    210      27.0
## Ethiopia            Ethiopia              Africa     79      84.2
## Zambia              Zambia                Africa    310     259.0
## Uganda              Uganda                Africa    134     160.0
## Japan               Japan                 Europe   3292      11.7
## Canada              Canada              Americas   4751      16.8
```

```
my.plot +
  geom_text_repel(data=annotation.data, aes(x=income, y=mortality, label=Country))
```

Problem 2: Chapter 14 Problem 2

Using the `datasets::trees` data, complete the following:

a. Create a regression model for $y =$ `Volume` as a function of $x =$ `Height`.

```
my.mod <- lm(data=trees, Volume ~ Height)
my.mod
```

```
##
## Call:
## lm(formula = Volume ~ Height, data = trees)
##
## Coefficients:
## (Intercept)       Height
##     -87.124        1.543
```

b. Using the `summary` command, get the y-intercept and slope of the regression line.

```
my.slope <- round(summary(my.mod)$coefficients[2,1], 2)
my.intercept <- round(summary(my.mod)$coefficients[1,1], 2)
my.slope
```

```
## [1] 1.54
```

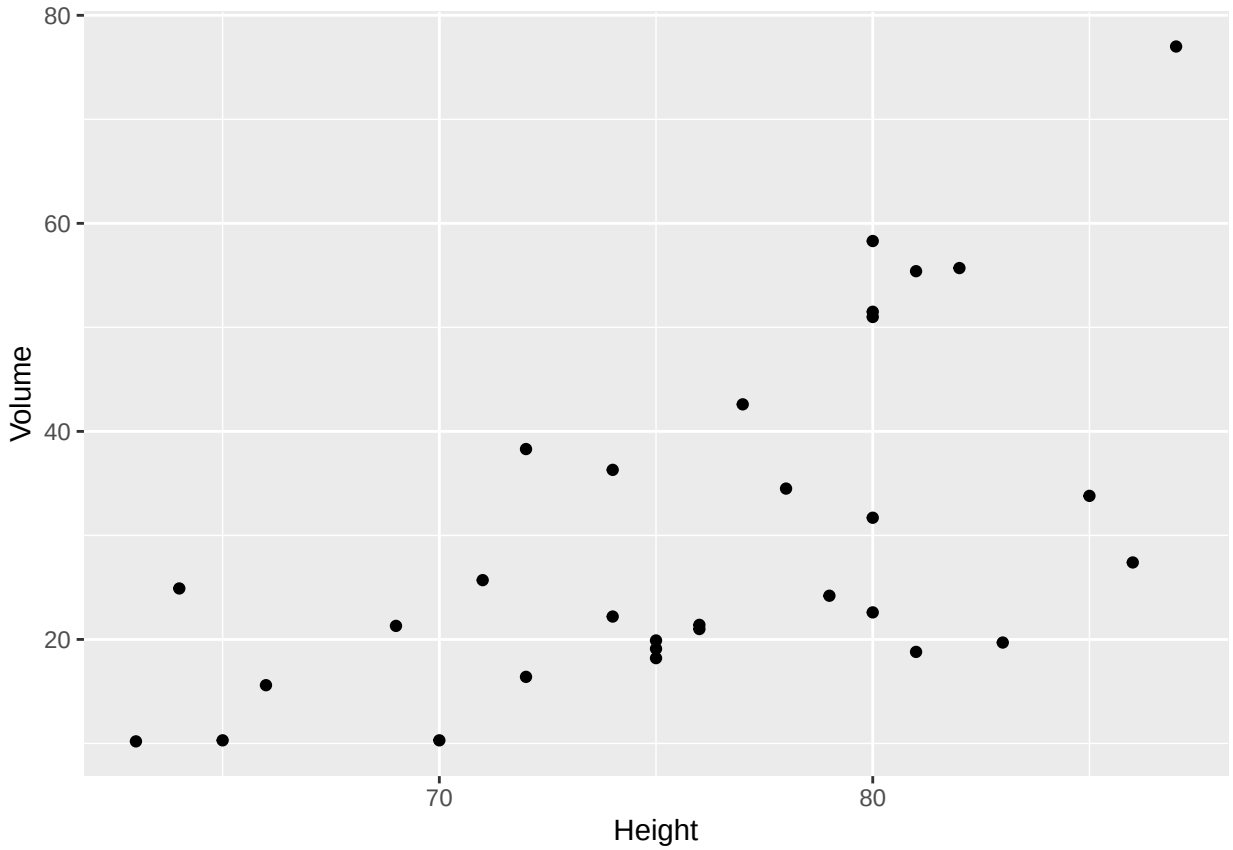
```
my.intercept
```

```
## [1] -87.12
```

c. Using `ggplot2`, create a scatter plot of Volume vs Height.

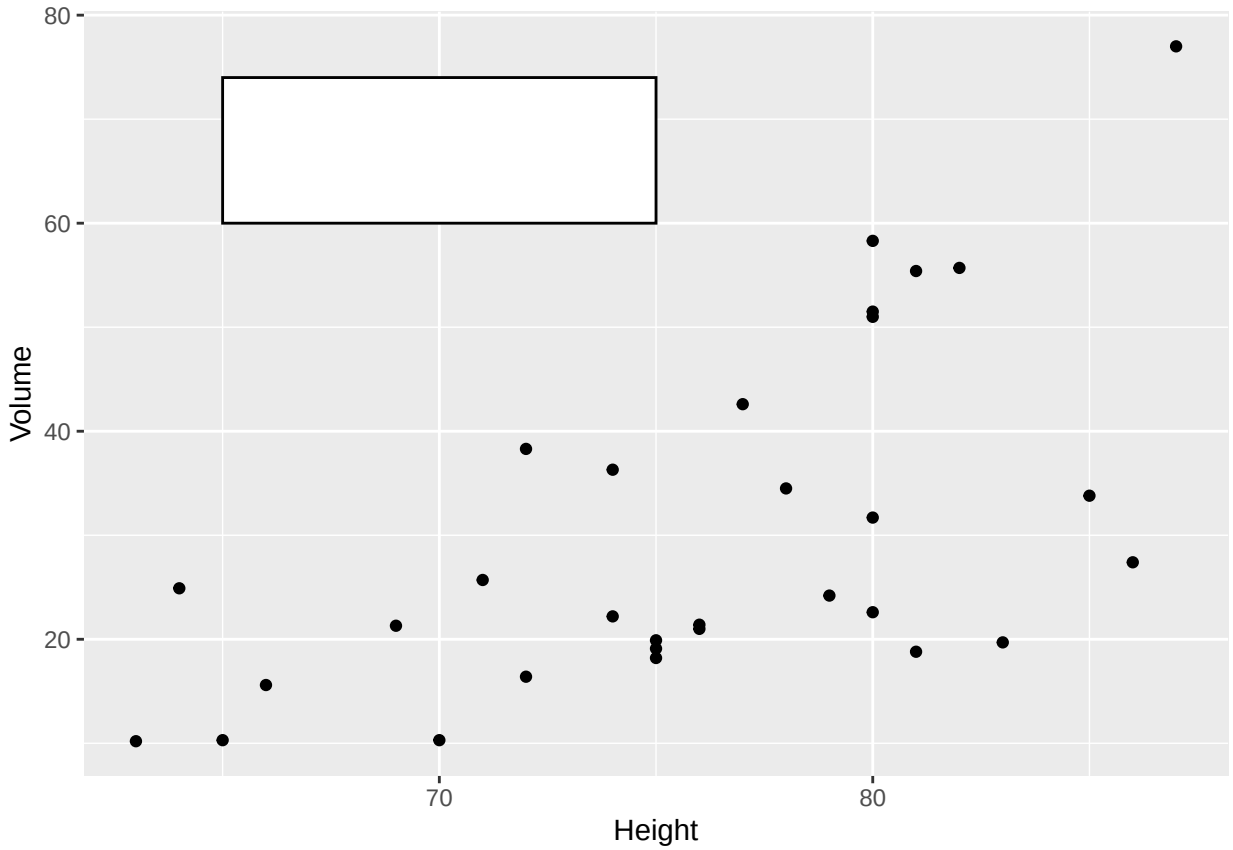
```
plot1 <- ggplot(data=trees) +
  geom_point(aes(x= Height, y=Volume))

plot1
```



d. Create a nice white filled rectangle to add text information to using by adding the following annotation layer.

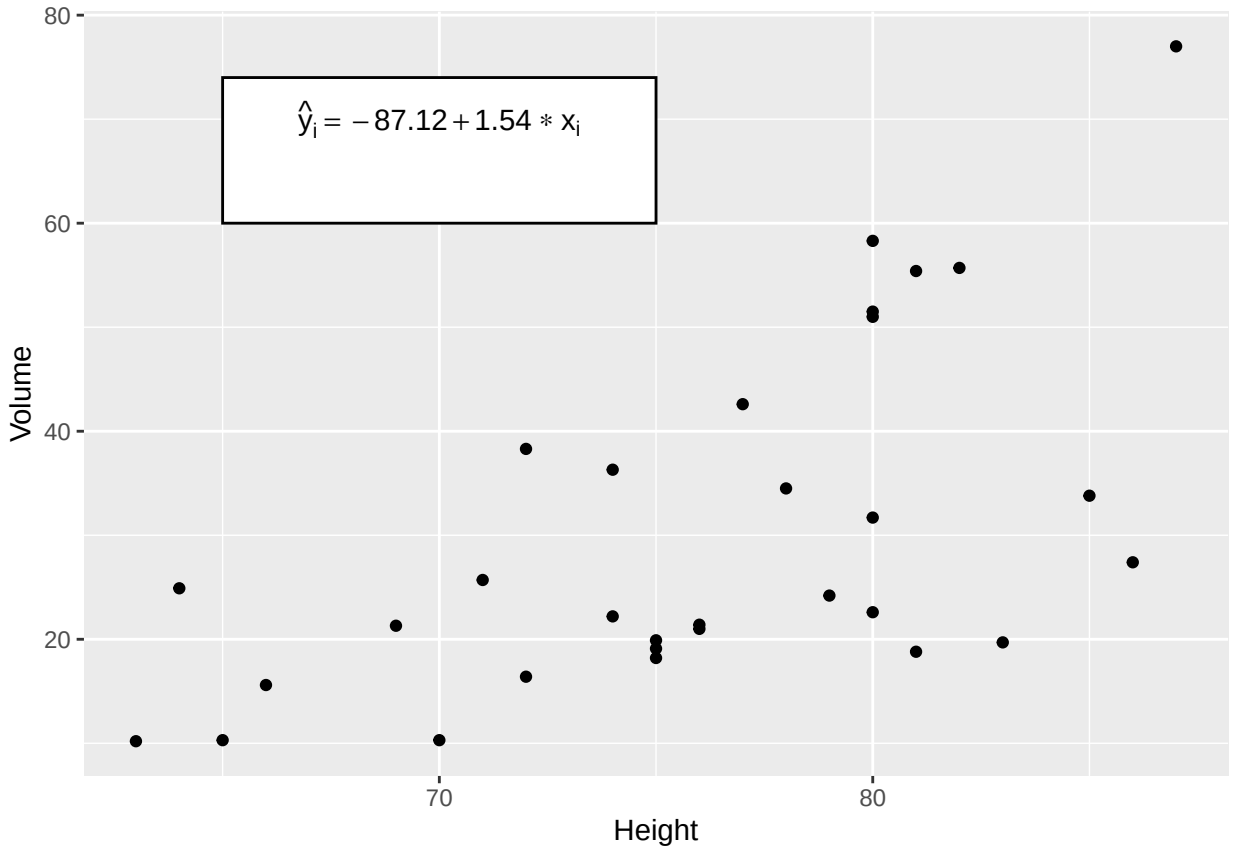
```
plot2 <- plot1 +
  annotate('rect', xmin=65, xmax=75, ymin=60, ymax=74, fill='white', color='black')
plot2
```



e.

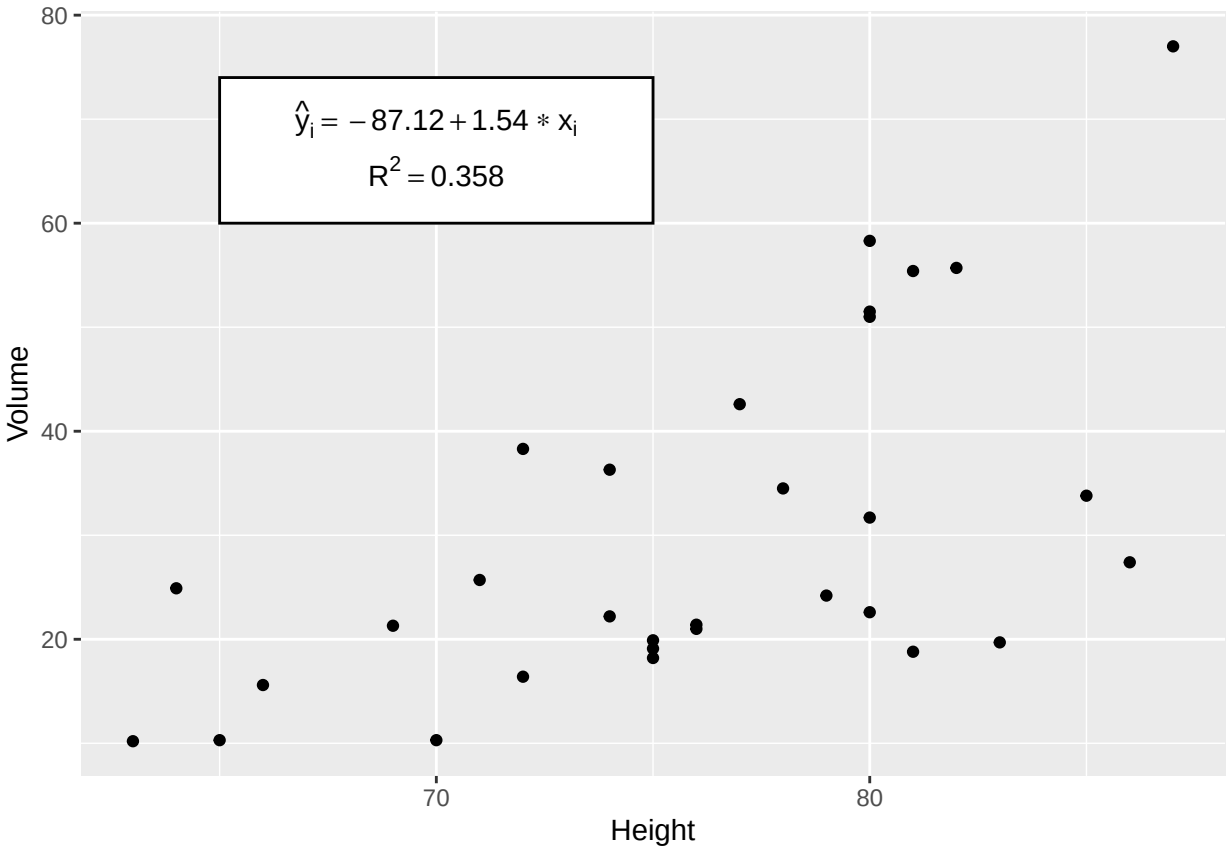
Add some annotation text to write the equation of the line $\hat{y}_i = -87.12 + 1.54 * x_i$ in the text area.

```
plot3 <- plot2 +
  annotate('text', x=70,  y=70, label=latex2exp::TeX("$\\hat{y}_i = -87.12 + 1.54 * x_i$"))
plot3
```



f. Add annotation to add $R^2 = 0.358$

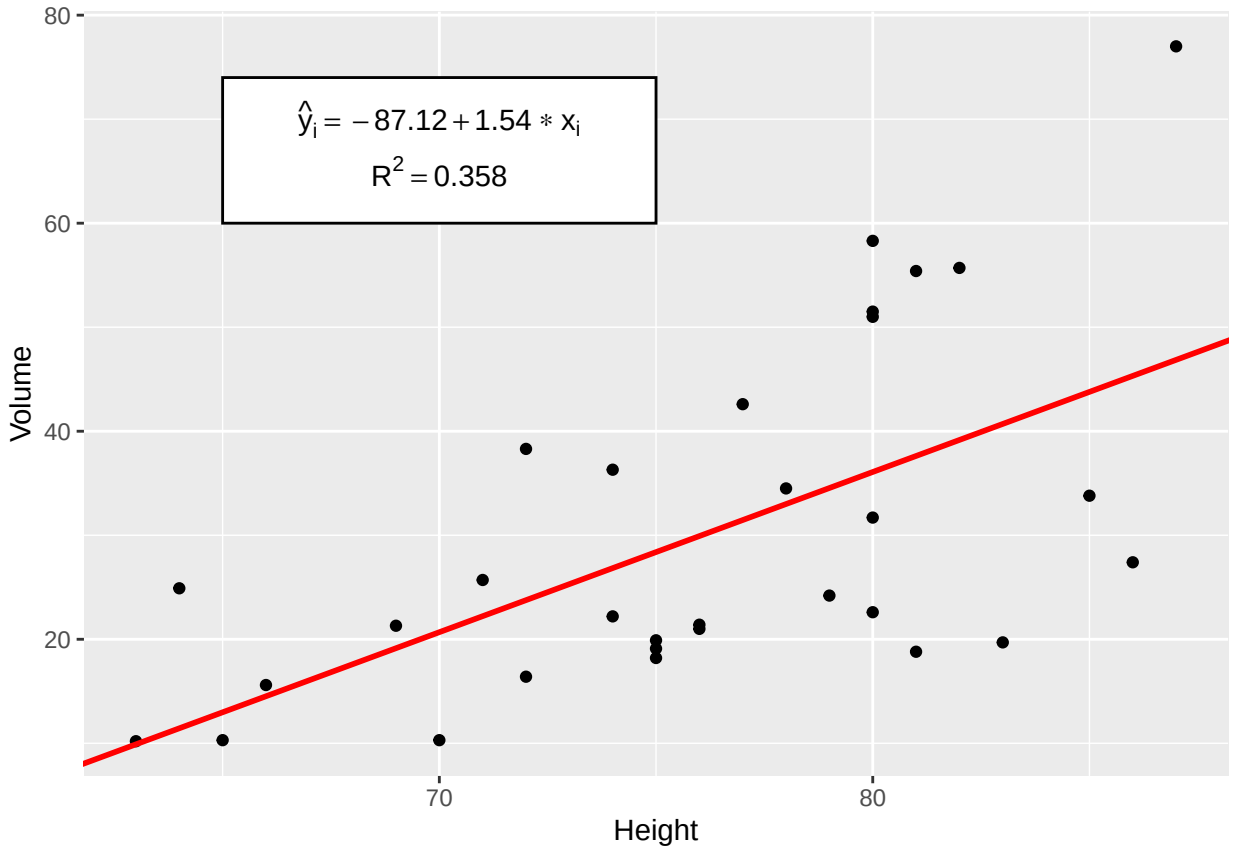
```
plot4 <- plot3 +
  annotate('text', x=70,  y=65, label=latex2exp::TeX("$R^2 = 0.358"))
plot4
```



g.

Add the regression line in red. The most convenient layer function to uses is `geom_abline()`.

```
plot4 +
  geom_abline(slope = my.slope, intercept = my.intercept,
                    color="red", size=1.0)
```


$\hat{y}_i = -87.12 + 1.54 * x_i$
$R^2 = 0.358$
Volume
Height
80
60
40
20
70
80