STA_445_Assignment 7

Sophia Kubisiak

04/02/2024

Load your packages here:

```
library(tidyverse)
library(patchwork)
library(viridis)
library(latex2exp)
library(plotly)
library(ggplot2)
```

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 The information is relatively out of date, but will be fun to graph.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.

```
library(faraway)
```

```
## Warning in check_dep_version(): ABI version mismatch:
## lme4 was built with Matrix ABI version 1
## Current Matrix ABI version is 0
## Please re-install lme4 from source or restore original 'Matrix' package
```

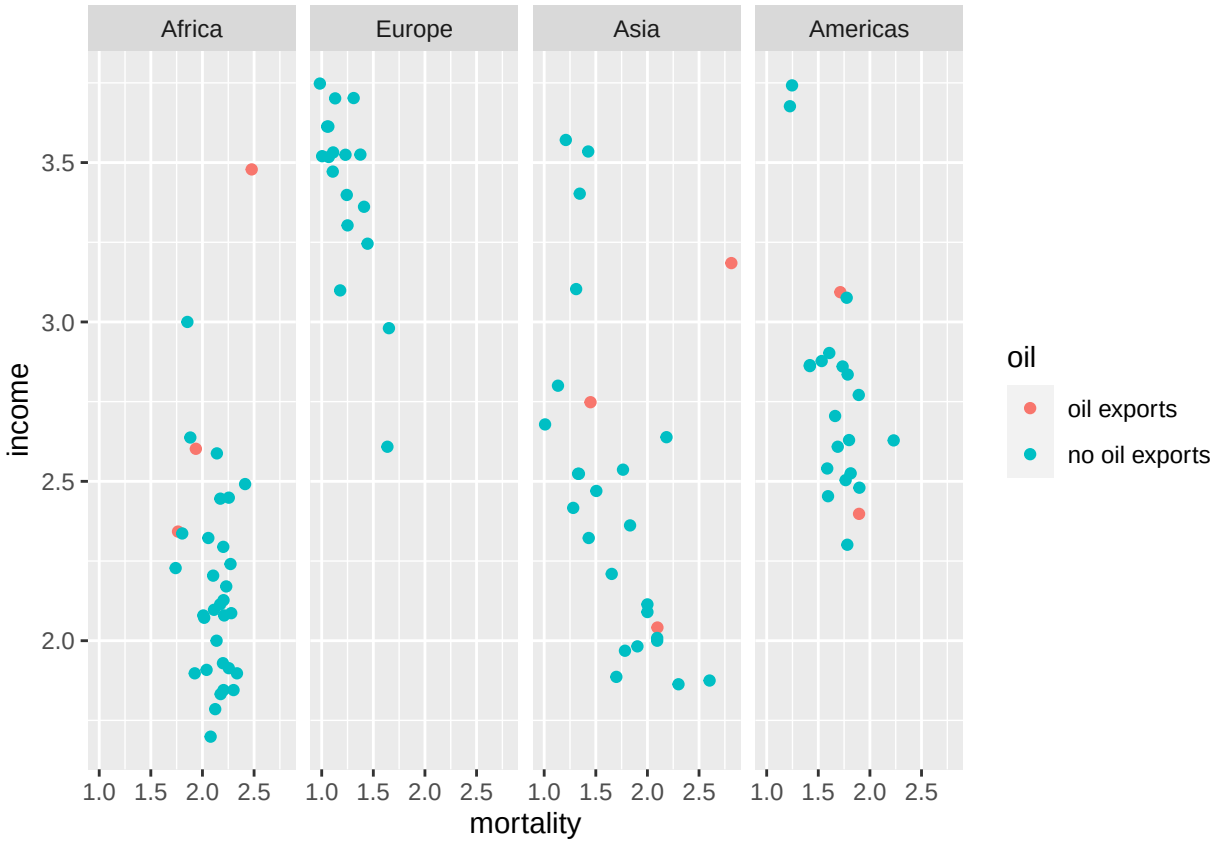
```
data(infmort)
```

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

```
infmort2 <- infmort %>%
  mutate(country = rownames(infmort))
```

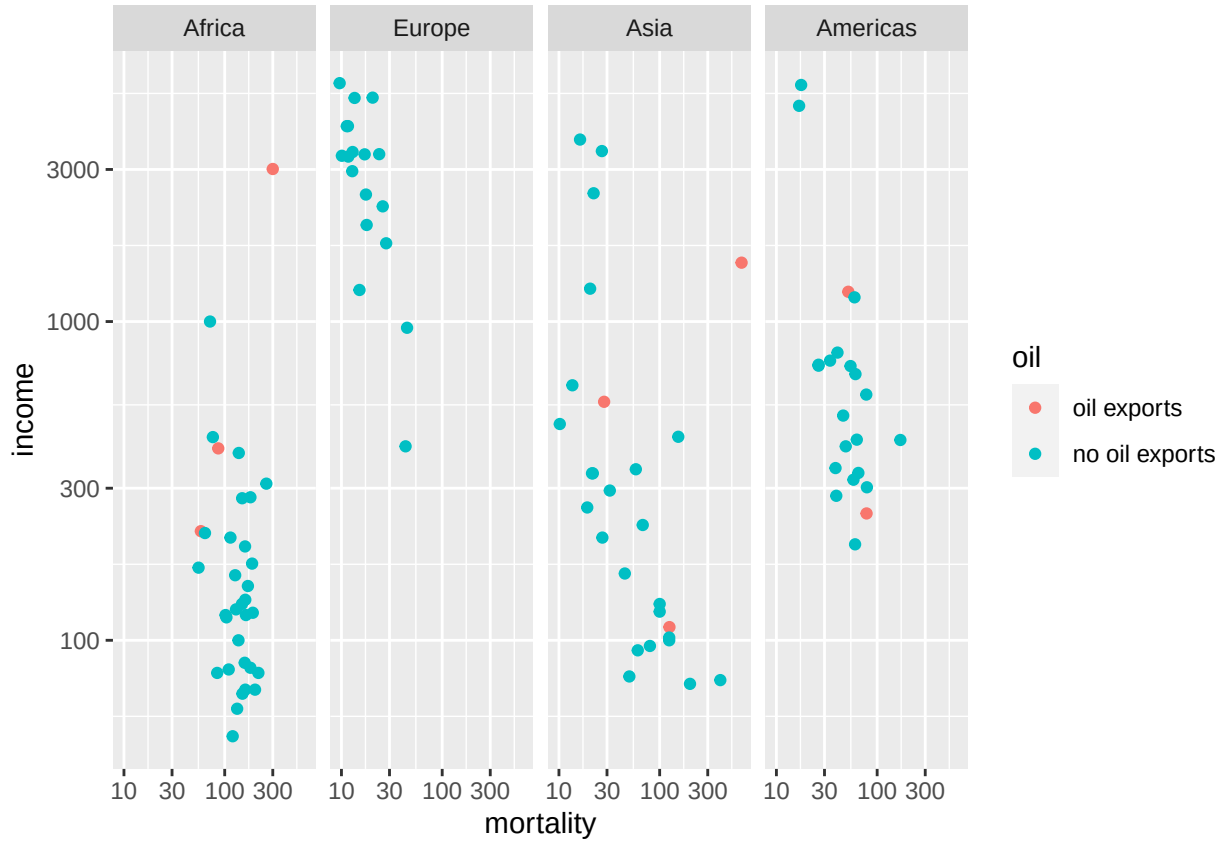
b. Create scatter plots with the `log10()` transformation inside the `aes()`command.

```
ggplot(data=infmort2, aes(x = log10(mortality) ,y = log10(income))) +
  geom_point(aes(color=oil))+
  facet_grid(facets = . ~ region)+
  labs(x = 'mortality' , y = 'income')
```



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.

```
ggplot(data=infmort2, aes(x = mortality , y = income)) +
  geom_point(aes(color = oil)) +
  facet_grid(facets = . ~ region)+
  scale_x_log10()+
  scale_y_log10()
```



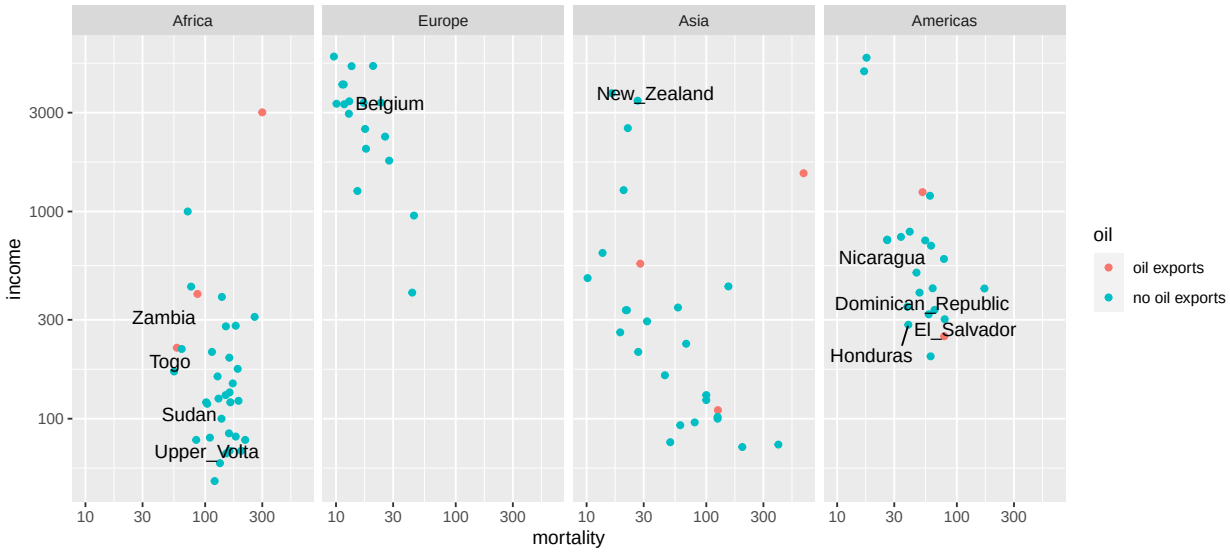
I think that the scale_x_log10 and scale_y_log10 is easier to read, because they are layers added to the graph. It's easier to see the scale change.

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.

```
library(ggrepel)

countries <- slice_sample(infmort2 , n=10)

ggplot(data = infmort2) +
  geom_point(aes(x = mortality , y = income , color = oil)) +
  facet_grid(facets = . ~ region)+
  scale_x_log10()+
  scale_y_log10()+
  geom_text_repel(data = countries , aes( x = mortality , y = income , label = country))
```



Problem 2

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

```
library(datasets)
```

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

```
model <- lm(Volume ~ Height, data = trees)
```

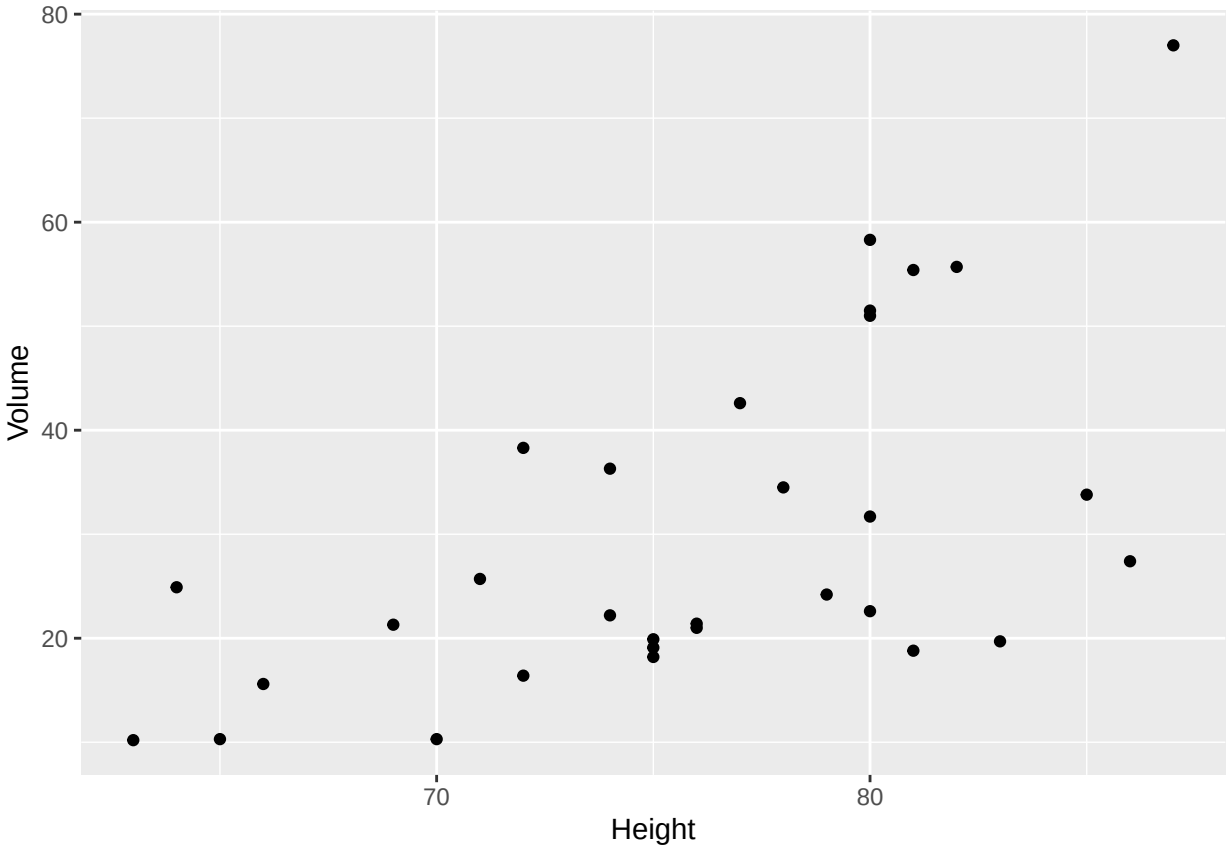
b. Using the str(your model's name) command, to get a list of all the information stored in the linear model object. Use $ to extract the slope and intercept of the regression line (the coefficients).

```
summary(model)$coefficients
```

```
##              Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) -87.12361 29.2731221 -2.976232 0.0058346689
## Height        1.54335  0.3838693  4.020509 0.0003783823
```

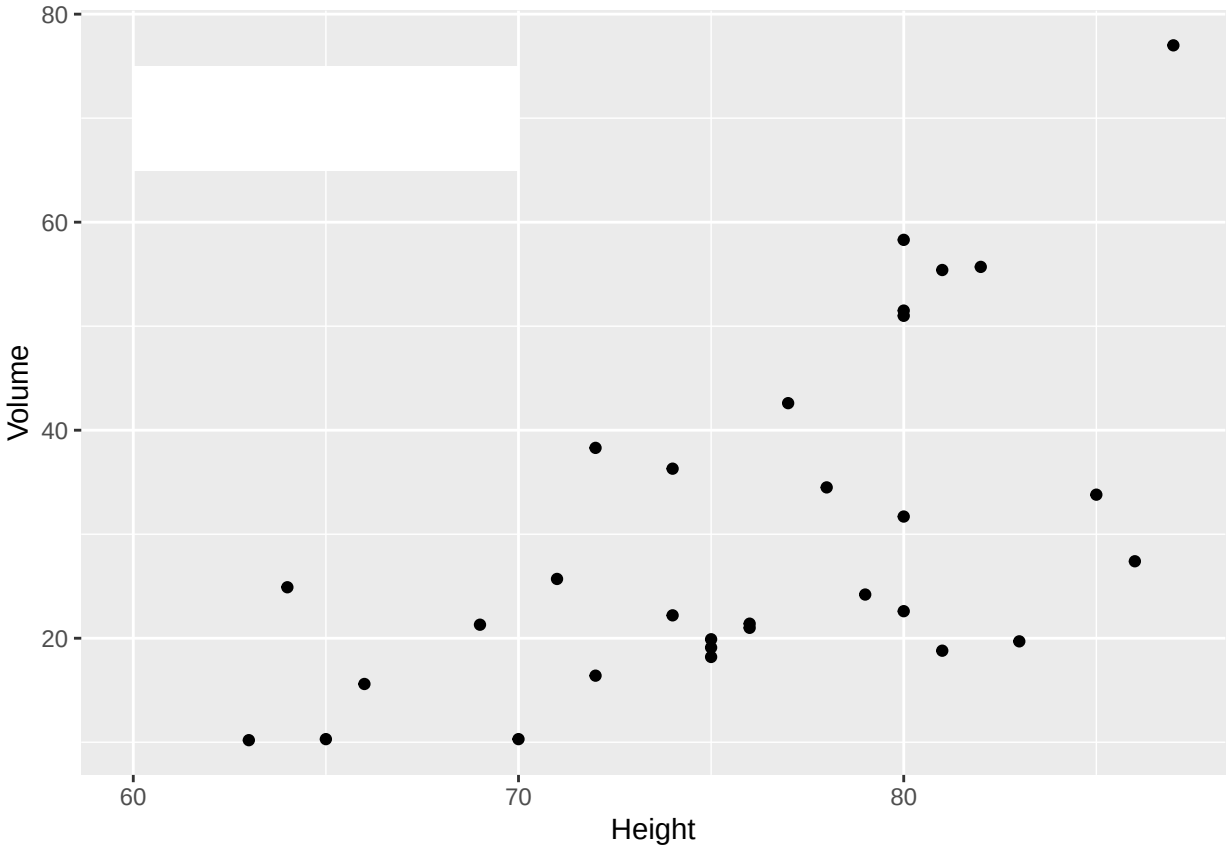
c. Using `ggplot2`, create a scatter plot of Volume vs Height.

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



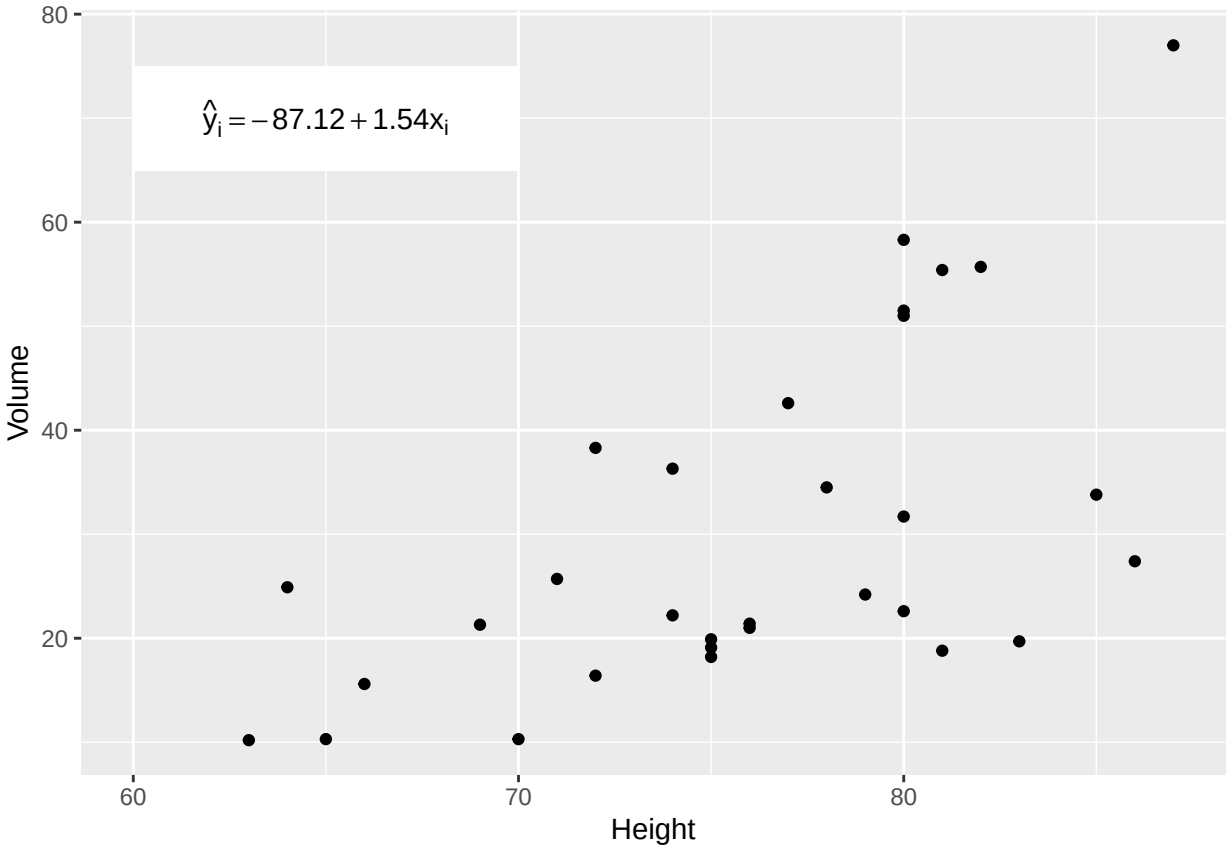
d. Create a nice white filled rectangle to add text information to using by adding the following annotation layer.

```
ggplot(data = trees, aes(x = Height, y = Volume)) +
  geom_point() +
  annotate('rect', xmin=60, xmax=70, ymin=65, ymax=75,  fill='white')
```



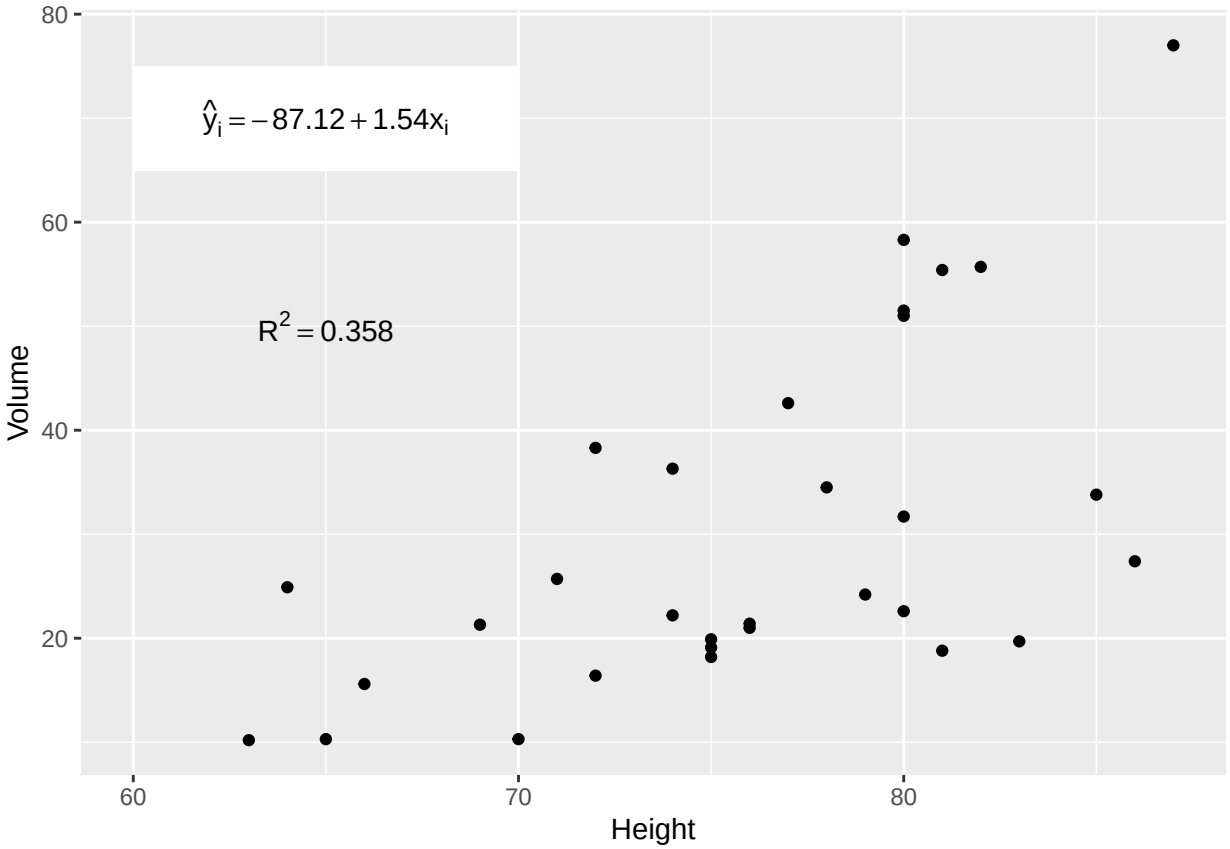
e. Add some annotation text to write the equation of the line $\hat{y}_i = -87.12 + 1.54 * x_i$ in the text area.

```
ggplot(data = trees, aes(x = Height, y = Volume)) +
  geom_point() +
  annotate('rect', xmin=60, xmax=70, ymin=65, ymax=75,  fill='white') +
  annotate('text', x = 65, y = 70, label = expression(hat(y)[i] == -87.12 + 1.54*x[i]))
```



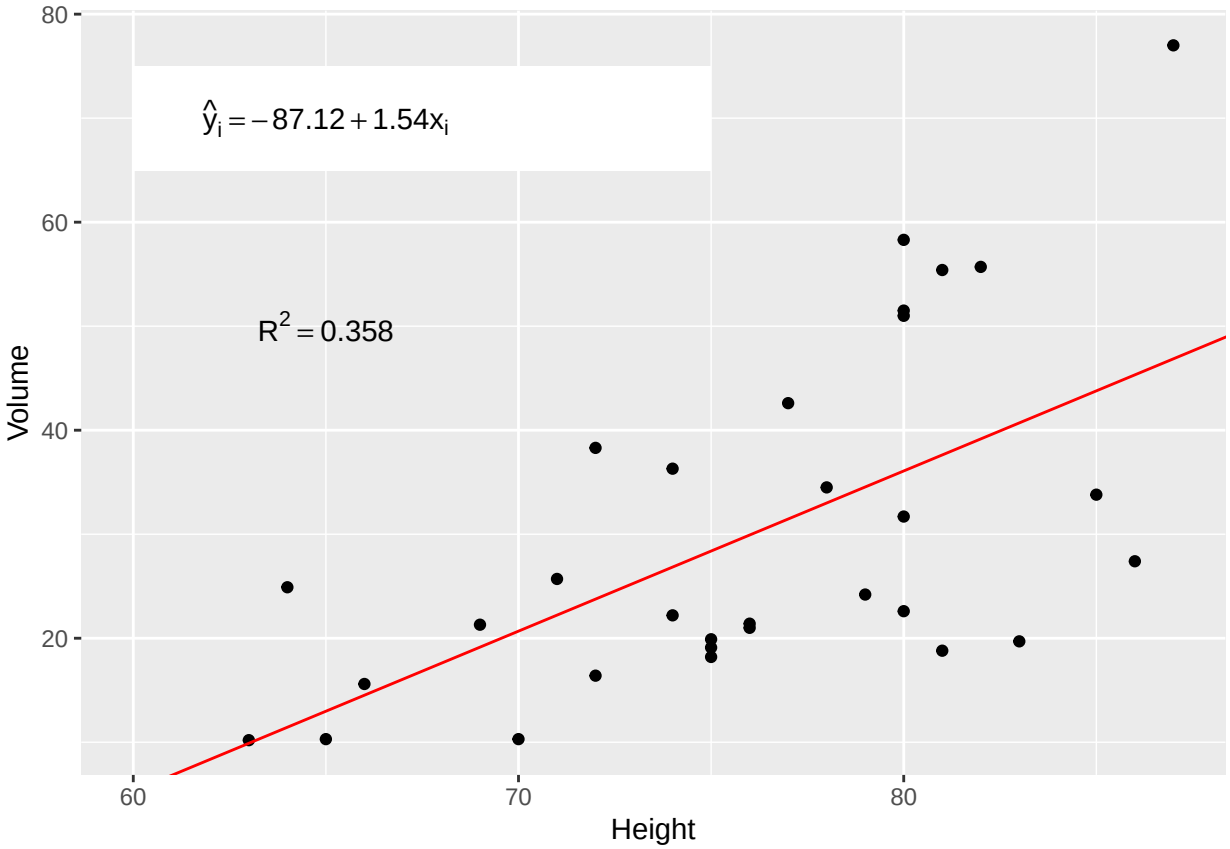
f. Add annotation to add $R^2 = 0.358$

```
ggplot(data = trees, aes(x = Height, y = Volume)) +
  geom_point() +
  annotate('rect', xmin=60, xmax=70, ymin=65, ymax=75,  fill='white') +
  annotate('text', x = 65, y = 70, label = expression(hat(y)[i] == -87.12 + 1.54*x[i])) +
  annotate('text', x = 65, y = 50, label = expression(R^2 == 0.358))
```



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

```
ggplot(data = trees, aes(x = Height, y = Volume)) +
  geom_point() +
  annotate('rect', xmin=60, xmax=75, ymin=65, ymax=75,  fill='white') +
  annotate('text', x = 65, y = 70, label = expression(hat(y)[i] == -87.12 + 1.54*x[i])) +
  annotate('text', x = 65, y = 50, label = expression(R^2 == 0.358)) +
  geom_abline(intercept = -87.12, slope = 1.54, color = "red")
```



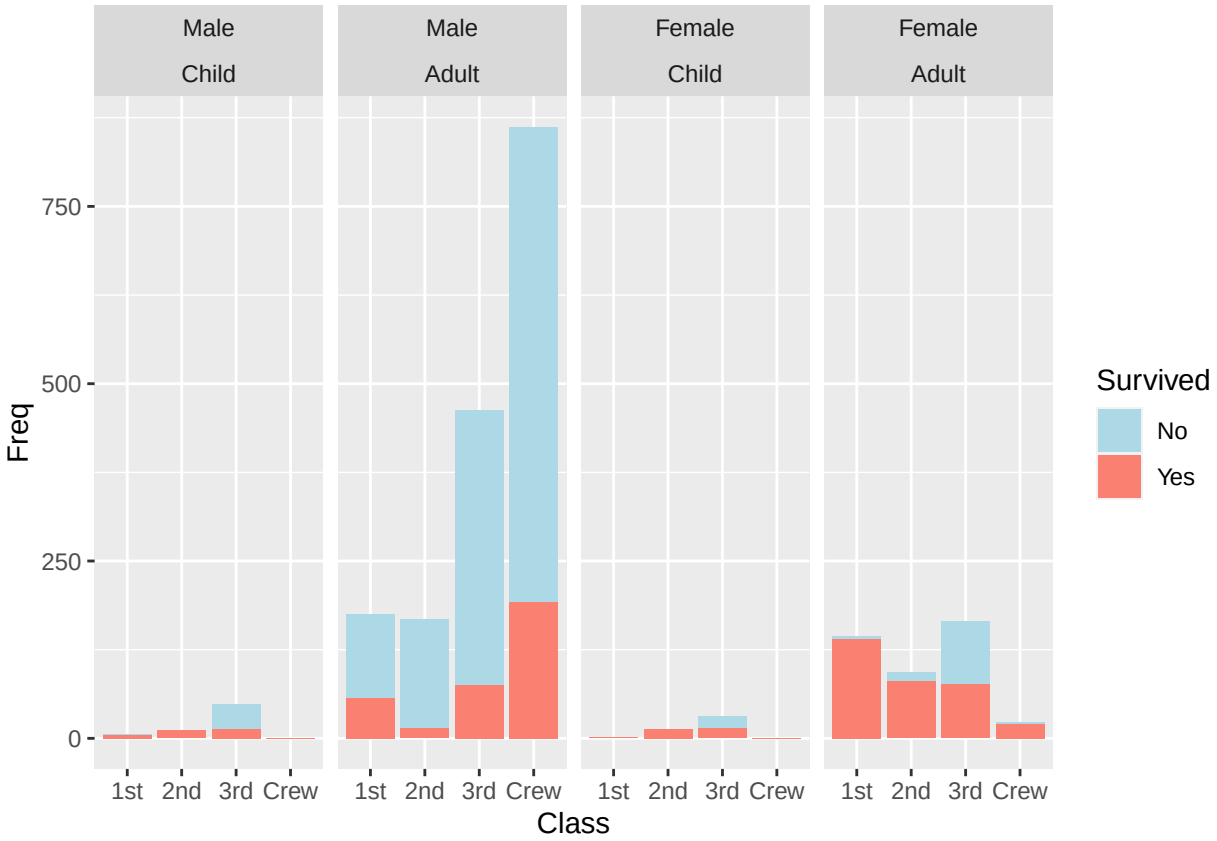
Problem 3

In `datasets::Titanic` table summarizes the survival of passengers aboard the ocean liner *Titanic*. It includes information about passenger class, sex, and age (adult or child). Create a bar graph showing the number of individuals that survived based on the passenger `Class`, `Sex`, and `Age` variable information. You'll need to use faceting and/or color to get all four variables on the same graph. Make sure that differences in survival among different classes of children are perceivable. *Unfortunately, the data is stored as a* `table` *and to expand it to a data frame, the following code can be used.*

```r
Titanic <- Titanic %>% as.data.frame()
```

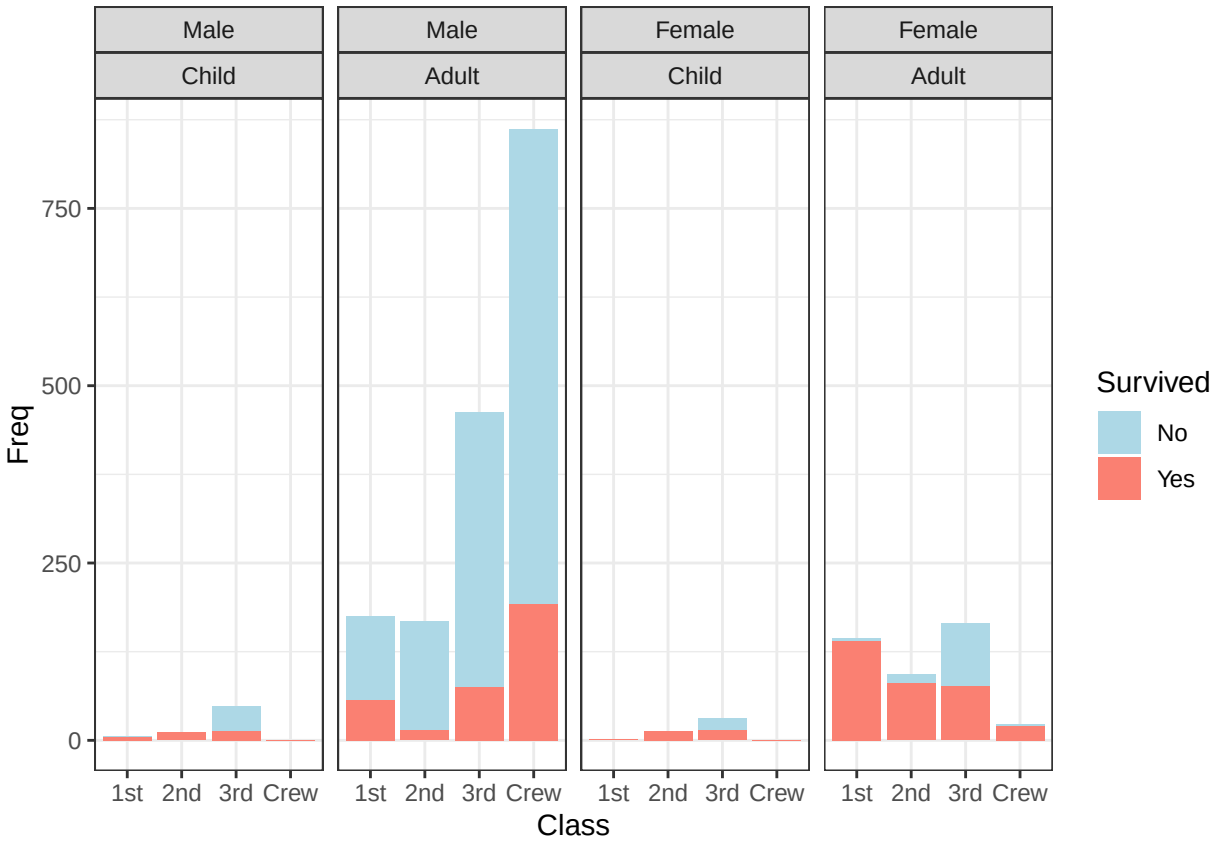
a. Make this graph using the default theme. *If you use color to denote survivorship, modify the color scheme so that a cold color denotes death.*

```r
ggplot(data = Titanic) +
 geom_bar(aes(x = Class, y = Freq , fill = Survived) , stat = "identity")+
 facet_grid(. ~ Sex + Age)+
 scale_fill_manual(values = c("lightblue", "salmon"), name = "Survived", labels=c("No", "Yes"))
```



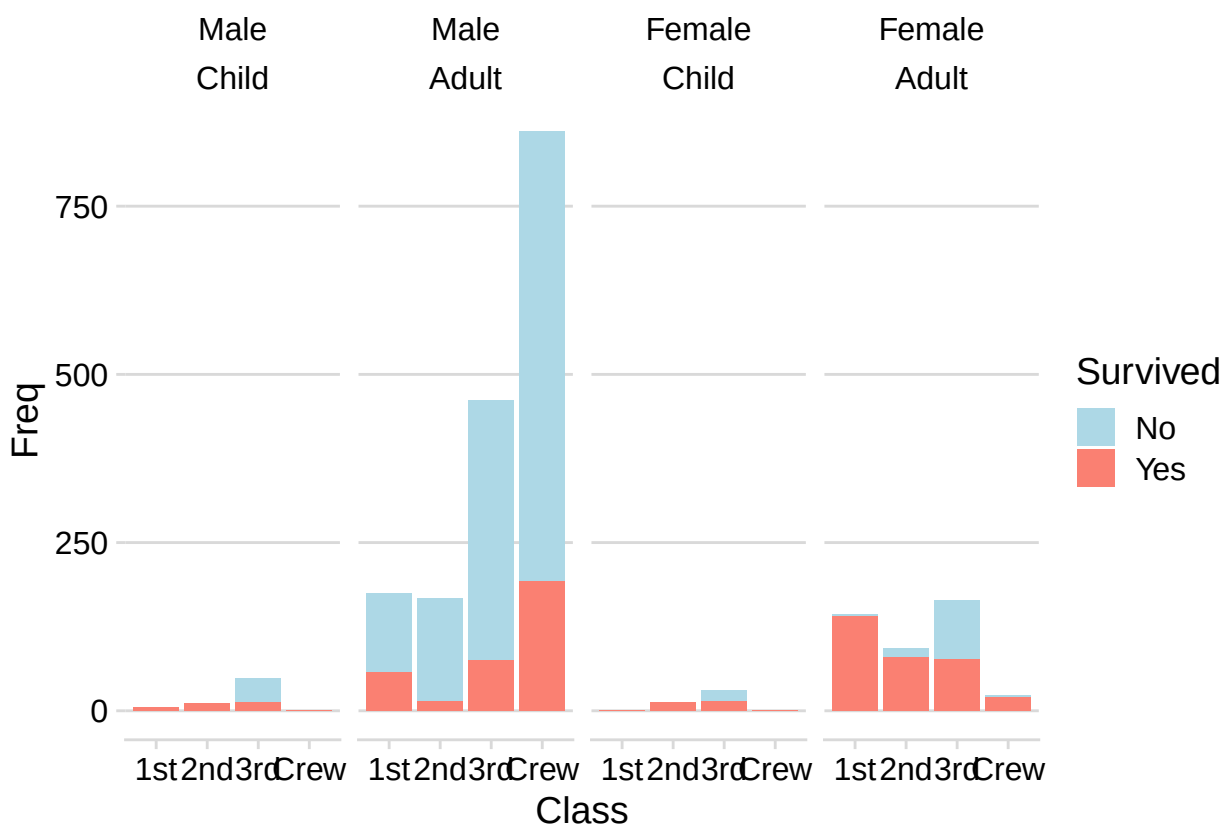
b. Make this graph using the `theme_bw()` theme.

```
ggplot(data = Titanic) +
  geom_bar(aes(x = Class, y = Freq , fill = Survived) , stat = "identity")+
  facet_grid(. ~ Sex + Age)+
  scale_fill_manual(values = c("lightblue", "salmon"), name = "Survived", labels=c("No", "Yes"))+
  theme_bw()
```



c. Make this graph using the `cowplot::theme_minimal_hgrid()` theme.

```
ggplot(data = Titanic) +
  geom_bar(aes(x = Class, y = Freq , fill = Survived) , stat = "identity")+
  facet_grid(. ~ Sex + Age)+
  scale_fill_manual(values = c("lightblue", "salmon"), name = "Survived", labels=c("No", "Yes"))+
  cowplot::theme_minimal_hgrid()
```



d. Why would it be beneficial to drop the vertical grid lines? No, because the vertical grid lines separate class. The lines make it easier to see each class.