Homework 2 Report

Name

Vijetha Ramdas

Setup

Read in the data with `read_csv()` and store the data as an R object named `dataset`. Check the data to make sure all of the expected observations and variables are there.

```
## Load the data and any necessary packages here.
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.2     v tibble    3.3.0
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.1.0
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
df <- read_csv("../maacs.csv.gz")
```

```
## Rows: 150 Columns: 10
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (3): id, gender, mouse_allergic
## dbl (7): age, fev1, eNO, cough_days, pm25, no2, mouse
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Part 1

We will first consider the relationship between FEV1 and age. In general, it is expected that as children get older (and hence, larger in size), their FEV1 values should get higher.

Consider the statement "FEV1 values in children are higher in older children relative to younger children".

Write a function in R that takes the `dataset` object as an argument and returns `TRUE` if the statement above is true for the dataset and `FALSE` otherwise.

NOTE: In order to write this function, you will need to translate the statement above into something that can be checked with the data. There are many ways in which you can do that translation correctly and you only need to pick one way here.

NOTE: For this part, do not use any plots.

```r
## Write your function here

test_statement <- function(df) {
  mid <- df$age |> median()
  df = df |>
    mutate(group = ifelse(age >= mid, "Old", "Young"))

  older_children_mean <- df |>
    filter(group == "Old") |>
    summarise(mean(fev1, na.rm = TRUE))

  younger_children_mean <- df |>
    filter(group == "Young") |>
    summarise(mean(fev1, na.rm = TRUE))

  mean_diff <- older_children_mean - younger_children_mean

  if (mean_diff > .5) {
    return(TRUE)}
  else{
    return(FALSE)}
}

test_statement(df)
```

```
## [1] TRUE
```

Part 2

Fit a linear regression model with FEV1 as the outcome and age as a predictor.

How much does FEV1 change for a 1-year increase in the child's age?

```r
## Add your code here

model <- lm(fev1 ~ age, data = df)
summary(model)
```

```
##
## Call:
## lm(formula = fev1 ~ age, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.51099 -0.27100  0.00196  0.23616  2.15477
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.001199   0.134696  -0.009    0.993
## age          0.171162   0.010984  15.583   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4827 on 131 degrees of freedom
##   (17 observations deleted due to missingness)
```

```
## Multiple R-squared:  0.6496, Adjusted R-squared:  0.6469
## F-statistic: 242.8 on 1 and 131 DF,  p-value: < 2.2e-16
```

Write your data analysis statement interpreting the regression model here:

For a one year increase in the child's age, FEV1 increases by 0.171162 L.

Part 3

Develop **three** supporting premises derived from the data that support the statement you wrote in Part 2. These can be plots, other summary statistics, or model results.

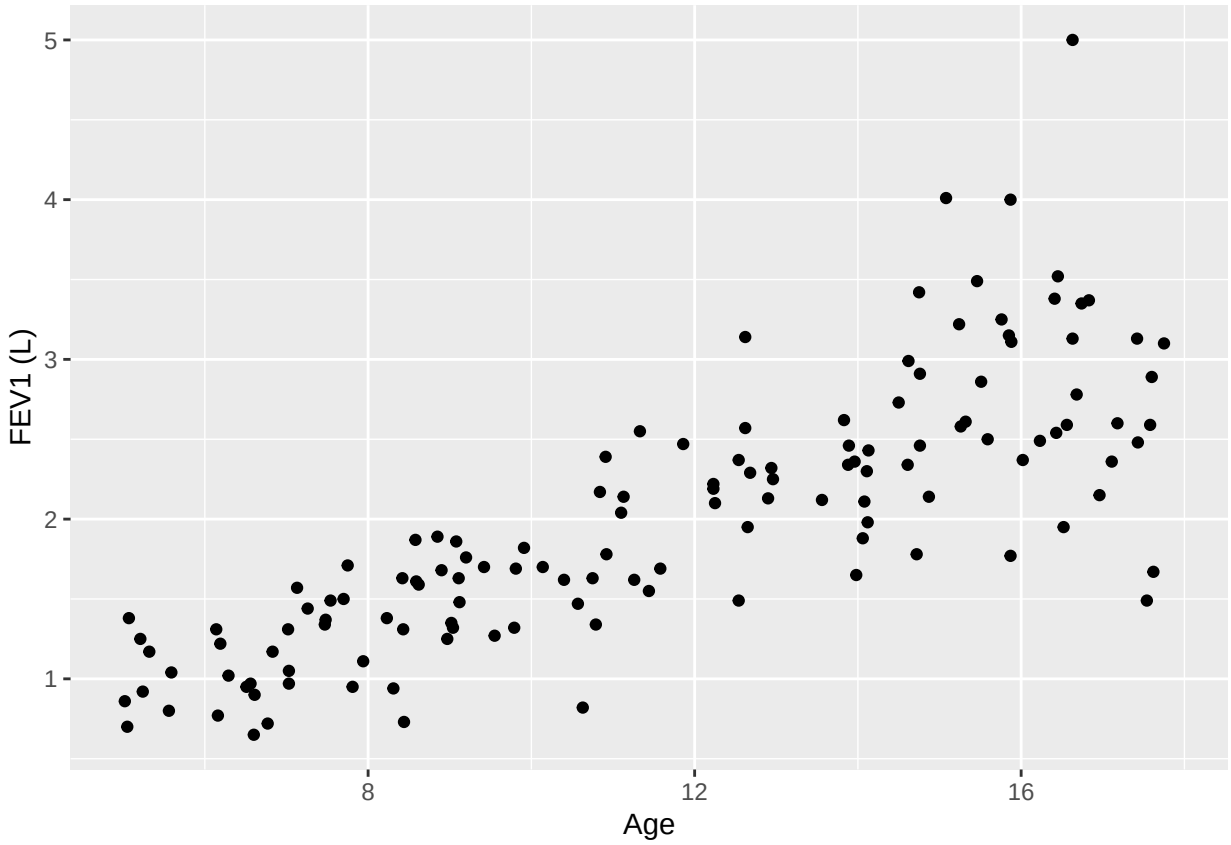
NOTE: At least one supporting premise should use a plot.

```
## Add your code here
mid <- df$age |> median()
print(mid)
```

```
## [1] 10.915
```

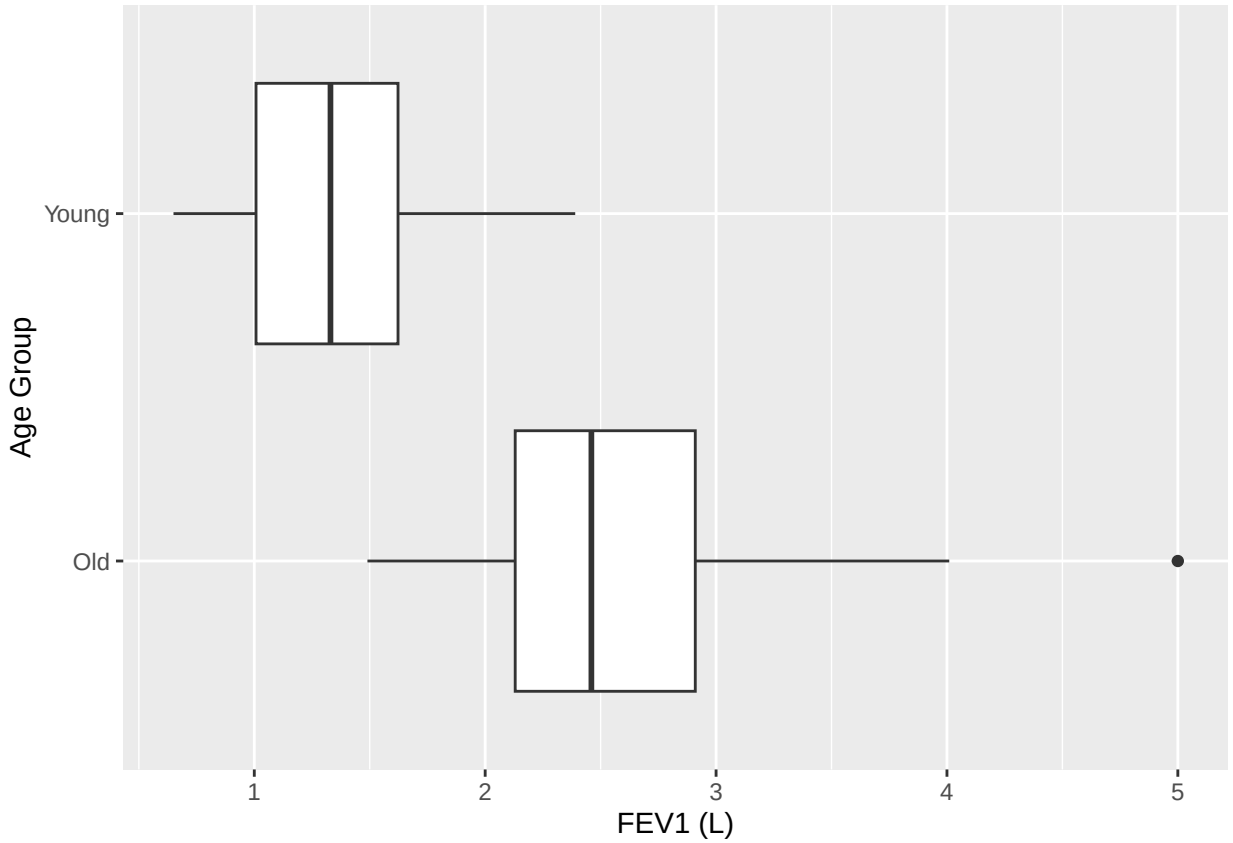
```
df = df |>
  mutate(group = ifelse(age >= mid, "Old", "Young"))

ggplot(df) +
  geom_point(aes(x = age, y = fev1)) +
  labs(x = "Age", y = "FEV1 (L)")
```

```
## Warning: Removed 17 rows containing missing values or values outside the scale range
## (`geom_point()`).
```



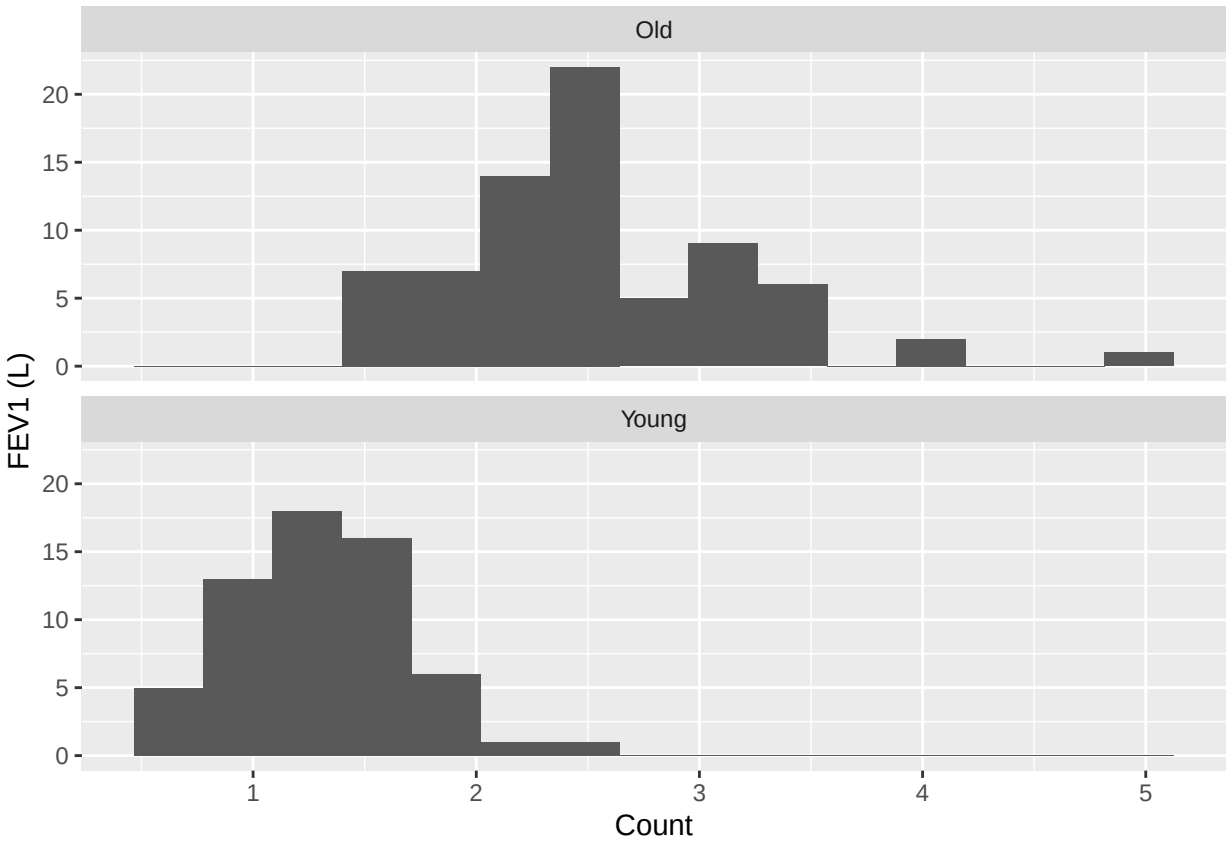
```
ggplot(df) +
  geom_boxplot(aes(x = fev1, y = group)) +
  labs(x = "FEV1 (L)", y = "Age Group")
```

```
## Warning: Removed 17 rows containing non-finite outside the scale range
## (`stat_boxplot()`).
```




```
ggplot(df) +
  geom_histogram(aes(x = fev1), bins = 15) +
  facet_wrap(~ group, nrow = 2) +
  labs(x = 'Count', y = "FEV1 (L)")
```

```
## Warning: Removed 17 rows containing non-finite outside the scale range
## (`stat_bin()`).
```



Write the three supporting premise statements here:

1.
2.
3.

Part 4

For each of the supporting premises above, write a function that takes the `dataset` object as an argument and returns `TRUE` if the supporting premise statement above is true for the dataset and `FALSE` otherwise.

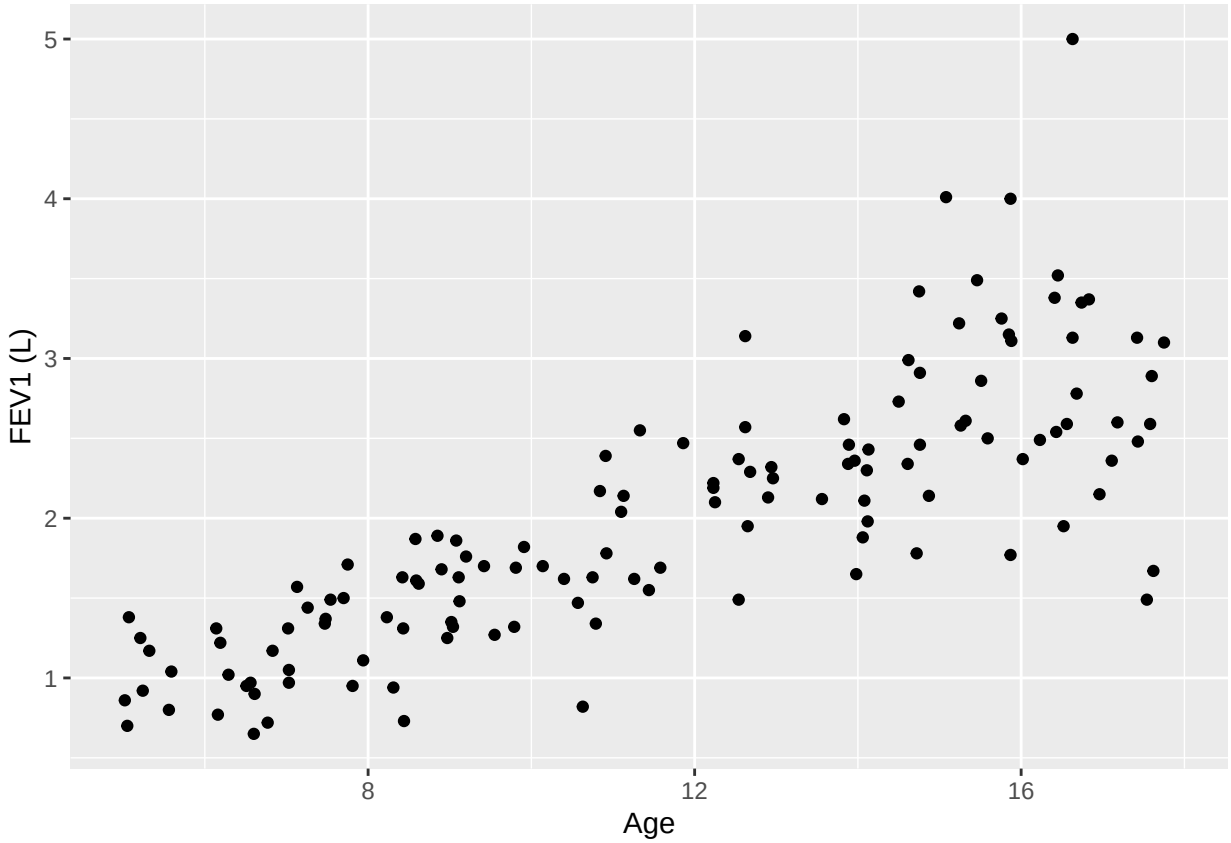
For statements involving plots, instead of returning `TRUE` or `FALSE`, you function should do two things:

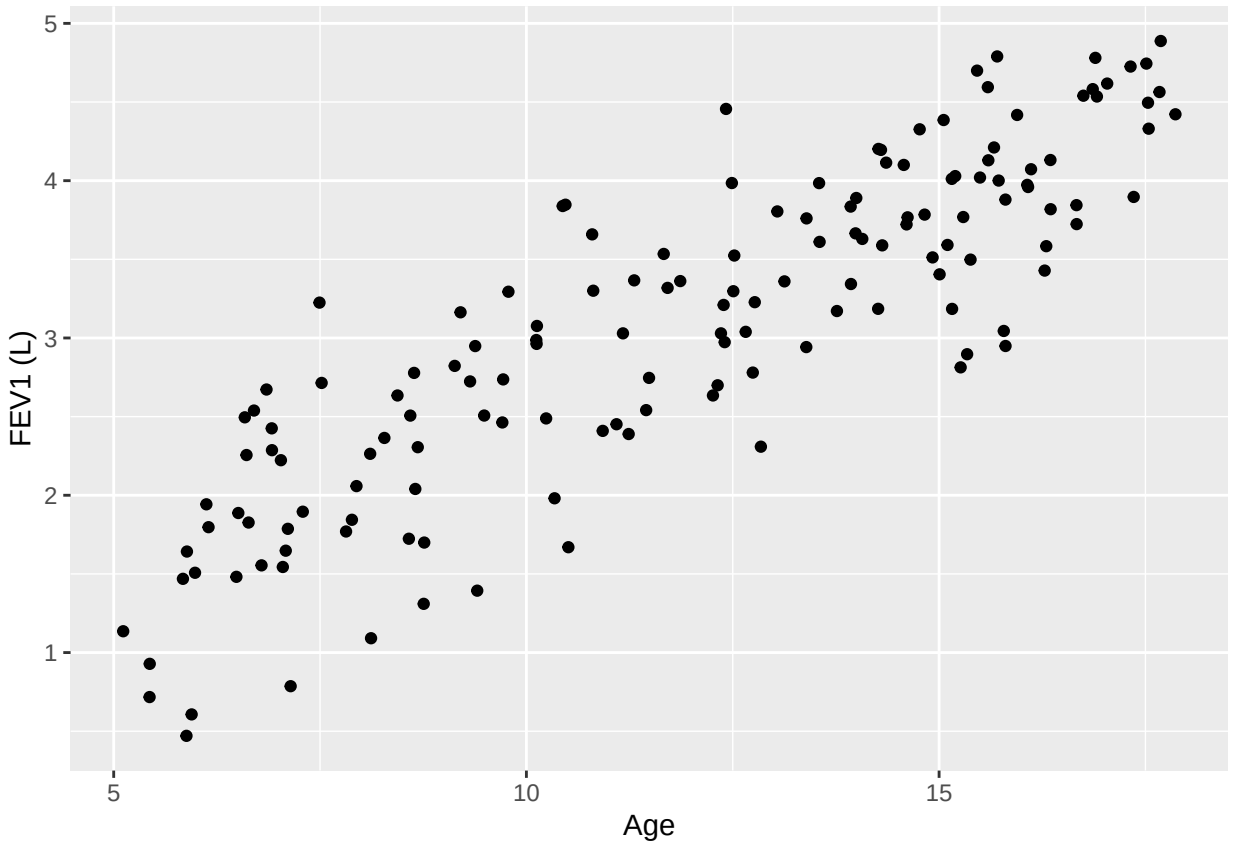
1. Produce the plot that is used in the statement
2. Produce a hypothetical version of the plot in the event that the statement is true. This can be done using simulated data or by simply hand drawing a plot.

```
set.seed(19)
## Function for supporting premise statement 1
premise_1 <- function(df){
  real_plot <- ggplot(df) +
    geom_point(aes(x = age, y = fev1)) +
    labs(x = "Age", y = "FEV1 (L)")

  # generate distribution
  x = runif(150, min(df$age), max(df$age))
  y = .25*x + rnorm(75, 0, .5)

  fake_dist <- tibble(x, y)
```

```r
  fake_plot <- ggplot(fake_dist) +
    geom_point(aes(x = x, y = y)) +
    labs(x = "Age", y = "FEV1 (L)")

  print(real_plot)
  print(fake_plot)
}

premise_1(df)
```

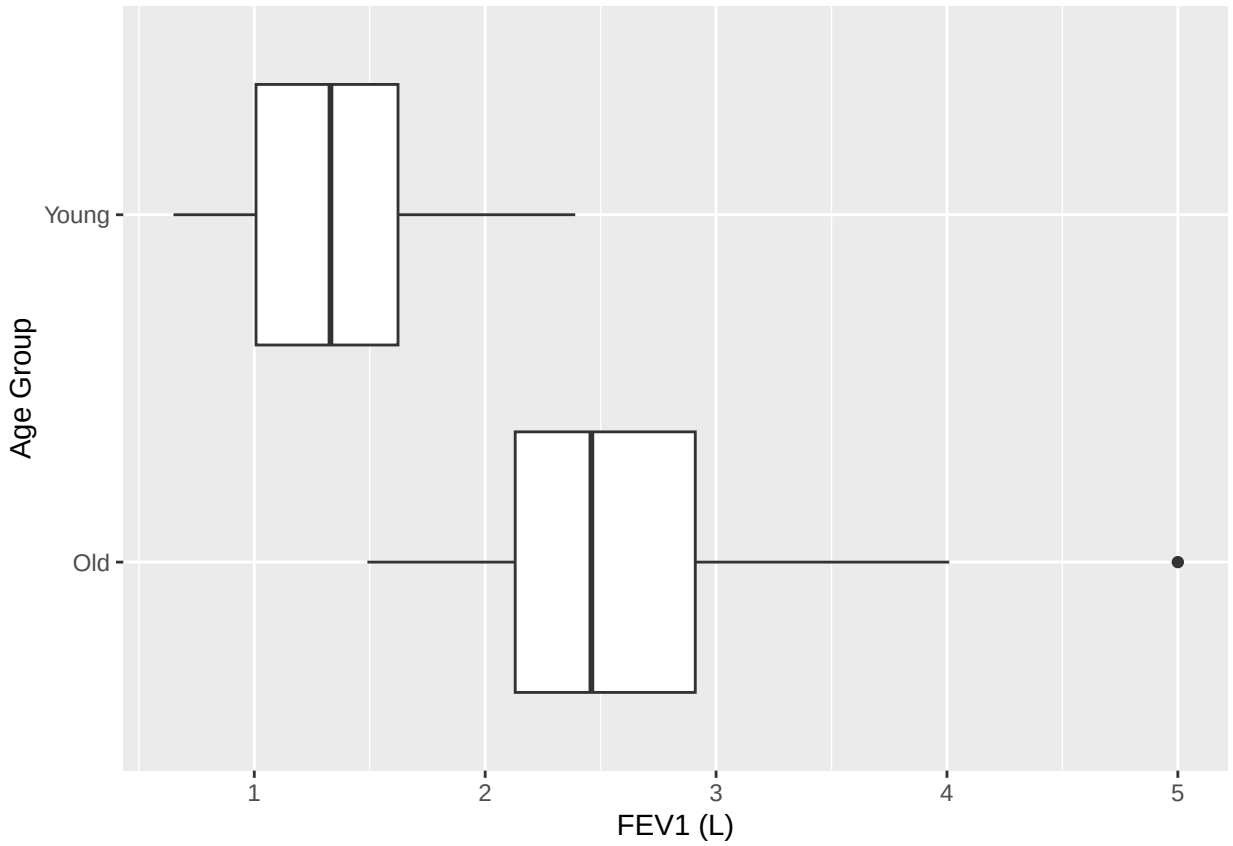
```
## Warning: Removed 17 rows containing missing values or values outside the scale range
## (`geom_point()`).
```

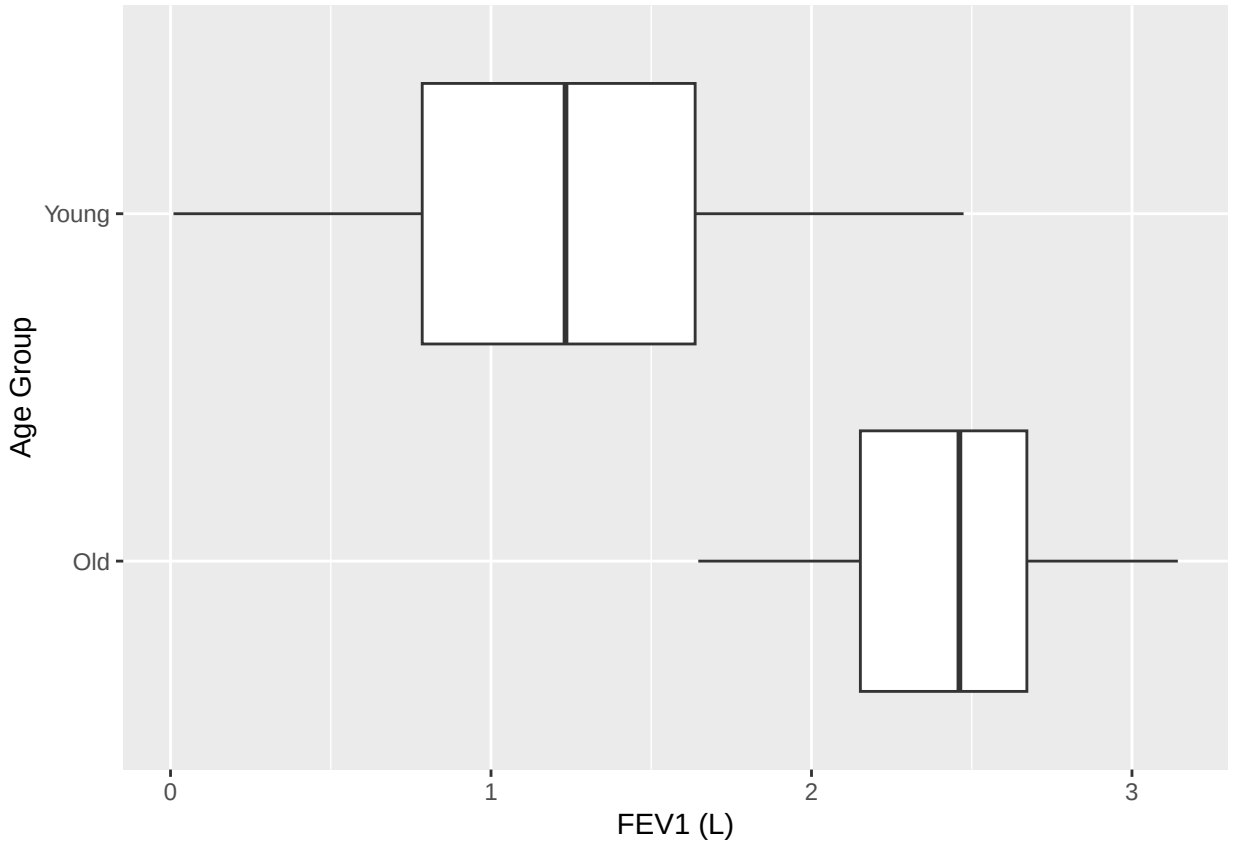





```
## Function for supporting premise statement 2
premise_2 <- function(df){
  real_plot <- ggplot(df) +
  geom_boxplot(aes(x = fev1, y = group)) +
  labs(x = "FEV1 (L)", y = "Age Group")

  fake_data <- tibble(
    x = c(rnorm(75, 1.3, .5), rnorm(75, 2.5, .35)),
    y = c(rep("Young", 75), rep("Old", 75))
  )
  fake_plot <- ggplot(fake_data) +
  geom_boxplot(aes(x = x, y = y)) +
  labs(x = "FEV1 (L)", y = "Age Group")

  print(real_plot)
  print(fake_plot)
}

premise_2(df)
```

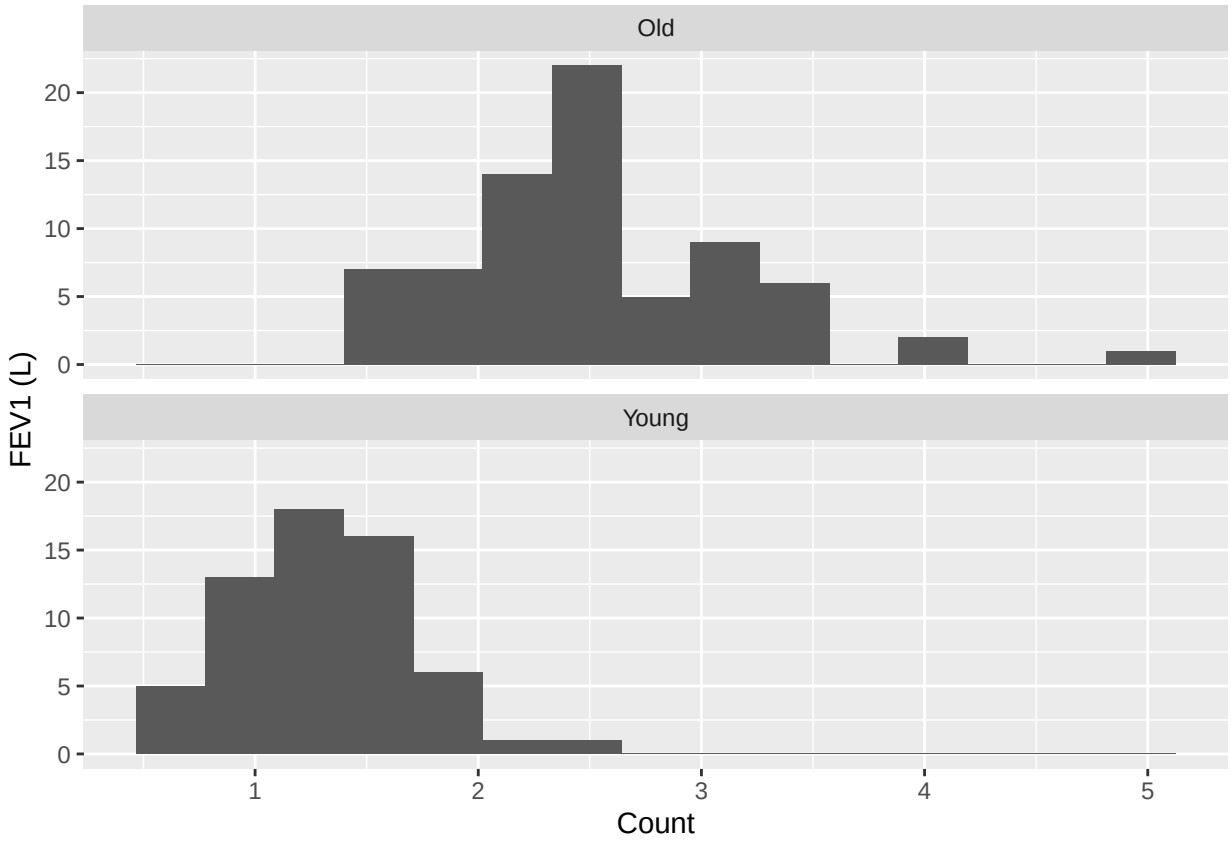
```
## Warning: Removed 17 rows containing non-finite outside the scale range
## (`stat_boxplot()`).
```

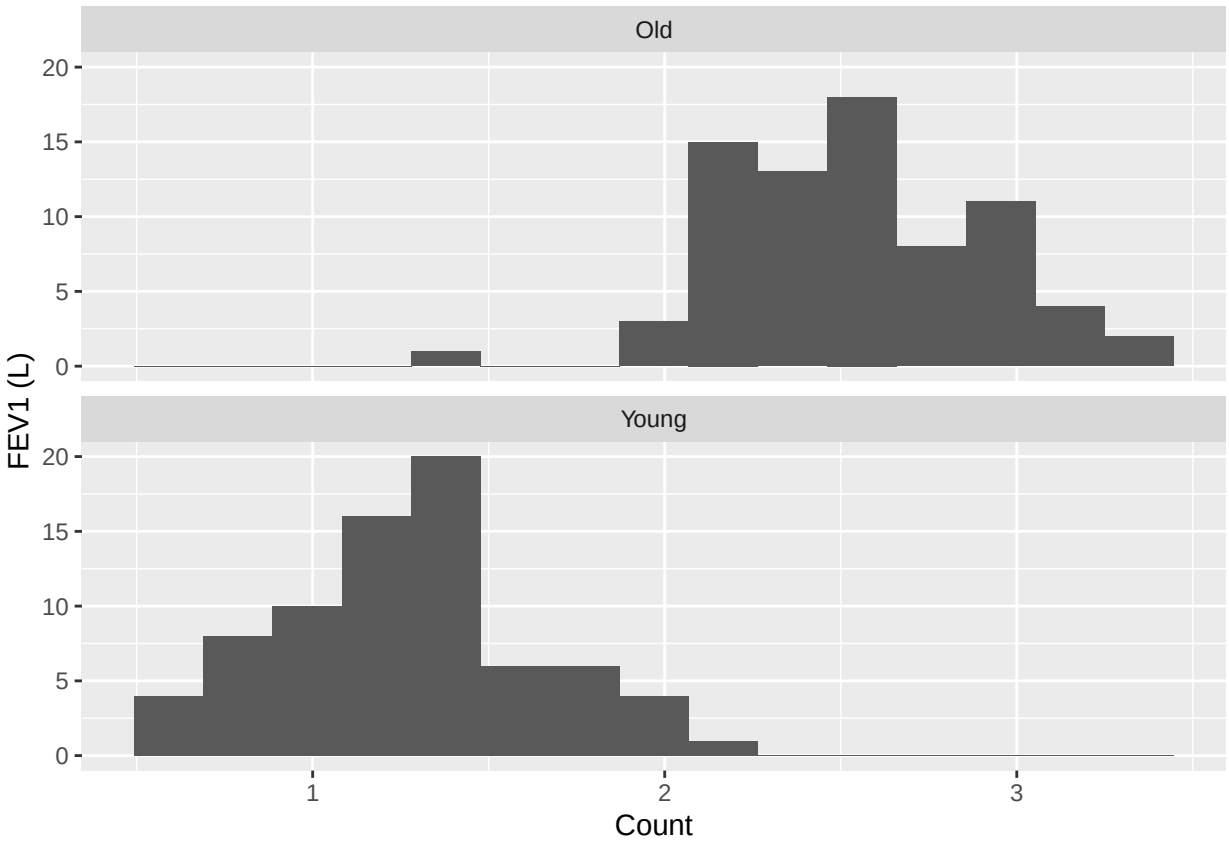

Young
Old
Age Group
1
2
3
4
5
FEV1 (L)


Young
Old
Age Group
0
1
2
3
FEV1 (L)

```r
## Function for supporting premise statement 3
premise_3 <- function(df){
  real_plot <- ggplot(df) +
  geom_histogram(aes(x = fev1), bins = 15) +
  facet_wrap(~ group, nrow = 2) +
  labs(x = 'Count', y = "FEV1 (L)")

  # generate data
  fake_data <- tibble(
    x = c(rnorm(75, 1.3, .35), rnorm(75, 2.5, .35)),
    y = c(rep("Young", 75), rep("Old", 75))
  )

  fake_plot <- ggplot(fake_data) +
  geom_histogram(aes(x = x), bins = 15) +
  facet_wrap(~ y, nrow = 2) +
  labs(x = 'Count', y = "FEV1 (L)")

  print(real_plot)
  print(fake_plot)
}

premise_3(df)
```

```
## Warning: Removed 17 rows containing non-finite outside the scale range
## (`stat_bin()`).
```





Execute each of your function and show that the produce the expected output.

Part 5

Describe one alternative to the primary statement "FEV1 values in children are higher in older children relative to younger children".

Create a fault tree for the alternative outcome describing how the alternative outcome could be realized in the data even if the primary statement were true.

Your fault tree should be created as a separate image and does not need to be created in R. Upload the image of the fault tree to Canvas.