Homework 2

Johnny Lydon

2024-02-09

Homework 2

```
library(tibble)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(readr)
library(tidyr)
library(ggplot2)
library(purrr)
library(cowplot)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v forcats   1.0.0     v stringr   1.5.1
## v lubridate 1.9.3

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter()    masks stats::filter()
## x dplyr::lag()       masks stats::lag()
## x lubridate::stamp() masks cowplot::stamp()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(knitr)
#Making sure all the packages are included.
```

Question 1

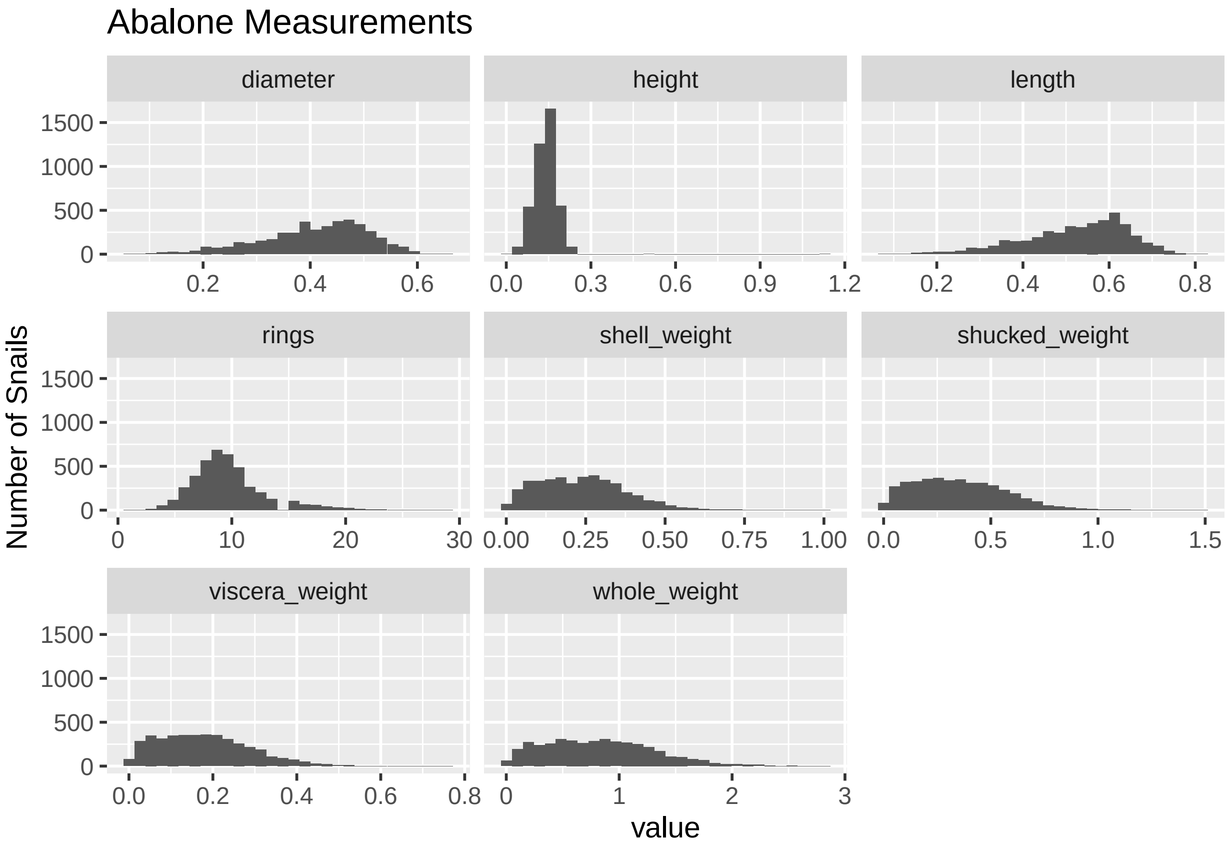
```r
#1.1
library(readr)
url <- "http://archive.ics.uci.edu/ml/machine-learning-databases/abalone/abalone.data"

abalone_col_names <- c(
  "sex",
  "length",
  "diameter",
  "height",
  "whole_weight",
  "shucked_weight",
  "viscera_weight",
  "shell_weight",
  "rings"
)

abalone <- read_csv(url, col_names = abalone_col_names)
```

```
## Rows: 4177 Columns: 9
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): sex
## dbl (8): length, diameter, height, whole_weight, shucked_weight, viscera_wei...
##
## 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.
```

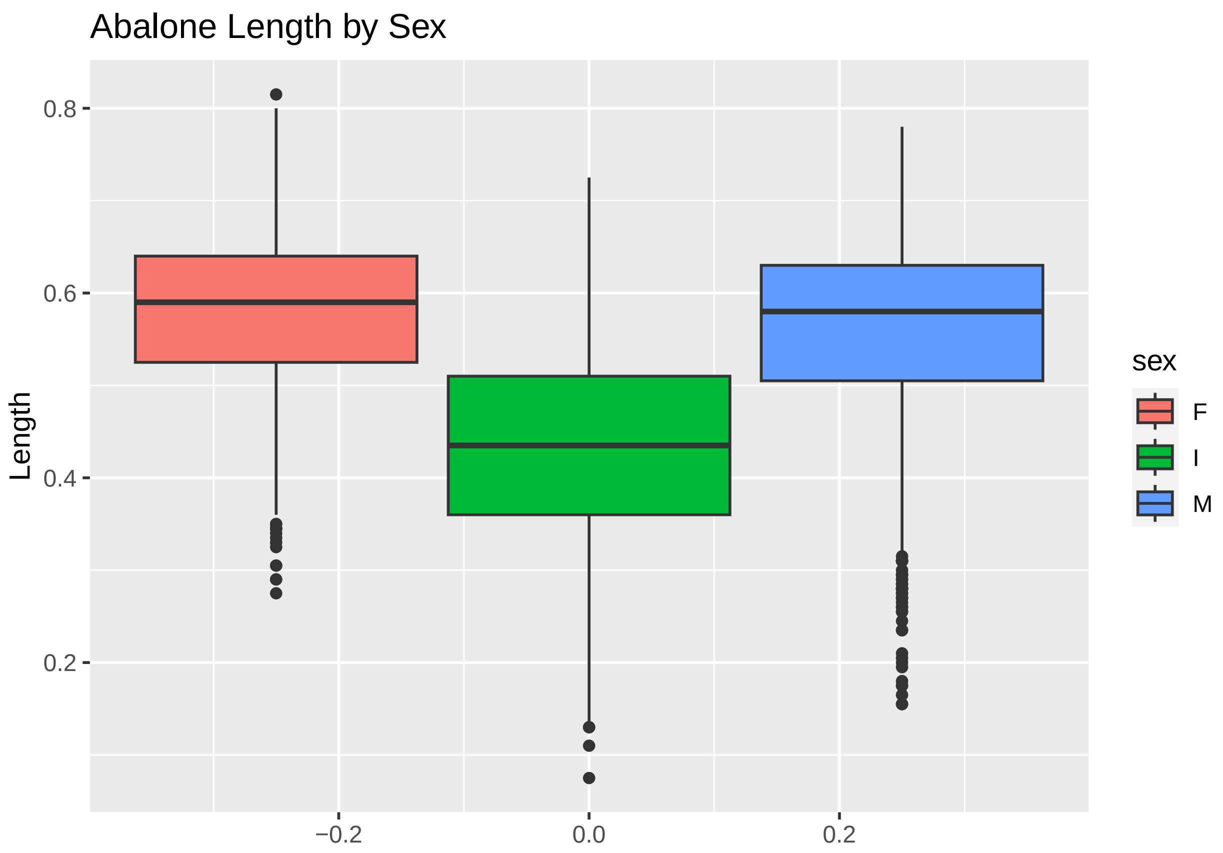
```r
#1.2
df <- abalone %>%
  drop_na()
```

```r
#1.3
df %>%
  select(where(is.numeric)) %>%
  gather() %>%
  ggplot() +
  geom_histogram(aes(value)) +
  facet_wrap(~key, scales = 'free_x')+
  labs(title = "Abalone Measurements") +
  ylab("Number of Snails")
```

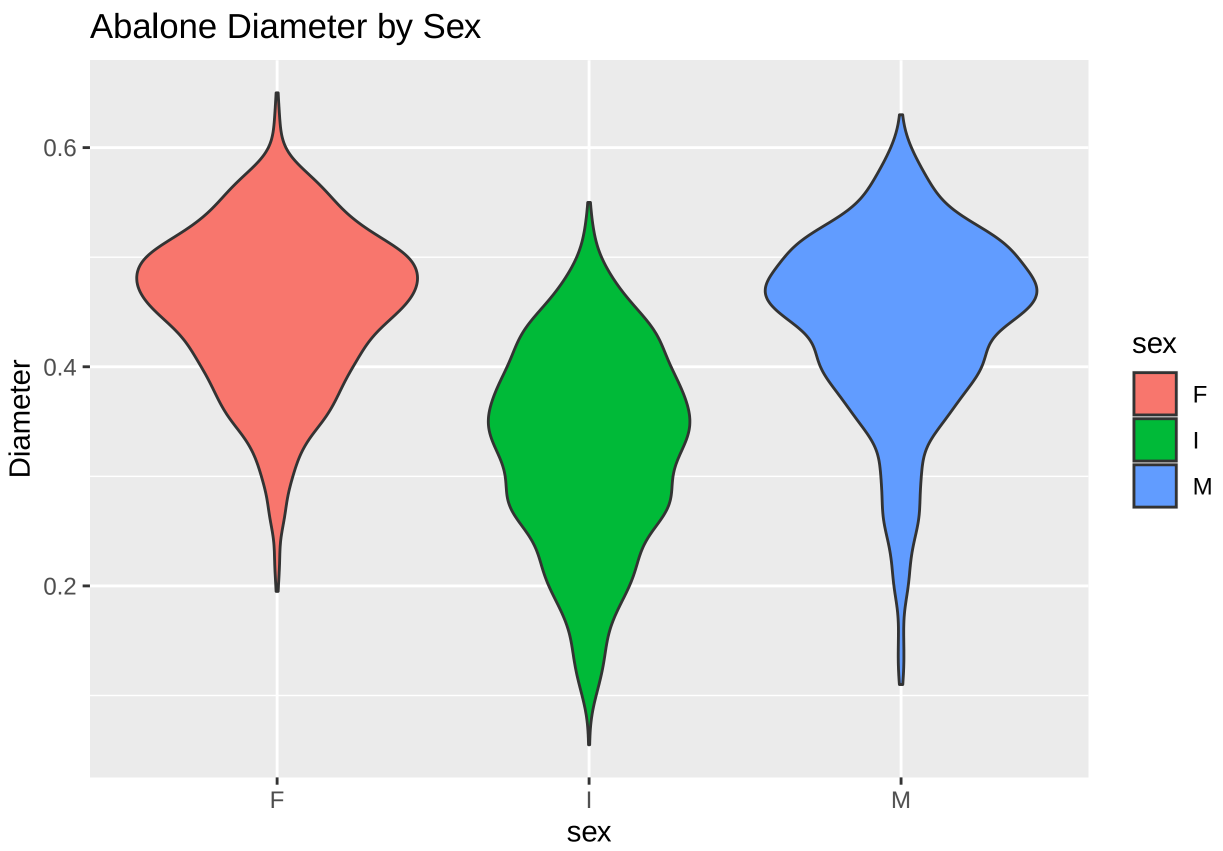
```
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
```



```r
#1.4
abaloneplot <- ggplot(df)

abaloneplot + geom_boxplot(aes(y = length, fill = sex)) +
  labs(title = "Abalone Length by Sex") +
  ylab("Length")
```

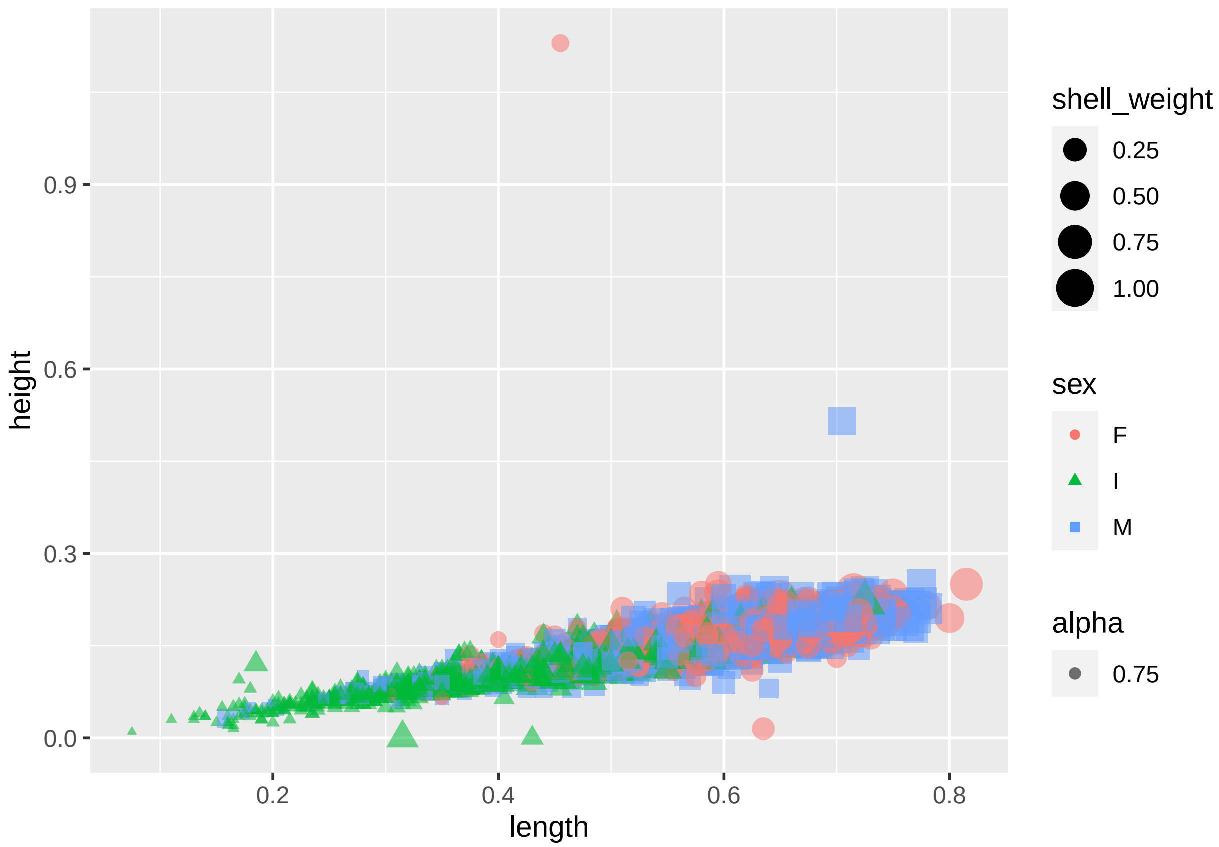


```
abaloneplot + geom_violin(aes(y = diameter, x = sex, fill = sex)) +
  labs(title = "Abalone Diameter by Sex") +
  ylab("Diameter")
```



```
# Are there any notable differences in the physical appearences of abalones based on your analysis here

# I don't think so.
```

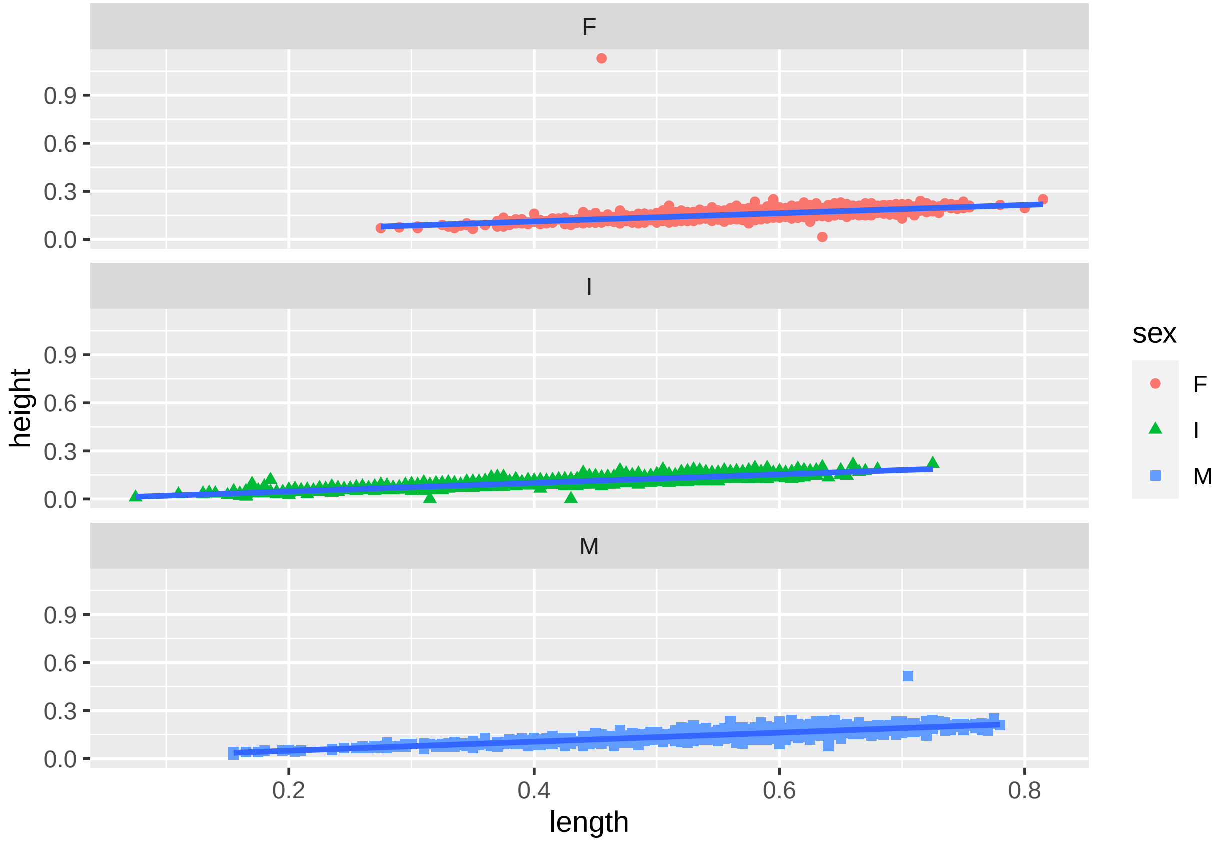
```
#1.5
abaloneplot + geom_point(aes(x = length, y = height, shape = sex, color = sex, group = sex, size = shel
```



```
# Are there any notable anomalies in the dataset?

# It looks like there are a handful of small anomalies that are shorter, and two very large anomalies t
```

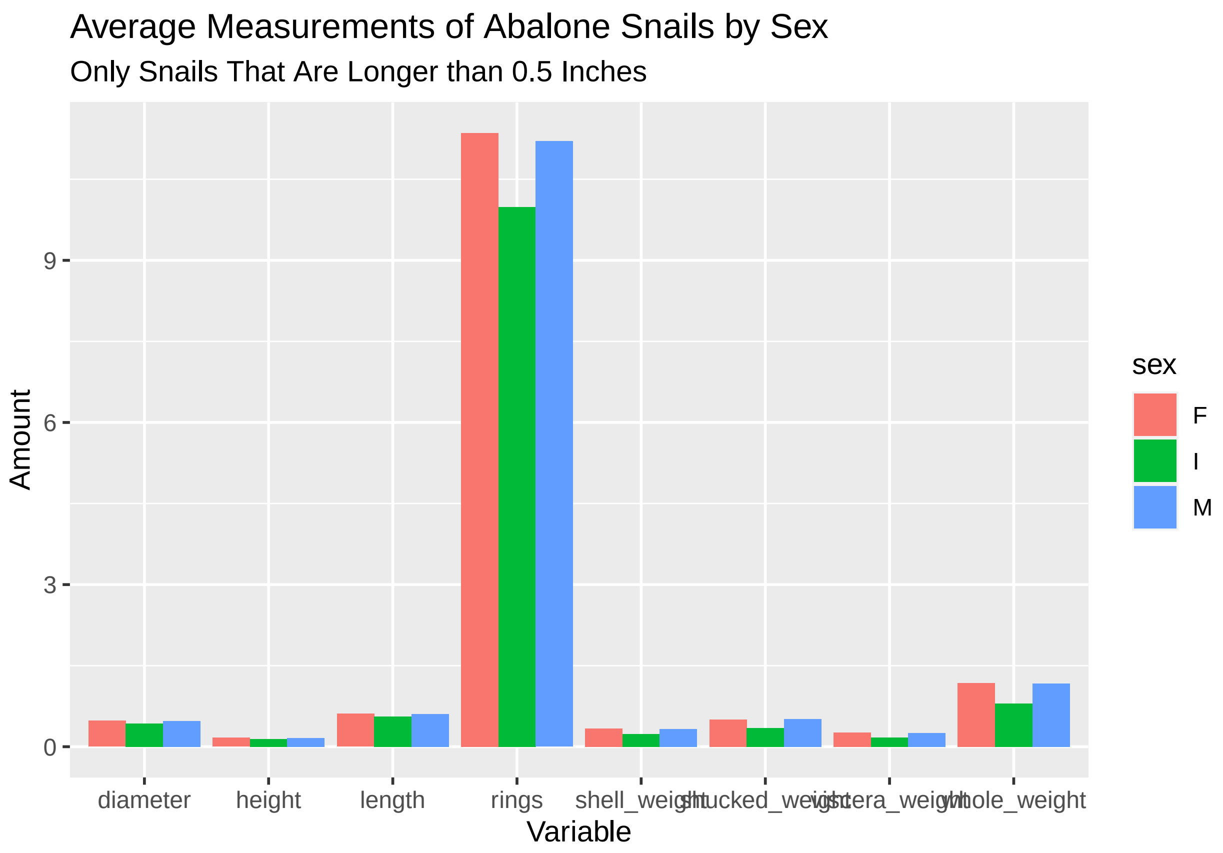
```
#1.6
abaloneplot + geom_point(aes(x = length, y = height, shape = sex, color = sex)) +
  geom_smooth(aes(x = length, y = height), method = lm) +
  facet_wrap(~sex, 3, 1)
```

```
## ‘geom_smooth()‘ using formula = ’y ~ x’
```



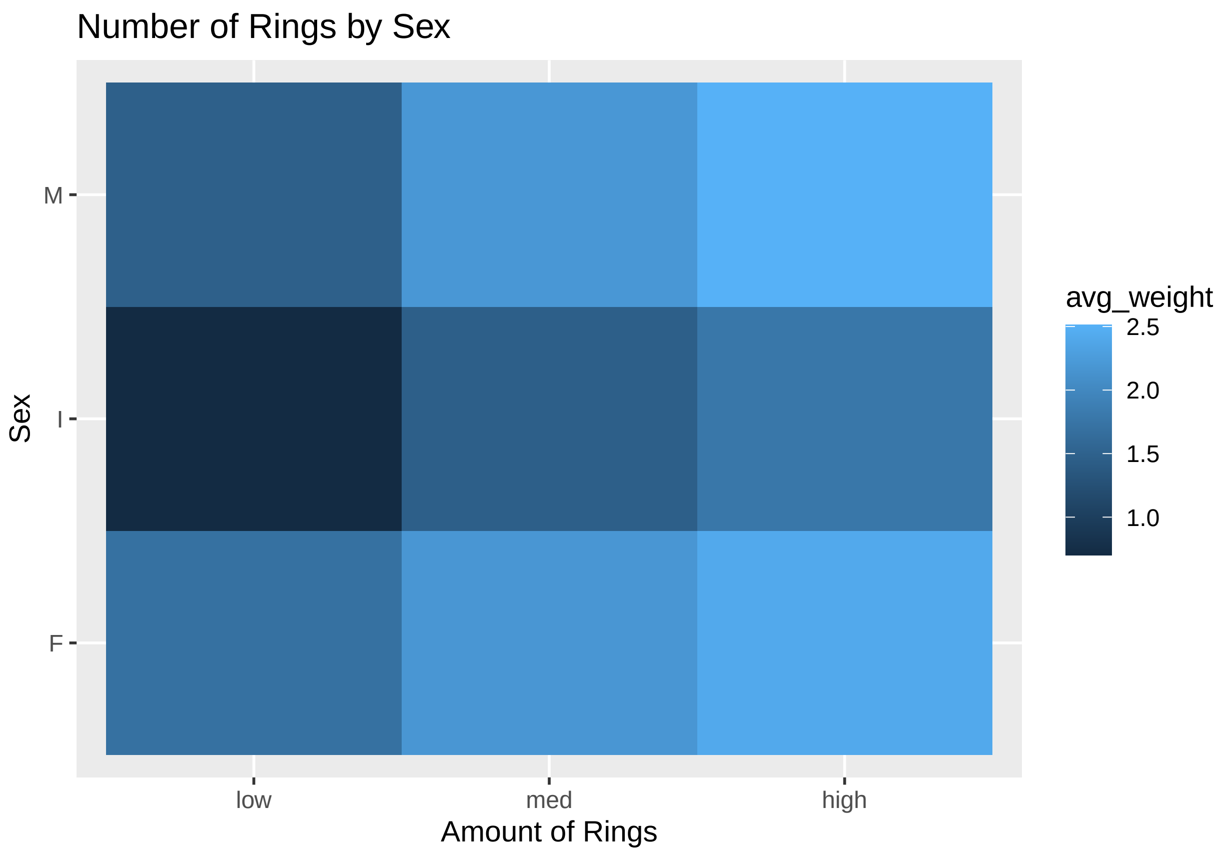
Question 2

```
#2.1
df %>%
  filter(length > 0.5) %>%
  group_by(sex) %>%
  summarise_if(is.numeric, mean) %>%
  pivot_longer(-c(sex)) %>%
  ggplot() +
  geom_bar(aes(x = name, y = value, fill = sex), stat = 'identity', position = 'dodge') +
  xlab("Variable") +
  ylab("Amount") +
  labs(title = "Average Measurements of Abalone Snails by Sex", subtitle = "Only Snails That Are Longer
```



```
#2.2
df %>%
  mutate(num_rings = case_when(rings < 10 ~ "low", rings > 20 ~ "high", TRUE ~ "med")) %>%
  group_by(num_rings, sex) %>%
  summarise(avg_weight = mean(whole_weight + shucked_weight + viscera_weight + shell_weight)) %>%
  mutate(num_rings = factor(num_rings, levels = c("low", "med", "high"))) %>%
  ggplot() +
  geom_tile(aes(x = num_rings, y = sex, fill = avg_weight))+
  labs(title = "Number of Rings by Sex") +
  xlab("Amount of Rings") +
  ylab("Sex")
```

```
## ‘summarise()‘ has grouped output by ’num_rings’. You can override using the
## ‘.groups‘ argument.
```



```
#2.3
df %>%
  select(where(is.numeric)) %>%
  cor() %>%
  round(digits = 2)
```

```
##                length diameter height whole_weight shucked_weight
## length           1.00     0.99   0.83         0.93           0.90
## diameter         0.99     1.00   0.83         0.93           0.89
## height           0.83     0.83   1.00         0.82           0.77
## whole_weight     0.93     0.93   0.82         1.00           0.97
## shucked_weight   0.90     0.89   0.77         0.97           1.00
## viscera_weight   0.90     0.90   0.80         0.97           0.93
## shell_weight     0.90     0.91   0.82         0.96           0.88
## rings            0.56     0.57   0.56         0.54           0.42
##                viscera_weight shell_weight rings
## length                   0.90         0.90  0.56
## diameter                 0.90         0.91  0.57
## height                   0.80         0.82  0.56
## whole_weight             0.97         0.96  0.54
## shucked_weight           0.93         0.88  0.42
## viscera_weight           1.00         0.91  0.50
## shell_weight             0.91         1.00  0.63
## rings                    0.50         0.63  1.00
```

```
#2.4
df %>%
  select(where(is.numeric)) %>%
  map2()

# I'm not sure how to do this one.
```

Question 3

```
#3.1
linreg <- lm(height ~ diameter, df)
summary(linreg)
```

```
##
## Call:
## lm(formula = height ~ diameter, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.15513 -0.01053 -0.00147  0.00852  1.00906
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.003803   0.001512  -2.515   0.0119 *
## diameter     0.351376   0.003602  97.544   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0231 on 4175 degrees of freedom
## Multiple R-squared:  0.695,  Adjusted R-squared:  0.695
## F-statistic:  9515 on 1 and 4175 DF,  p-value: < 2.2e-16
```

```
# The median and minimum are negative, which doesn't really make sense and I'm not sure if that's my fa
```

```
#3.2
plot(height ~ diameter, df, pch = 20)
abline(linreg, col = "orange")

# Is the linear model an appropriate fit for this relationship? Explain.

# I would definitely say it's an appropriate fit for the relationship. If you look at the graph, there'
```

```
#3.3
new_diameters <- c(
  0.15218946,
  0.48361548,
  0.58095513,
  0.07603687,
  0.50234599,
  0.83462092,
  0.95681938,
  0.92906875,
  0.94245437,
  0.01209518
)

new_data <- data.frame(diameter = new_diameters)

new_heights <- predict(linreg, new_data)

abline(v = new_diameters, col = "violet")
points(new_diameters, new_heights, col = "violet", pch = 20, cex = 2)

# I thought I did this right, but it won't work.
```