

R Refresher

Spencer Edwards

Question 1

```
L <- c(50, 55, 45, 30) #Create vector 1
W <- c(3.6, 3.7, 3.3, 2.8) #Create vector 2
MlutLW <- L*W #multiply vectors
print(MlutLW)
```

```
## [1] 180.0 203.5 148.5 84.0
```

Question 2

```
library(palmerpenguins) # Load the dataset
bl <- palmerpenguins::penguins$bill_length_mm #assign it a variable to make it
#easier to work with
meanbl <- mean(bl, na.rm = TRUE) #Calculate mean and discard NAs
maxbl <- max(bl, na.rm = TRUE)
minbl <- min(bl, na.rm = TRUE)
print(meanbl)
```

```
## [1] 43.92193
```

```
print(maxbl)
```

```
## [1] 59.6
```

```
print(minbl)
```

```
## [1] 32.1
```

Question 3

```
library(tidyverse) #Loads dplyr and ggplot
```

```
## -- Attaching packages -----
```

```
## v ggplot2 3.3.2    v purrr  0.3.4
## v tibble  3.0.3    v dplyr  1.0.2
## v tidyr   1.1.1    v stringr 1.4.0
## v readr   1.3.1    v forcats 0.5.0
```

```
## -- Conflicts ----- tid
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(plotly) #I thought I needed this to perform nls modeling,
```

```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
##
##   last_plot
```

```
## The following object is masked from 'package:stats':
##
##   filter
```

```
## The following object is masked from 'package:graphics':
##
##   layout
```

```
#turns out I didn't
```

```
soledata <- read.csv("Sole.Lengths.csv") #Load csv
tl <- soledata$TL #assign variables
sl <- soledata$SL
mean(tl)
```

```
## [1] 83.208
```

```
sd(tl)
```

```
## [1] 23.66646
```

```
mean(sl)
```

```
## [1] 71.304
```

```
sd(sl)
```

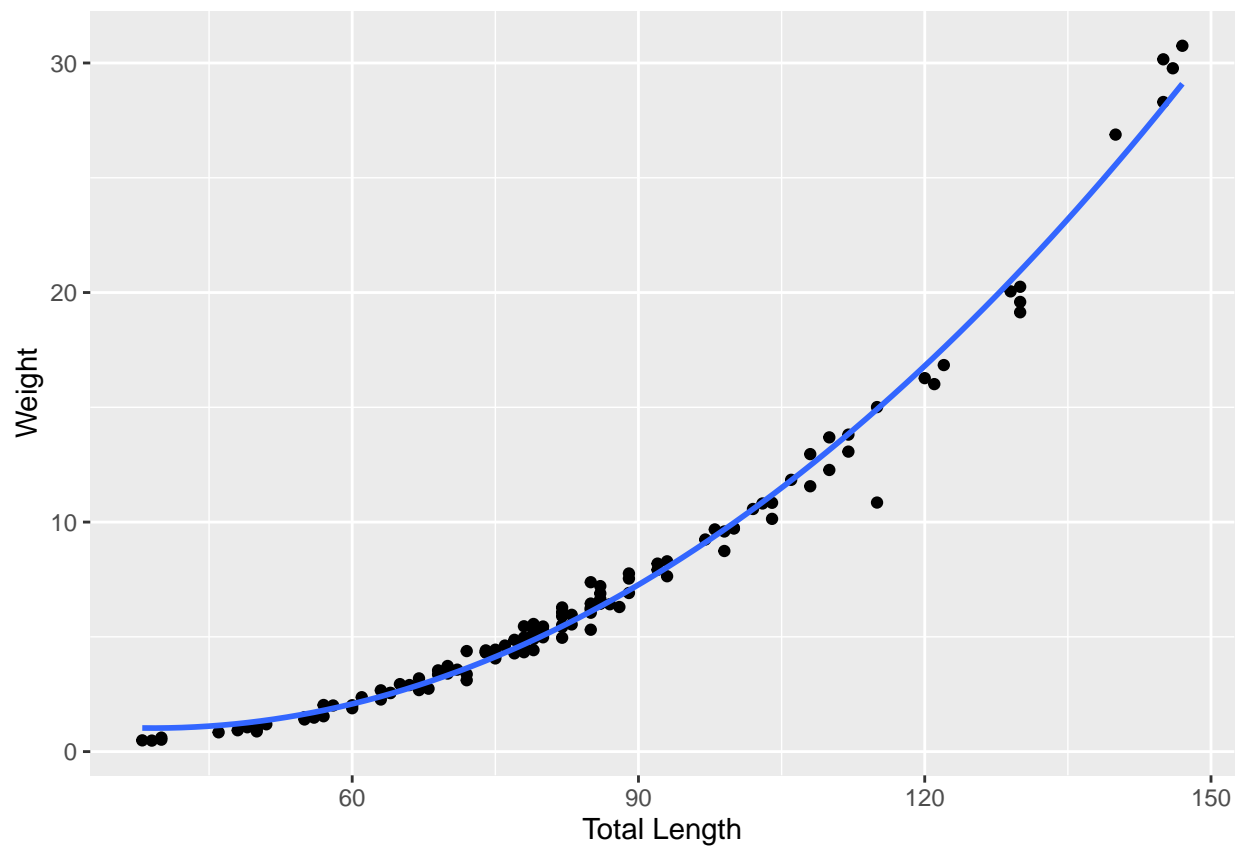
```
## [1] 20.52152
```

Question 4

Total length refers to the length of the whole fish from the mouth to the end of the tail (when the tail is pinched together), whereas standard length is from the mouth the end of the fleshy part of the body (excluding the tail entirely).

Question 5

```
soledata %>% #pipe in data
  ggplot(aes(x = TL, y = WT)) + geom_point() +
  geom_smooth(method = "lm", formula= (y ~ poly(x, 2)), se=FALSE) +
  xlab("Total Length") + ylab("Weight") #this tells ggplot to set the aesthetic,
```



```
#make points and a polynomial regression
```

Question 6

The points form a quadratic function, and are very precise (they're close together).

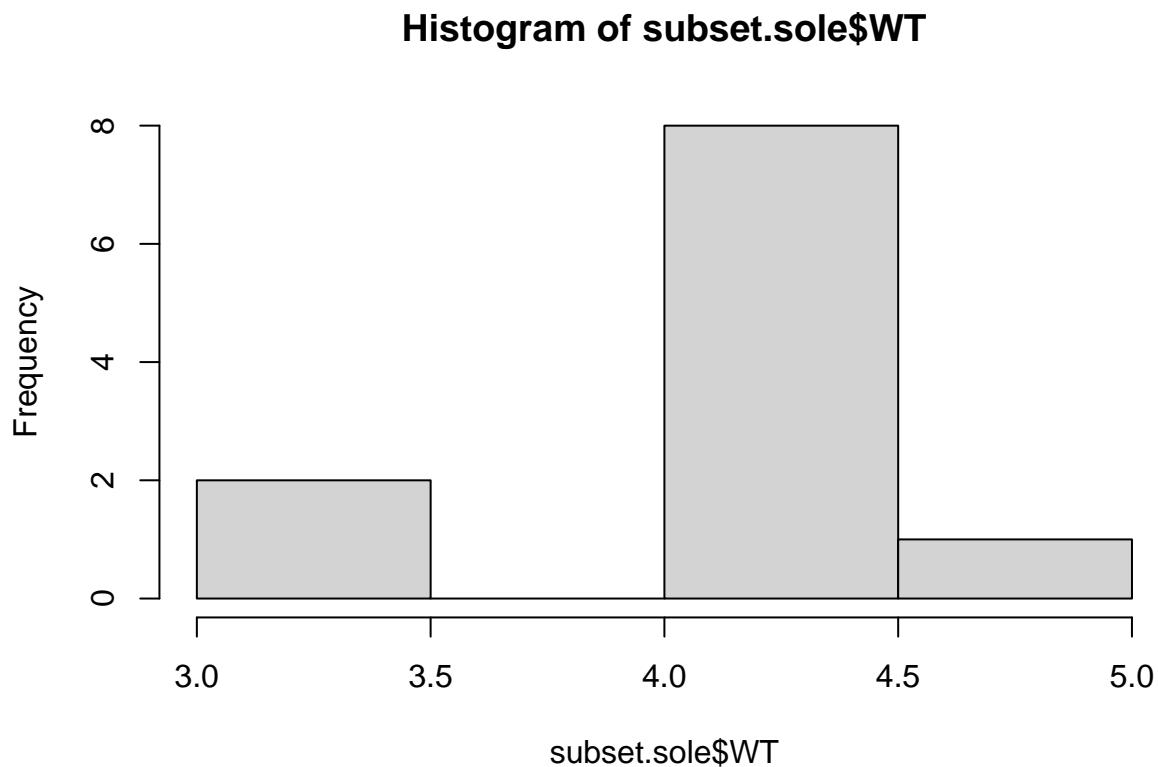
Question 7

```
subset.sole <- soledata %>% #pipe data and select weight
  select(WT) %>% #pipe again and select a subset from 35 to 60
  slice(40:50)
mean(subset.sole$WT)
```

```
## [1] 4.136364
```

Question 8

```
hist(subset.sole$WT)
```



The data appear to be right skewed, thus it's not a normal distribution.

Question 9

1/5. I had some difficulty when plotting because I thought it was exponential at first, so I tried using (method = nls, formula = $y \sim \exp(x)$) however it was actually quadratic. I also had to remind myself how to use the dplyr() function slice(). But overall, it was very simple.

Question 10

Yes, I feel very confident in my ability. I hope I am able to help other people with their labs as well.