R Bridge Assignment

2023-07-25

Introduction By Frederick Jones

This analysis tries to find whether the male or female report their height accurately. This analysis use the data named as 'davis' which is available freely at the link http://vincentarelbundock.github.io/Rdatasets/davis. This data contains six columns one X which is just an index, 'weight' and 'height' are the measured weights and heights of participants. 'repwt' and 'repht' are reported weights and reported heights of participants. ## Problem Statement ### Who is more accurate about reporting weight and height, an adult male or an adult female? This analysis will answer this question based on data collected. ## Import data

```
dataset <- read.csv('davis.csv')
head(dataset)
```

```
##   X sex weight height repwt repht
## 1 1   M     77    182    77   180
## 2 2   F     58    161    51   159
## 3 3   F     53    161    54   158
## 4 4   M     68    177    70   175
## 5 5   F     59    157    59   155
## 6 6   M     76    170    76   165
```

Qyestion 1

Data Exploration

Before calculating anything, we must check if there is any missing values in dataset. The function anyNA do this job efficiently.

```
print(sapply(dataset, anyNA))
```

```
##      X    sex weight height  repwt  repht
##  FALSE  FALSE  FALSE  FALSE   TRUE   TRUE
```

It can be seen that the columns 'repwt' and 'repht' have missing values. We can replace the missing values by average of the rest of the values in the column or we can just remove the rows containing the missing values. I think for this dataset which consist of reported weight and reported height, it will be good if the missing values are replaced by average of the other values in column.

Handling missing data

```
dataset$repwt <- ifelse(is.na(dataset$repwt),
                        ave(dataset$repwt, FUN= function(x) mean(x, na.rm = TRUE)),
                        dataset$repwt)
dataset$repht <- ifelse(is.na(dataset$repht),
                        ave(dataset$repht, FUN= function(x) mean(x, na.rm = TRUE)),
                        dataset$repht)
```

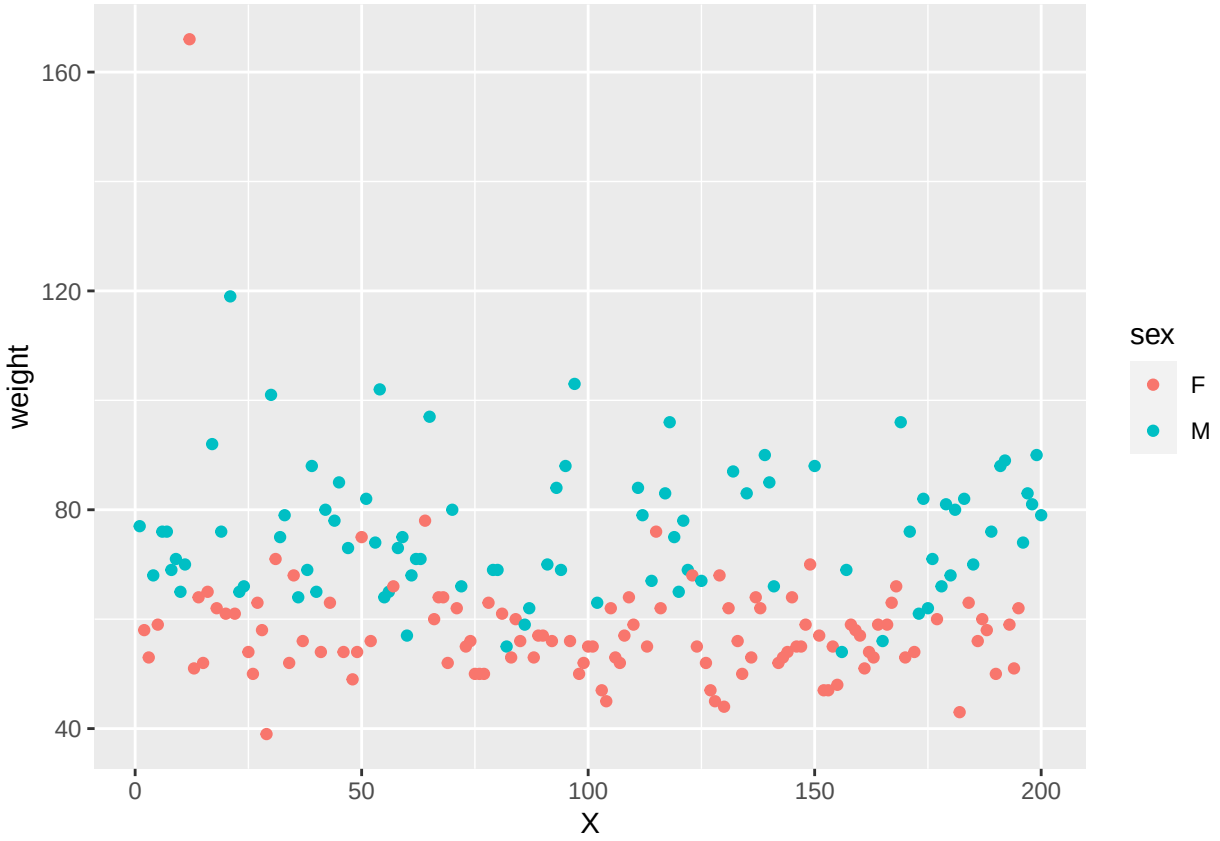
Let us check if there is any missing value in our dataset

```
print(sapply(dataset, anyNA))
```

```
##      X    sex weight height  repwt  repht
##  FALSE  FALSE  FALSE  FALSE  FALSE  FALSE
```

It can be seen no missing values in the modified dataset.Now as the next step, it will be necessary to check if there is any outlier in the data. The easiest method is by creating a scatter-plot or box-plot and check if there is any outlier in the data. Let's create a scatter-plot.

```
library(ggplot2)
ggplot(data = dataset)+
  geom_point( mapping=aes (x=X, y=weight, color=sex))
```



The scatter plot shows that ther is an outlier in the data related to female. The outlier containing weight more than 160 is an outlier in this data. So, we have to remove it from the data. Let's remove this extreme entry, it might be typing error.

```r
dataset<- subset(dataset, dataset$weight<160)
head(dataset)
```

```
##   X sex weight height repwt repht
## 1 1   M     77    182    77   180
## 2 2   F     58    161    51   159
## 3 3   F     53    161    54   158
## 4 4   M     68    177    70   175
## 5 5   F     59    157    59   155
## 6 6   M     76    170    76   165
```

In the next step, the mean of measured as well as reported weight and height of male and female can be found as follows:

```r
aggregate(dataset[,3:6], list(dataset$sex), FUN = mean)
```

```
##   Group.1   weight   height   repwt    repht
## 1       F 56.89189 164.7027 57.6293 162.8150
## 2       M 75.89773 178.0114 75.8152 175.7271
```

It can be seen that the mean of measured weight and measured height of 199 females differ by small value from mean of reported weight and reported height.For example mean measured weight of females is 56.892 while mean reported weight is 57.629. Similarly mean measured height of females is 164.703 cm and mean of reported heights is 162.82 cm.
On the contrary, mean measured height of males is 178.01 cm while mean of reported heights is 175.73 which is less than the measured height. Mean measured weight of surveyed males is 75.898 and mean reported weight of male is 75.815.

Median of data based on gender

```r
print(aggregate(dataset[,3:6], list(dataset$sex), FUN = median))
```

```
##   Group.1 weight height repwt repht
## 1       F     56    165    57   163
## 2       M     75    178    73   175
```

Minimum values in the data

```r
aggregate(dataset[,3:6], list(dataset$sex), FUN = min)
```

```
##   Group.1 weight height repwt repht
## 1       F     39    148    41   148
## 2       M     54    163    56   161
```

Maximum values in the data

```r
aggregate(dataset[,3:6], list(dataset$sex), FUN = max)
```

```
##   Group.1 weight height repwt repht
## 1       F     78    178    77   176
## 2       M    119    197   124   200
```

Question 2. Data wrangling.

The data contains X which is just an index number and we don't need it. So, we can take subset of data as follows.

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v lubridate 1.9.2     v tibble    3.2.1
## v purrr     1.0.1     v tidyr     1.3.0
## -- 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<- dataset[, 2:6]
df_male = filter(df, sex=='M')  #df_male contains data only for males
df_female = filter(df, sex=='F')
```

Now we can have summary for males and females separately

```
summary(df_male)
```

```
##      sex                weight           height        repwt
##  Length:88          Min.   : 54.00   Min.   :163   Min.   : 56.00
##  Class :character   1st Qu.: 67.75   1st Qu.:173   1st Qu.: 67.00
##  Mode  :character   Median : 75.00   Median :178   Median : 73.00
##                     Mean   : 75.90   Mean   :178   Mean   : 75.82
##                     3rd Qu.: 83.00   3rd Qu.:183   3rd Qu.: 82.25
##                     Max.   :119.00   Max.   :197   Max.   :124.00
##      repht
##  Min.   :161.0
##  1st Qu.:170.0
##  Median :175.0
##  Mean   :175.7
##  3rd Qu.:180.0
##  Max.   :200.0
```

```
summary(df_female)
```

```
##      sex                weight           height          repwt
##  Length:111         Min.   :39.00   Min.   :148.0   Min.   :41.00
##  Class :character   1st Qu.:52.50   1st Qu.:161.5   1st Qu.:53.00
##  Mode  :character   Median :56.00   Median :165.0   Median :57.00
##                     Mean   :56.89   Mean   :164.7   Mean   :57.63
##                     3rd Qu.:62.00   3rd Qu.:169.0   3rd Qu.:62.50
##                     Max.   :78.00   Max.   :178.0   Max.   :77.00
##      repht
##  Min.   :148.0
```

```
##  1st Qu.:159.5
##  Median :163.0
##  Mean   :162.8
##  3rd Qu.:168.0
##  Max.   :176.0
```

```
df_male<- rename(df_male, Weight=weight, Height = height, Reported_Wt = repwt, Reported_Ht= repht)
df_female<- rename(df_female, Weight=weight, Height = height, Reported_Wt = repwt, Reported_Ht=repht)
head(df_female)
```

```
##   sex Weight Height Reported_Wt Reported_Ht
## 1   F     58    161          51         159
## 2   F     53    161          54         158
## 3   F     59    157          59         155
## 4   F     51    161          52         158
## 5   F     64    168          64         165
## 6   F     52    163          57         160
```

Now two additional columns will be added to dataset one is |repwt-weight| and other is |repht=height|

```
df_male$Abs_diff_Wt = abs(df_male$Reported_Wt-df_male$Weight)
df_female$Abs_diff_Ht = abs(df_female$Reported_Ht-df_female$Height)
df_male$Abs_diff_Ht = abs(df_male$Reported_Ht-df_male$Height)
df_female$Abs_diff_Wt = abs(df_female$Reported_Wt-df_female$Weight)
mean_abs_error_wt_male = mean(df_male$Abs_diff_Wt)
sprintf("Mean absolute error in Weight of male is %.2f", mean_abs_error_wt_male)
```

```
## [1] "Mean absolute error in Weight of male is 2.44"
```

```
mean_abs_error_wt_female = mean(df_female$Abs_diff_Wt)
sprintf("Mean absulte error in Weight of female is %.2f", mean_abs_error_wt_female)
```

```
## [1] "Mean absulte error in Weight of female is 2.43"
```

```
mean_abs_error_ht_male = mean(df_male$Abs_diff_Ht)
sprintf("Mean absolute error in height of male is %.2f", mean_abs_error_ht_male)
```

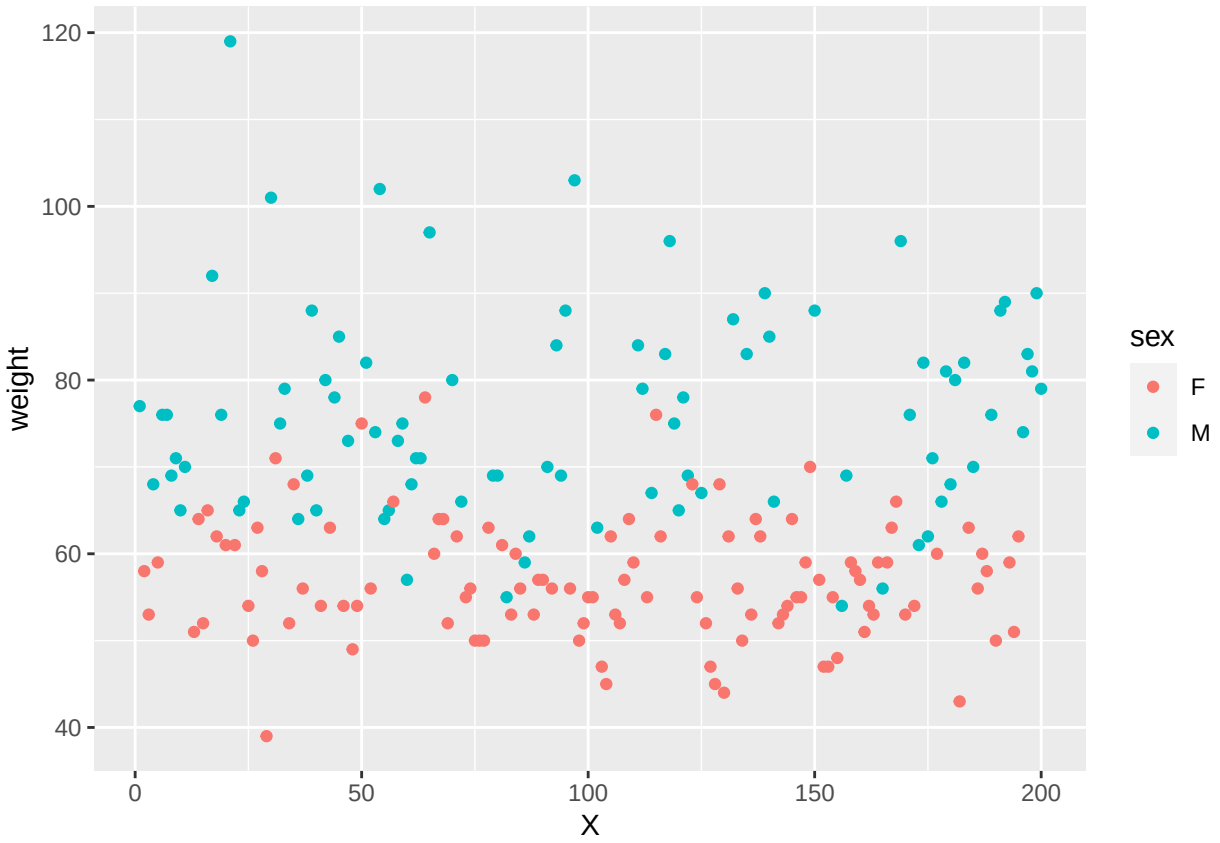
```
## [1] "Mean absolute error in height of male is 2.67"
```

```
mean_abs_error_ht_female = mean(df_female$Abs_diff_Ht)
sprintf("Mean absulte error in height of female is %.2f", mean_abs_error_ht_female)
```

```
## [1] "Mean absulte error in height of female is 2.78"
```

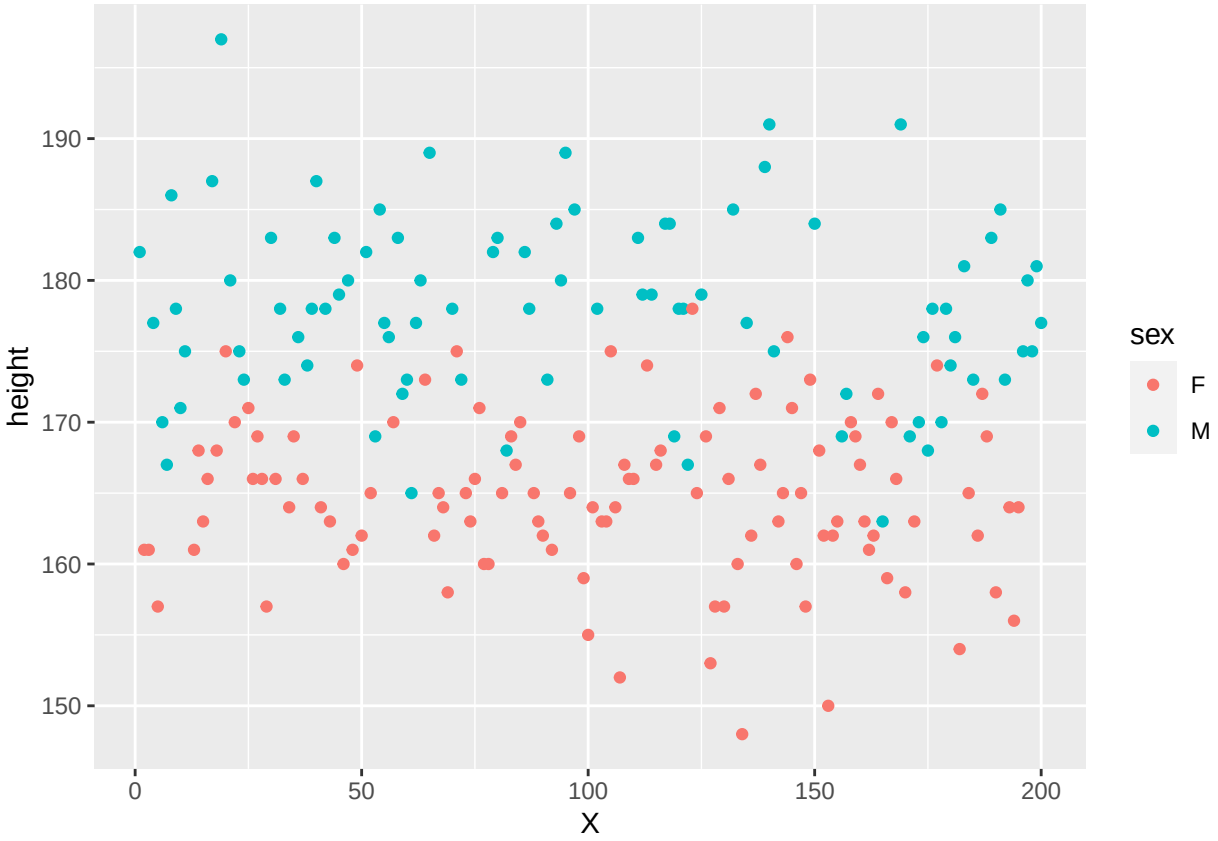
Question 3. Graphics or Visualization

```
library(ggplot2)
ggplot(data = dataset)+
  geom_point( mapping=aes (x=X, y=weight, color=sex))
```

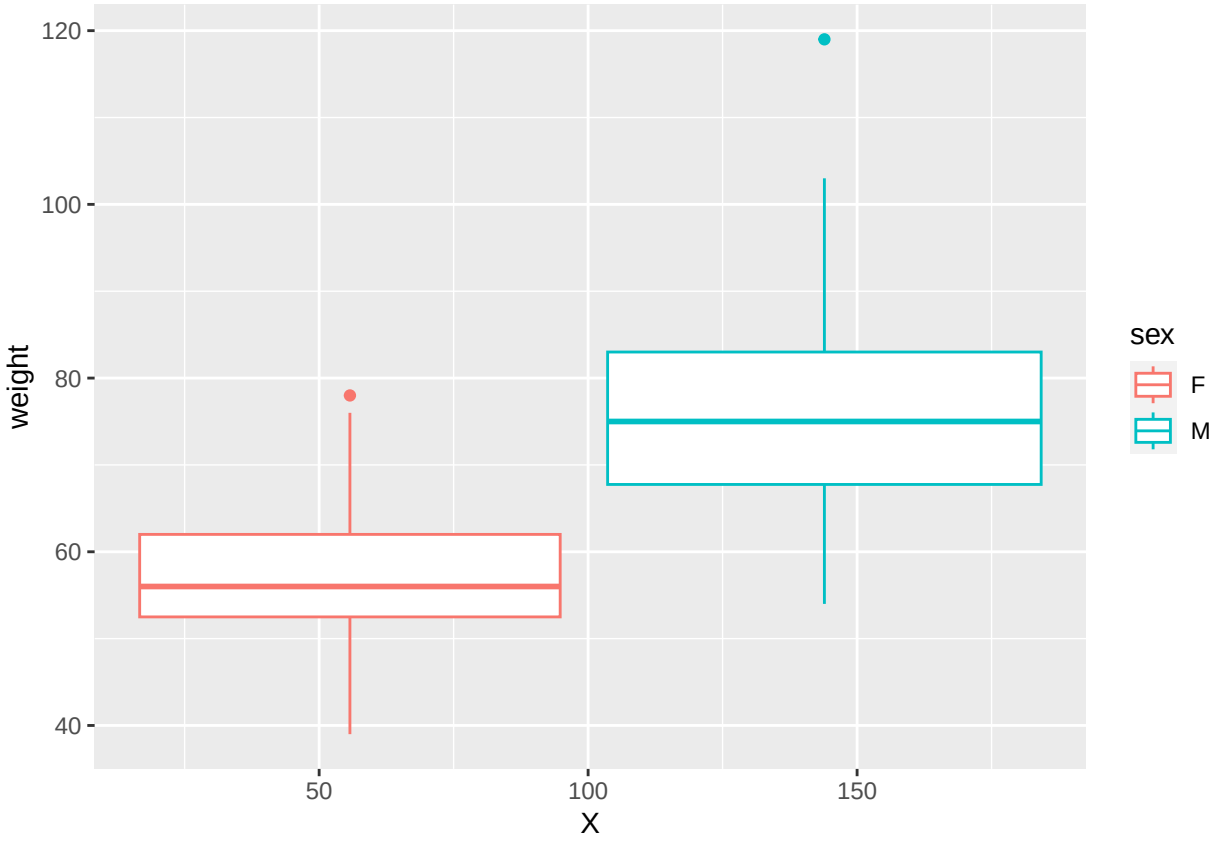


It shows an outlier in the df_female which must be addressed. The outlier might be due to typing error. So, on close observation the data, it was found that the in 4th row, there is an outlier. So, 4th row from the female data can be removed.

```
ggplot(data = dataset)+
  geom_point ( mapping=aes (x= X, y=height, color=sex))
```

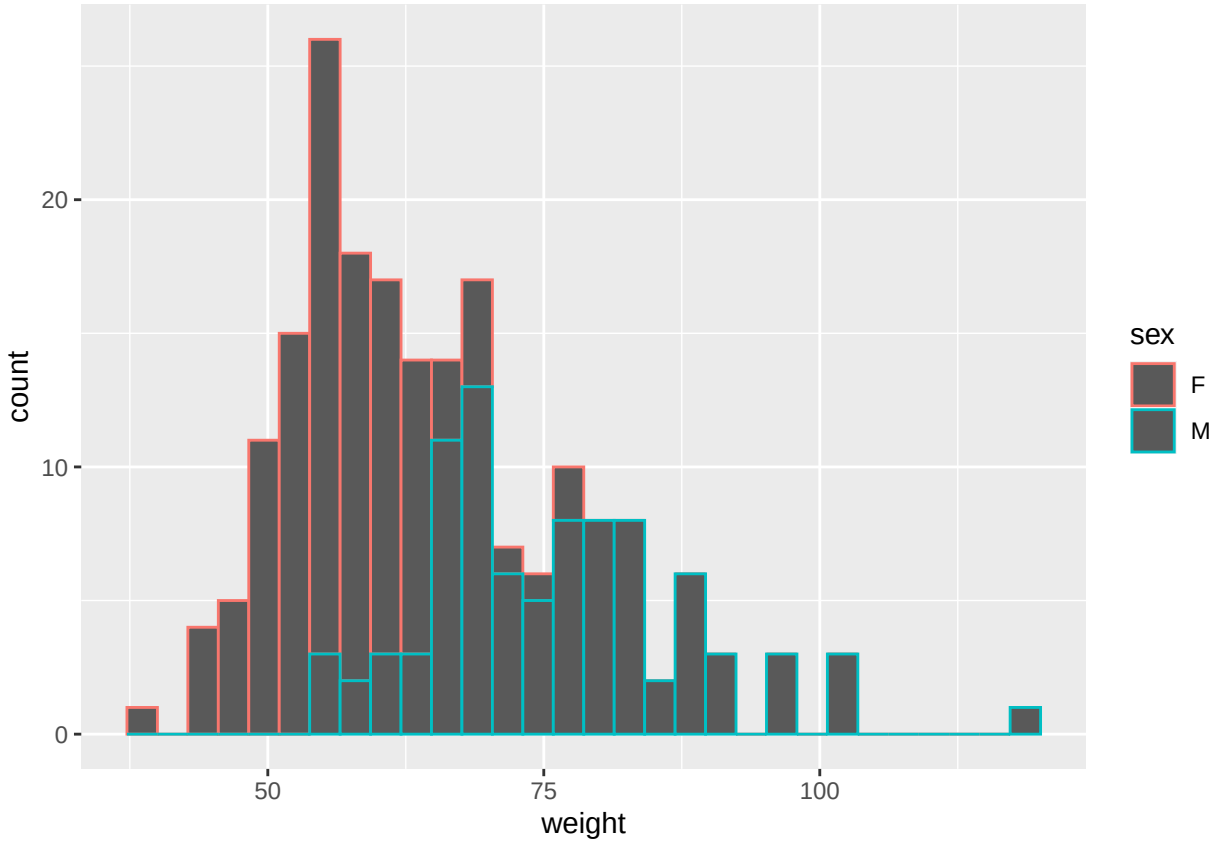



```
ggplot(data = dataset)+
  geom_boxplot (mapping = aes (x=X, y= weight, color = sex))
```




```
ggplot(data = dataset)+
  geom_histogram (mapping = aes (x= weight, color = sex))
```

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



Question 4. ## Conclusion: It does not appear that either male or female are more accurate in reporting their weight and height. The mean difference of absolute errors in reporting for both males and females are almost the same. Their is not much difference between error values.

Question 5. Bonus

Reading data from github

```
library(readr)
df2 <- read_csv("https://raw.githubusercontent.com/jewelercart/R/main/Davis.csv")
```

```
## New names:
## Rows: 200 Columns: 6
## -- Column specification
## -------------------------------------------------------- Delimiter: "," chr
## (1): sex dbl (5): ...1, weight, height, repwt, repht
## 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.
## * `` -> `...1`
```

```
summary(df2)
```

```
##       ...1            sex                weight          height
##  Min.   :  1.00   Length:200         Min.   : 39.0   Min.   : 57.0
##  1st Qu.: 50.75   Class :character   1st Qu.: 55.0   1st Qu.:164.0
```

```
##  Median :100.50   Mode  :character   Median : 63.0   Median :169.5
##  Mean   :100.50                      Mean   : 65.8   Mean   :170.0
##  3rd Qu.:150.25                      3rd Qu.: 74.0   3rd Qu.:177.2
##  Max.   :200.00                      Max.   :166.0   Max.   :197.0
##
##      repwt            repht
##  Min.   : 41.00   Min.   :148.0
##  1st Qu.: 55.00   1st Qu.:160.5
##  Median : 63.00   Median :168.0
##  Mean   : 65.62   Mean   :168.5
##  3rd Qu.: 73.50   3rd Qu.:175.0
##  Max.   :124.00   Max.   :200.0
##  NA's   :17       NA's   :17
```