Project 1: Statistical Analysis and Visualization of a Dataset Using R

Michael Massaad

2024-02-06

Data Exploration and Descriptive Statistics

Task 1

In this project, I will be using the height(in cm) and weight(in kg) dataset found on Kaggle (https://www.kaggle.com/datasets/galserge/weight-and-height-from-nhanes) which observes 8388 people (males and females) ages 20 to 150 years old . I will only be using the weight and standing height columns of the dataset for this project. Through this project, I can hopefully gain a better understanding on the relationship between the height and the weight and be able to visualize this dataset clearly.

Cleaning and presenting general structure information about dataset

Make sure to import the dataset through the RStudio environment or put the filepath into the quotation for the read.cvs()

```
NHANES.Weight.and.Height <-
  read.csv("/Users/mike2/OneDrive/Documents/SECOND/MAT2375/NHANES\ Weight\ and\ Height.csv")
dataset = NHANES.Weight.and.Height[, c(2, 3)]
dim(dataset)
```

```
## [1] 8388    2
```

```
str(dataset)
```

```
## 'data.frame':    8388 obs. of  2 variables:
##  $ Weight..kg.         : num  97.1 98.8 74.3 103.7 83.3 ...
##  $ Standing.Height..cm.: num  160 182 184 185 177 ...
```

```
head(dataset, 10)
```

```
##    Weight..kg. Standing.Height..cm.
## 1         97.1                160.2
## 2         98.8                182.3
## 3         74.3                184.2
## 4        103.7                185.3
## 5         83.3                177.1
## 6          91.1                 152.7
## 7          72.6                 158.4
## 8          73.0                 161.2
## 9          81.4                 161.3
## 10         86.0                 167.8
```

```
summary(dataset)
```

```
##   Weight..kg.     Standing.Height..cm.
##  Min.   : 32.60   Min.   :131.1
##  1st Qu.: 67.60   1st Qu.:159.1
##  Median : 79.90   Median :166.2
##  Mean   : 83.67   Mean   :166.6
##  3rd Qu.: 95.80   3rd Qu.:173.9
##  Max.   :254.30   Max.   :199.6
```

Task 2

In observing my data, since we are referring to the measurements of height and weight, which both can take values from an interval of positive real numbers (where $0 < height < infinity$ and $0 < weight < infinity$), therefore my dataset contains continuous data.

Summarizing data into a chart showing data value, frequency, relative frequency and cumulative relative frequency

1. Computing and summarizing data for height

```
Height_data = dataset$Standing.Height..cm.
frequencyH = table(Height_data)
rfrequencyH = prop.table(frequencyH)
crfrequencyH = cumsum(rfrequencyH)
dataH = data.frame(
  height = as.numeric(names(frequencyH)),
  Hfrequency = as.numeric(frequencyH),
  relative_frequencyH = as.numeric(rfrequencyH),
  cumulative_relative_fH = as.numeric(crfrequencyH)
)
head(dataH, 10) # visualizes first 10 rows of the table
```

```
##    height Hfrequency relative_frequencyH cumulative_relative_fH
## 1   131.1          1        0.0001192179           0.0001192179
## 2   135.3          1        0.0001192179           0.0002384359
## 3   138.3          1        0.0001192179           0.0003576538
## 4   139.0          1        0.0001192179           0.0004768717
## 5   139.7          2        0.0002384359           0.0007153076
## 6   140.5          1        0.0001192179           0.0008345255
## 7   140.7          1        0.0001192179           0.0009537434
## 8   140.8          1        0.0001192179           0.0010729614
## 9   141.1          2        0.0002384359           0.0013113972
## 10  141.4          2        0.0002384359           0.0015498331
```

2. Computing and summarizing data for weight

```
Weight_data = dataset$Weight..kg.
frequencyW = table(Weight_data)
rfrequencyW = prop.table(frequencyW)
crfrequencyW = cumsum(rfrequencyW)
dataW = data.frame(
  Weight = as.numeric(names(frequencyW)),
  Wfrequency = as.numeric(frequencyW),
  relative_frequencyW = as.numeric(rfrequencyW),
  cumulative_relative_fW = as.numeric(crfrequencyW)
)

head(dataW, 10) # visualizes first 10 rows of the table
```

```
##    Weight Wfrequency relative_frequencyW cumulative_relative_fW
## 1    32.6          1        0.0001192179           0.0001192179
## 2    36.7          1        0.0001192179           0.0002384359
## 3    36.8          1        0.0001192179           0.0003576538
## 4    36.9          1        0.0001192179           0.0004768717
## 5    37.5          1        0.0001192179           0.0005960897
## 6    37.7          1        0.0001192179           0.0007153076
## 7    38.2          1        0.0001192179           0.0008345255
## 8    38.5          1        0.0001192179           0.0009537434
## 9    39.1          1        0.0001192179           0.0010729614
## 10   39.3          1        0.0001192179           0.0011921793
```

Calculating mean, Q1, median, mode, Q3, 70th percentile and IQR for the variables height and weight

1. Calculations for height variable

```
meanH = mean(Height_data)
meanH
```

```
## [1] 166.6412
```

```
Q1H = quantile(Height_data, probs = 0.25, type = 6)
Q1H
```

```
##   25%
## 159.1
```

```
medianH = median(Height_data)
medianH
```

```
## [1] 166.2
```

```
Q3H = quantile(Height_data, probs = 0.75, type = 6)
Q3H
```

```
##   75%
## 173.9
```

```
percentile70H = quantile(Height_data, probs = 0.7, type = 6)
percentile70H
```

```
##    70%
## 172.23
```

```
IQRH = as.numeric(Q3H) - as.numeric(Q1H)
IQRH
```

```
## [1] 14.8
```

2. Calculations for weight variable

```
meanW = mean(Weight_data)
meanW
```

```
## [1] 83.66922
```

```
Q1W = quantile(Weight_data, probs = 0.25, type = 6)
Q1W
```

```
##  25%
## 67.6
```

```
medianW = median(Weight_data)
medianW
```

```
## [1] 79.9
```

```
Q3W = quantile(Weight_data, probs = 0.75, type = 6)
Q3W
```

```
##  75%
## 95.8
```

```
percentile70W = quantile(Weight_data, probs = 0.7, type = 6)
percentile70W
```

```
##   70%
## 91.83
```

```
IQRW = as.numeric(Q3W) - as.numeric(Q1W)
IQRW
```

```
## [1] 28.2
```

Determining measures of variability (range, variance, standard deviation) for the height and weight variables

1. Calculations for height variable

```r
rangeH = range(Height_data)
rangeH
```

```
## [1] 131.1 199.6
```

```r
VarH = var(Height_data)
VarH
```

```
## [1] 101.5865
```

```r
SDH = sd(Height_data)
SDH
```

```
## [1] 10.07901
```

2. Calculations for weight variable

```r
rangeW = range(Weight_data)
rangeW
```

```
## [1]  32.6 254.3
```

```r
VarW = var(Weight_data)
VarW
```

```
## [1] 536.9248
```

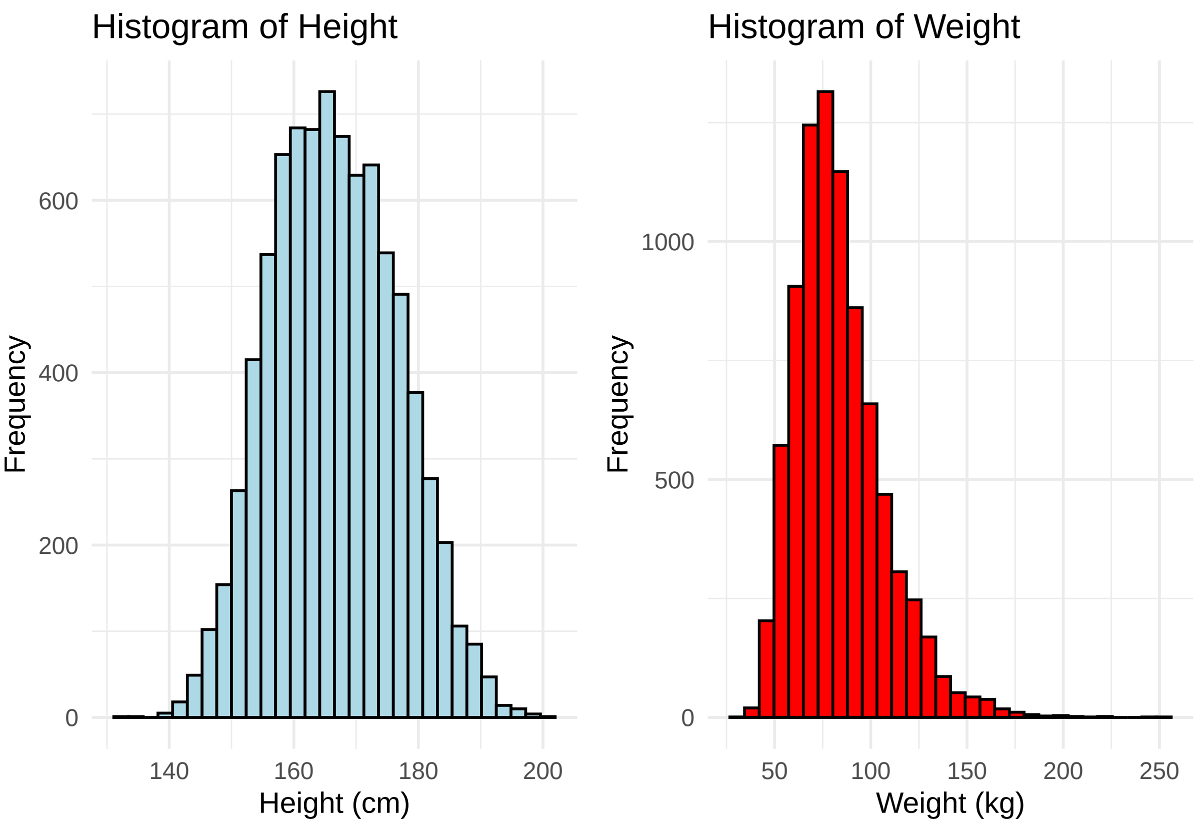
```r
SDW = sd(Weight_data)
SDW
```

```
## [1] 23.17164
```

Data Visualization and Distribution Analysis

Histograms of the frequency of weight and height measurements

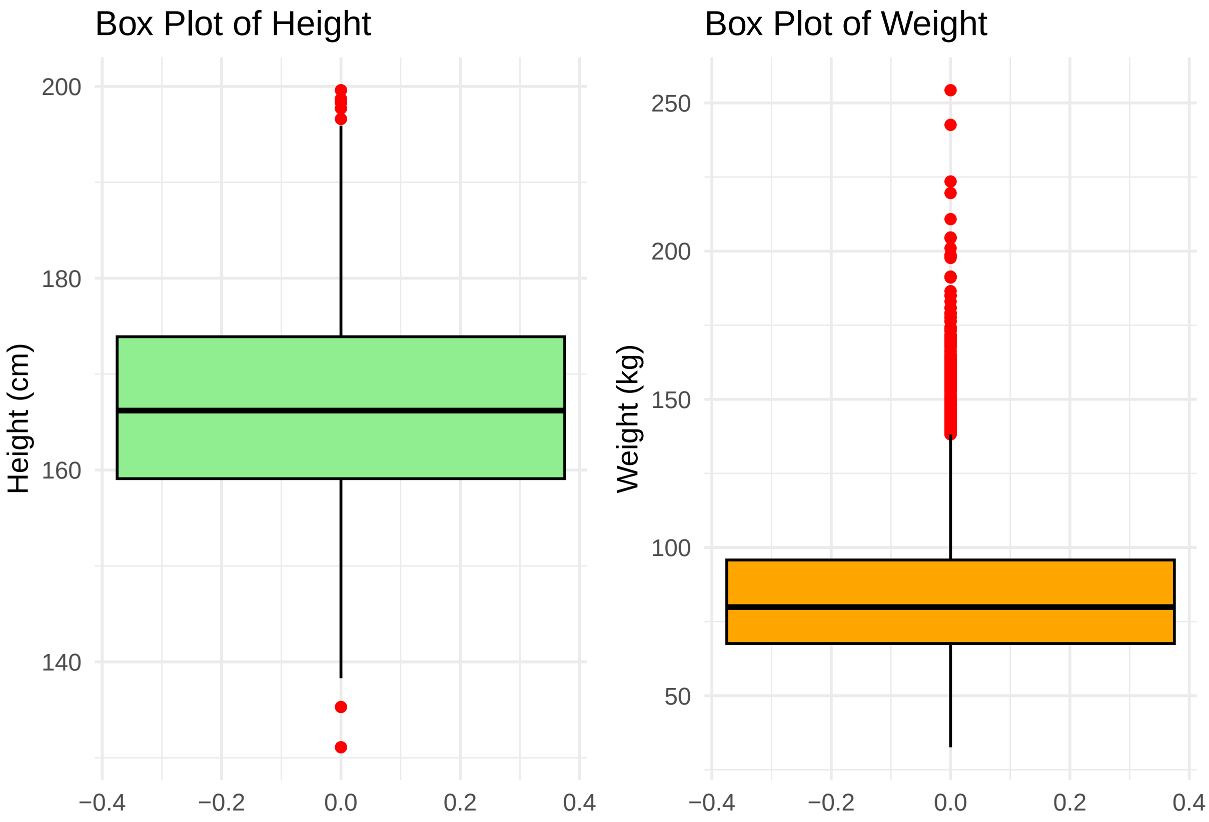
```r
library(patchwork)
library(ggplot2)
p1 = ggplot(NHANES.Weight.and.Height, aes(x = Standing.Height..cm.)) + geom_histogram(fill = "lightblue
  theme_minimal() + labs(title = "Histogram of Height", x = "Height (cm)", y = "Frequency")
p2 = ggplot(NHANES.Weight.and.Height, aes(x = Weight..kg.)) + geom_histogram(fill = "red", color = "bla
p1 | p2
```



In observing both graphs, the graph for the Height measurements is symmetric, however the graph for the Weight measurement is approximately asymmetric and skewed to the right (since most of the values are gathered on the left, and the potential outliers are on the right side of the graph).

Boxplots for the height and weight measurements

```r
library(patchwork)
library(ggplot2)
b1 = ggplot(NHANES.Weight.and.Height, aes(y = Standing.Height..cm.)) +
  geom_boxplot(
    fill = "lightgreen",
    color = "black",
    outlier.color = "red"
  ) + theme_minimal() + labs(title = "Box Plot of Height", y = "Height (cm)")
b2 = ggplot(NHANES.Weight.and.Height, aes(y = Weight..kg.)) +
  geom_boxplot(
    fill = "orange",
    color = "black",
    outlier.color = "red"
  ) + theme_minimal() + labs(title = "Box Plot of Weight", y = "Weight (kg)")
b1 | b2
```



In observing the boxplots of the height and weight variables, we can observe that there are many potential outliers for both the height and weight variables.In the height boxplot, we can observe that there is potentially around 7 outliers (observing the boxplot, we see around 6-8 red dots representing the outliers, where 2 are below the lower fence, and around 5 over the upper fence), and in the weight boxplot, we can observe that there is potentially more than 17 outliers (observing the boxplot, we can properly count 17 red dots, whereas the others are grouped together, which indicates that there are many potential outliers above the upper fence)

1. Outliers in height variable

$Lowerfence = Q1H - 1.5IQRH = 159.1 - 1.5(14.8) = 136.9$

$Upperfence = Q3H + 1.5IQRH = 173.9 + 1.5(14.8) = 196.1$

Therefore the outliers in height are 131.1, 135.3, 196.6, 197.7, 198.3, 198.4, 198.7 and 199.6

2. Outliers in weight variable:

$Lowerfence = Q1W - 1.5IQRW = 67.6 - 1.5(28.2) = 25.3$

$Upperfence = Q3W + 1.5IQRW = 95.8 + 1.5(28.2) = 138.1$

Therefore the outliers in weight are 138.2, 138.3, 138.5, 138.6, 138.7, 138.9, 139.4, 139.6, 139.9, 140.0, 140.1, 140.2, 140.3, 140.4, 140.5, 140.6, 140.9, 141.0, 141.3, 141.4, 141.6, 141.9, 142.2, 142.4, 142.7, 142.9, 143.0, 143.4, 143.5, 143.6, 143.7, 143.8, 144.0, 144.4, 144.6, 144.7, 145.1, 145.2, 145.3, 145.6, 145.7, 145.9, 146.2, 146.3, 146.4, 146.6, 146.7, 146.8, 146.9, 147.1, 147.4, 147.5, 147.6, 147.9, 148.0, 148.2, 148.4, 148.5, 148.6, 148.9, 149.0, 149.1, 149.2, 149.3, 149.4, 149.5, 149.7, 149.8, 150.6, 150.7, 150.9, 151.0, 151.1, 151.6, 151.7, 152.6, 152.8, 153.1, 153.4, 153.7, 153.8, 153.9, 154.0, 154.1, 154.3, 154.4, 154.5, 154.6, 154.7, 154.9, 155.4, 155.6, 155.9, 156.0, 156.2, 156.3, 156.9, 157.4, 157.6, 157.7, 157.9, 158.2, 158.5, 158.6, 158.7, 158.8, 158.9, 159.2, 159.6, 159.9, 160.1, 160.4, 160.6, 160.8, 160.9, 161.0, 161.1, 161.3, 161.4, 161.6, 162.1, 162.3, 162.3, 162.9, 163.0, 163.2, 163.8, 164.2, 164.8, 164.9, 165.1, 166.0, 166.6, 167.5, 167.9, 168.3, 168.9, 169.4, 169.7, 170.0, 170.8, 170.8, 171.1, 171.4, 171.4, 171.9, 173.0, 173.4, 174.0, 174.4, 176.2, 176.5, 177.6, 177.7, 178.9, 179.2, 180.8, 180.9, 183.0, 185.0, 186.5, 191.1, 191.3, 191.4, 197.7, 198.3, 198.7, 201.0, 204.4, 204.6, 210.8, 219.6, 223.5, 242.6 and 254.3

Spread/concentration of middle 50% of the data

As seen in class, the IQR represents the middle 50% and it's dispersion.

1. Observing the Height:

As found previously, IQRH = 14.8, and since the IQR of this distribution is of relatively small value, the middle 50% of the data appears to be concentrated. In observing the histogram, we can confirm this by observing that there are high frequencies of data that is part of the middle 50%. In addition, in the boxplot, we can see that the IQRH length (represented by the green box) appears to be relatively small, which also confirms our initial statement.

2. Observing the Weight:

As found previously, IQRW = 28.2, and since the IQR of this distribution is of relatively small value, the middle 50% of the data appears to be concentrated. In observing the histogram, we can confirm this by observing that there are high frequencies of data that is part of the middle 50%, near the median. In addition, in the boxplot, we can see that the IQRH length (represented by the orange box) appears to be relatively small, which also confirms our initial statement.

In observing the histogram and boxplot, we can conclude the Height has a normal distribution as it is approximately symmetric and the Weight does not have a normal distribution, as we can observe in the histogram, where the graph is skewed to the right.

Linear regression and Advanced Analysis

```r
options(repos = c(CRAN = "https://cran.rstudio.com/"))
install.packages("MPV")
```

```
## package 'MPV' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\mike2\AppData\Local\Temp\Rtmp0E9xxn\downloaded_packages
```

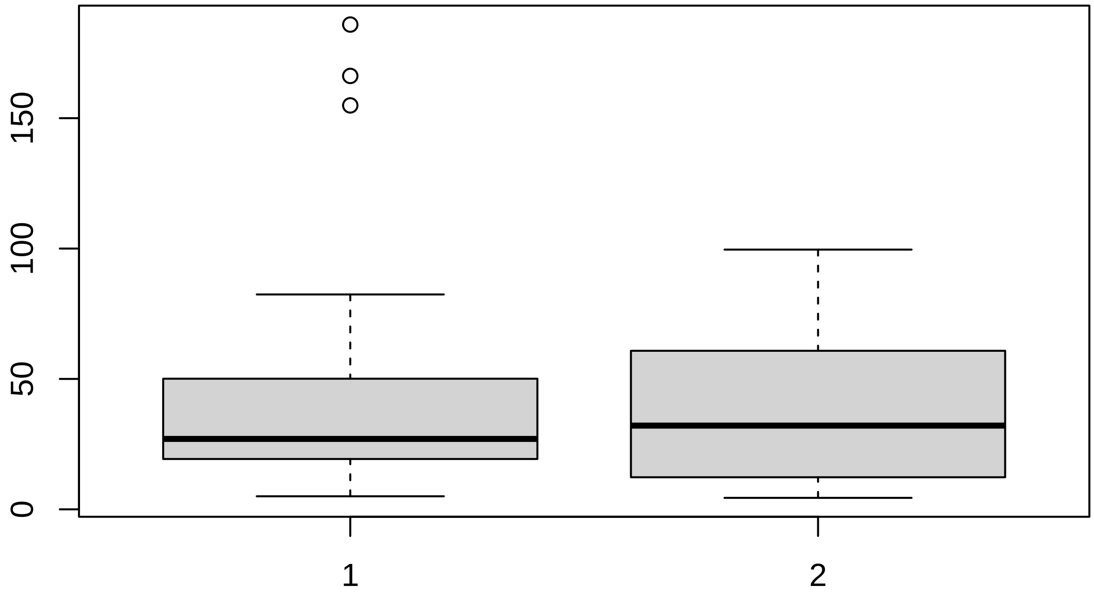
```r
library("MPV")

data(p2.18)
head(p2.18)
```

```
##               Firm Amount.Spent Returned.Impressions
## 1      Miller Lite         50.1                 32.1
## 2            Pepsi         74.1                 99.6
## 3          Stroh's         19.3                 11.7
## 4 Federal Express         22.9                 21.9
## 5      Burger King         82.4                 60.8
## 6        Coca-Cola         40.1                 78.6
```

```
dim(p2.18)
```

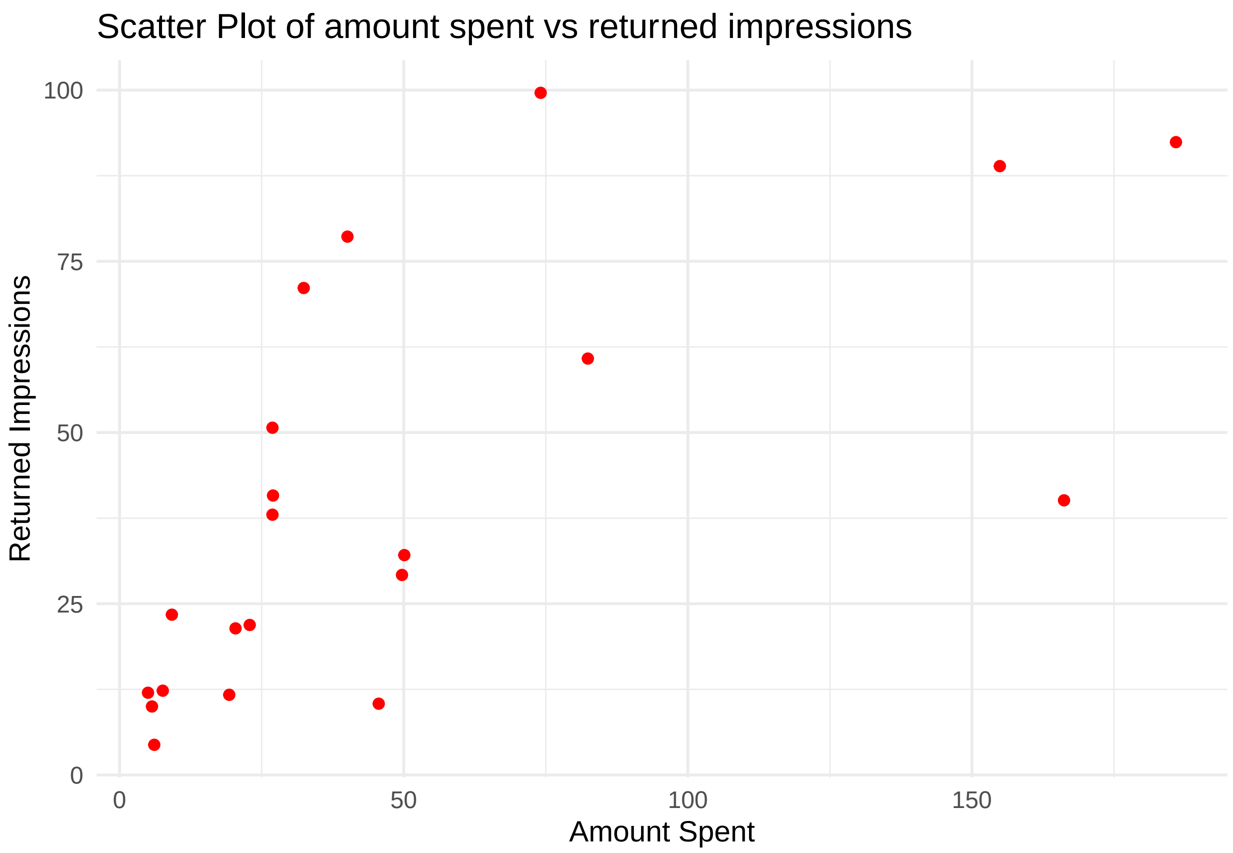
```
## [1] 21  3
```

```
boxplot(p2.18$Amount.Spent, p2.18$Returned.Impressions)
```



a) Making the scatterplot of the data

```
sp = ggplot(p2.18, aes(x = Amount.Spent, y = Returned.Impressions)) + geom_point(color = "red") + theme_
sp
```



b) Fitting a simple linear regression model to data

```
model = lm(Returned.Impressions ~ Amount.Spent, p2.18)
summary(model)
```

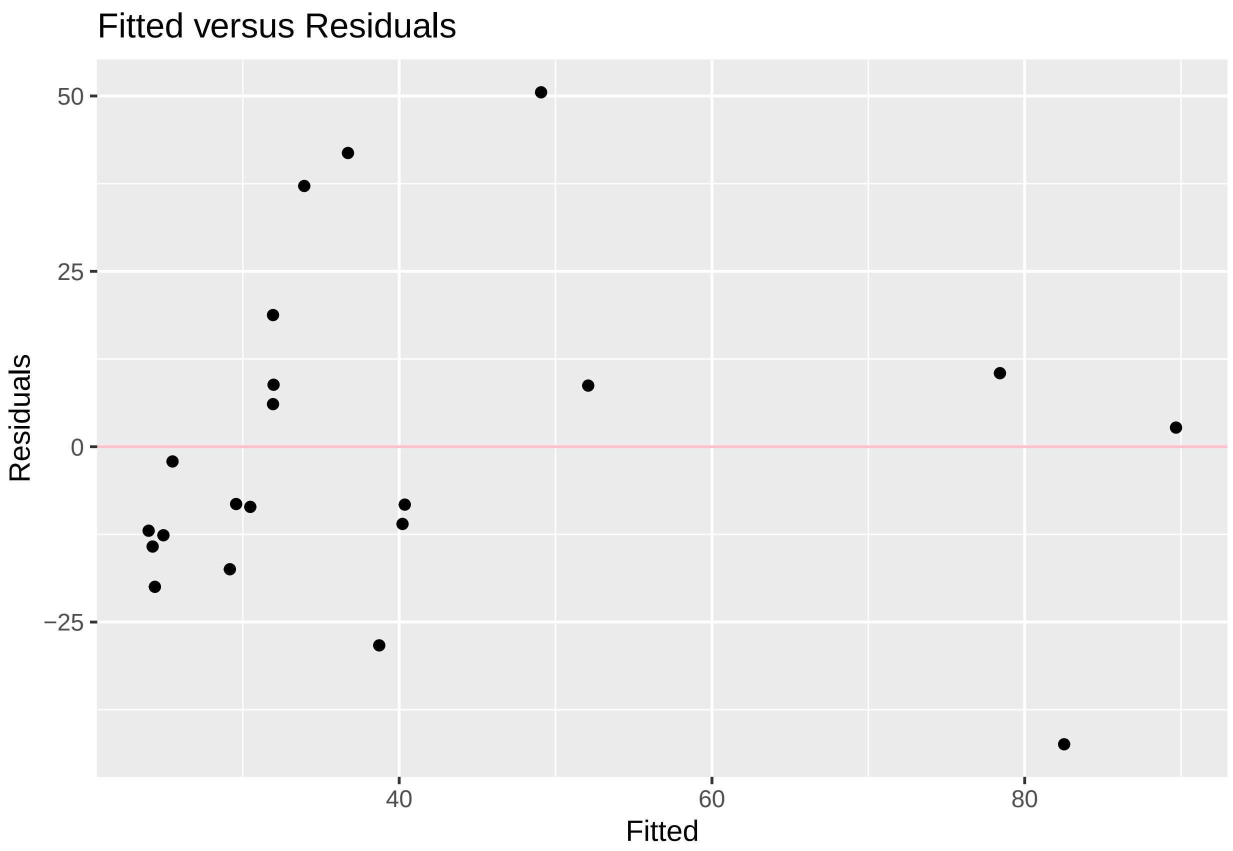
```
##
## Call:
## lm(formula = Returned.Impressions ~ Amount.Spent, data = p2.18)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -42.422 -12.623  -8.171   8.832  50.526
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  22.16269    7.08948   3.126  0.00556 **
## Amount.Spent  0.36317    0.09712   3.739  0.00139 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 23.5 on 19 degrees of freedom
## Multiple R-squared:  0.424,  Adjusted R-squared:  0.3936
## F-statistic: 13.98 on 1 and 19 DF,  p-value: 0.001389
```

In observing the summary of the model, we have the linear regression model: $\hat{y} = 0.36317 * (Amount.Spent) + 22.16269$.

c) Checking to see if there is a significant relationship between the amount a company spends on advertising and retained impressions

1. Residual Plot

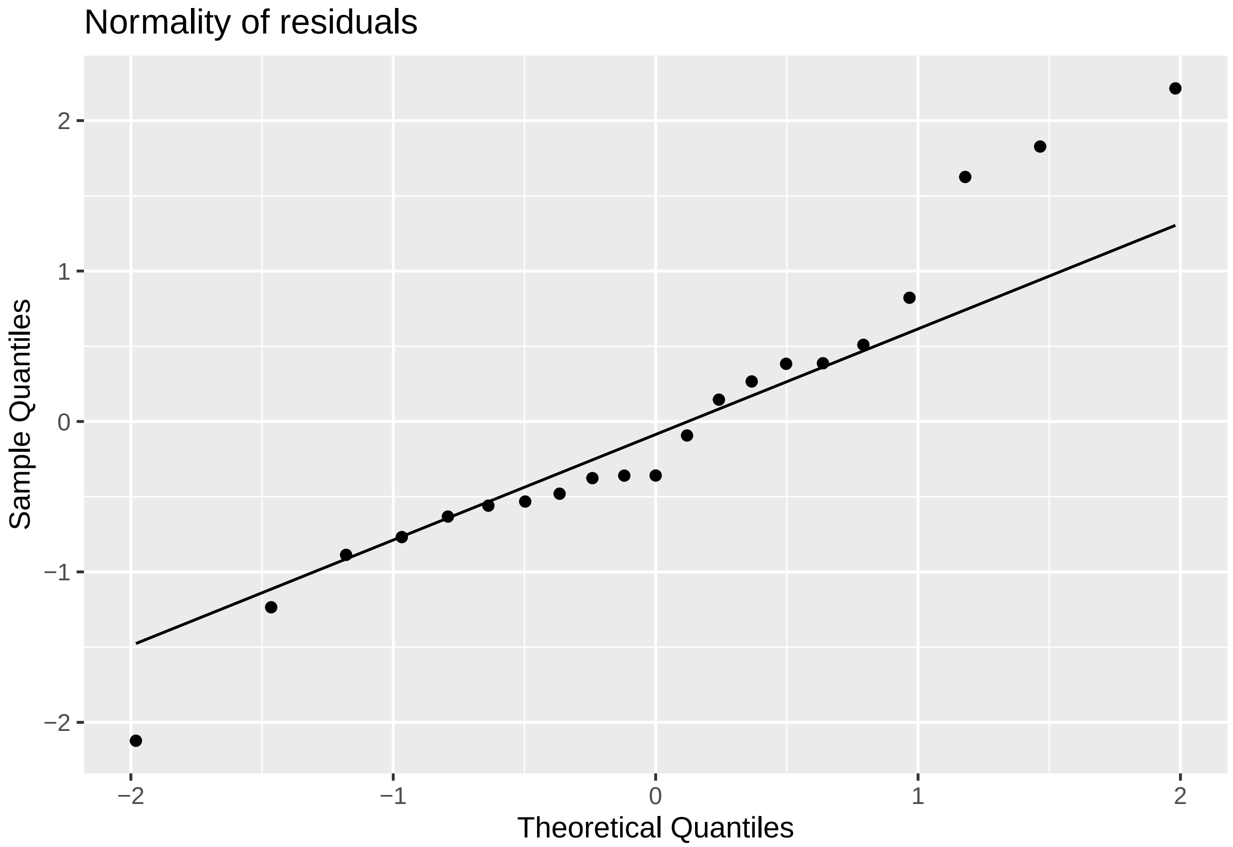
```
rp = ggplot(model, aes(.fitted, .resid)) + geom_point() + geom_hline(yintercept = 0, color = "pink") + l
rp
```



In observing the residual plot, we can see that the points are randomly and evenly scattered around the 0 line, which indicates that the variance is constant and that the linear regression model satisfies this assumption (that the variance is constant).

2. Normality of residuals through QQplot

```
qq = ggplot(model, aes(sample = rstandard(model))) + geom_qq() + stat_qq_line() + labs(title = "Normali
qq
```



In observing the QQplot of the residuals, since most of the points are on/around the line, we can conclude that the assumption of the normality of the residuals is confirmed.

In observing these 2 results, along with calculating the $R^2_{adjusted}$ (done in part d), we can conclude that the model satisfies the assumptions (normality of residuals and variance is constant) in order for the model to be a good fit to the data. However, we can also conclude that there is a weak/no relationship between the amount a company spends on advertising and the retained impressions since the $R^2_{adjusted}$ is near 0.

d) Calculating and analyzing the $R^2_{adjusted}$

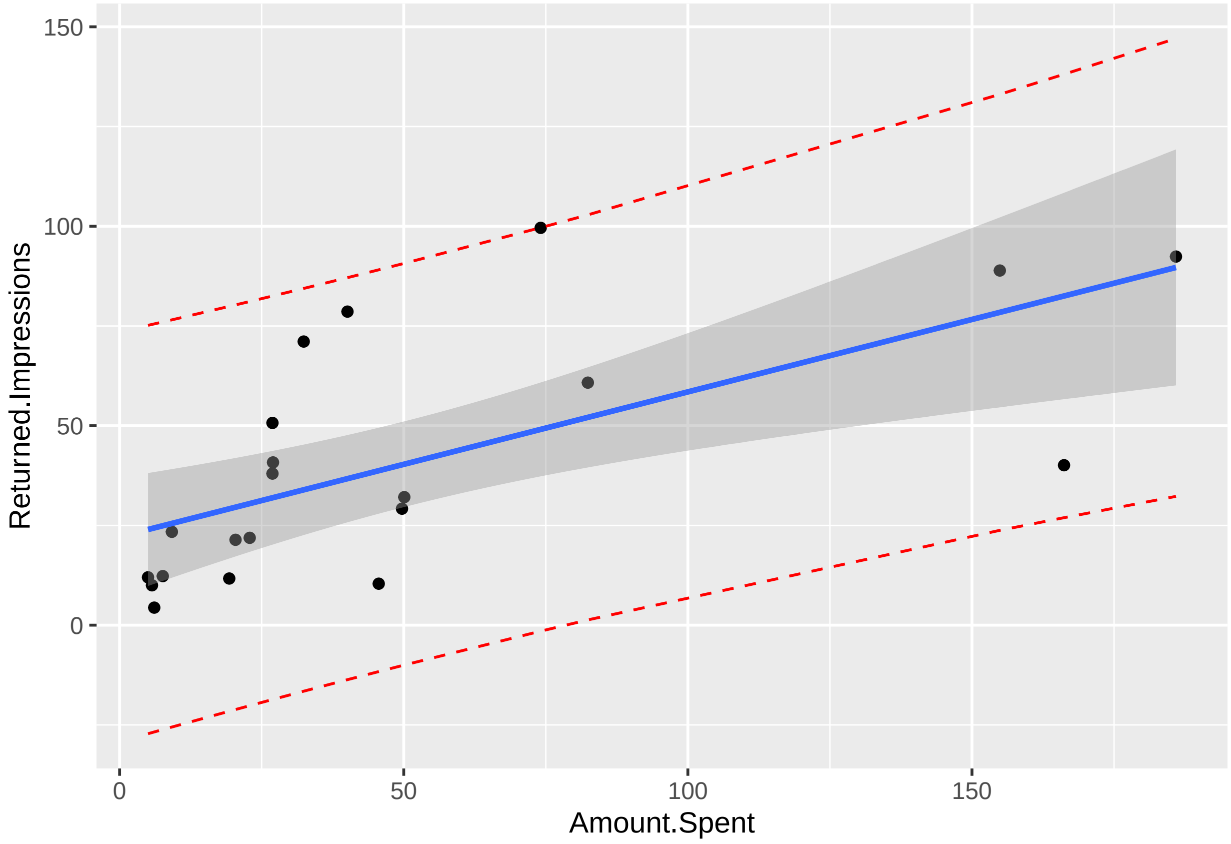
```r
summary(model)
```

```
##
## Call:
## lm(formula = Returned.Impressions ~ Amount.Spent, data = p2.18)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -42.422 -12.623  -8.171   8.832  50.526
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  22.16269    7.08948   3.126  0.00556 **
## Amount.Spent  0.36317    0.09712   3.739  0.00139 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 23.5 on 19 degrees of freedom
## Multiple R-squared:  0.424,  Adjusted R-squared:  0.3936
## F-statistic: 13.98 on 1 and 19 DF,  p-value: 0.001389
```

In observing the summary of the model, we can see that the $R^2_{adjusted}$ is 0.3936. In observing this value, we can say that the model is not a good fit for the dataset and that there is a weak/no relationship between the variables since the value is very close to 0.

e) Construct and plot the 95% confidence and prediction bands for the data.

```
data_prdF = predict(model, interval = "prediction", level = 0.95)
data_newF = cbind(p2.18, data_prdF)
ggplot(data_newF, aes(x = Amount.Spent, y = Returned.Impressions)) + geom_point() + geom_line(aes(y = lw
```



f) Give the 95% confidence and prediction intervals for the number of returned impressions for MCI.

```r
## The 95% confidence interval
confint(model)
```

```
##                 2.5 %     97.5 %
## (Intercept)  7.324244 37.0011425
## Amount.Spent 0.159899  0.5664492
```

```r
## Prediction for Amount.Spent = 200 and Amount.Spent = 250, along with their prediction intervals
newX = data.frame(Amount.Spent = c(200, 250))
predict(model, newdata = newX, interval = 'prediction')
```

```
##         fit      lwr      upr
## 1  94.79751 35.97943 153.6156
## 2 112.95621 48.29535 177.6171
```

The CI for β_0 is [7.324244, 37.0011425] and the CI for β_1 is [0.159899, 0.5664492] for the linear model $\hat{y} = \beta_0 + \beta_1(Amount.Spent)$. In using the model to predict the returned impressions for the amount spent being 200 and 250, we obtain the values 94.79751 with a prediction interval of [35.97943, 153.6156] for spending 200 and 112.95621 with a prediction interval of [48.29535, 177.6171] for spending 250.