Assignment_1

April 1, 2023

0.1 Assignment - 1

0.1.1 Implementing Correlation & Regression using R Programming Language

Name	USN	Class
Ajay Nagaraj M	2SD19CS008	8TH Sem CSE B Div

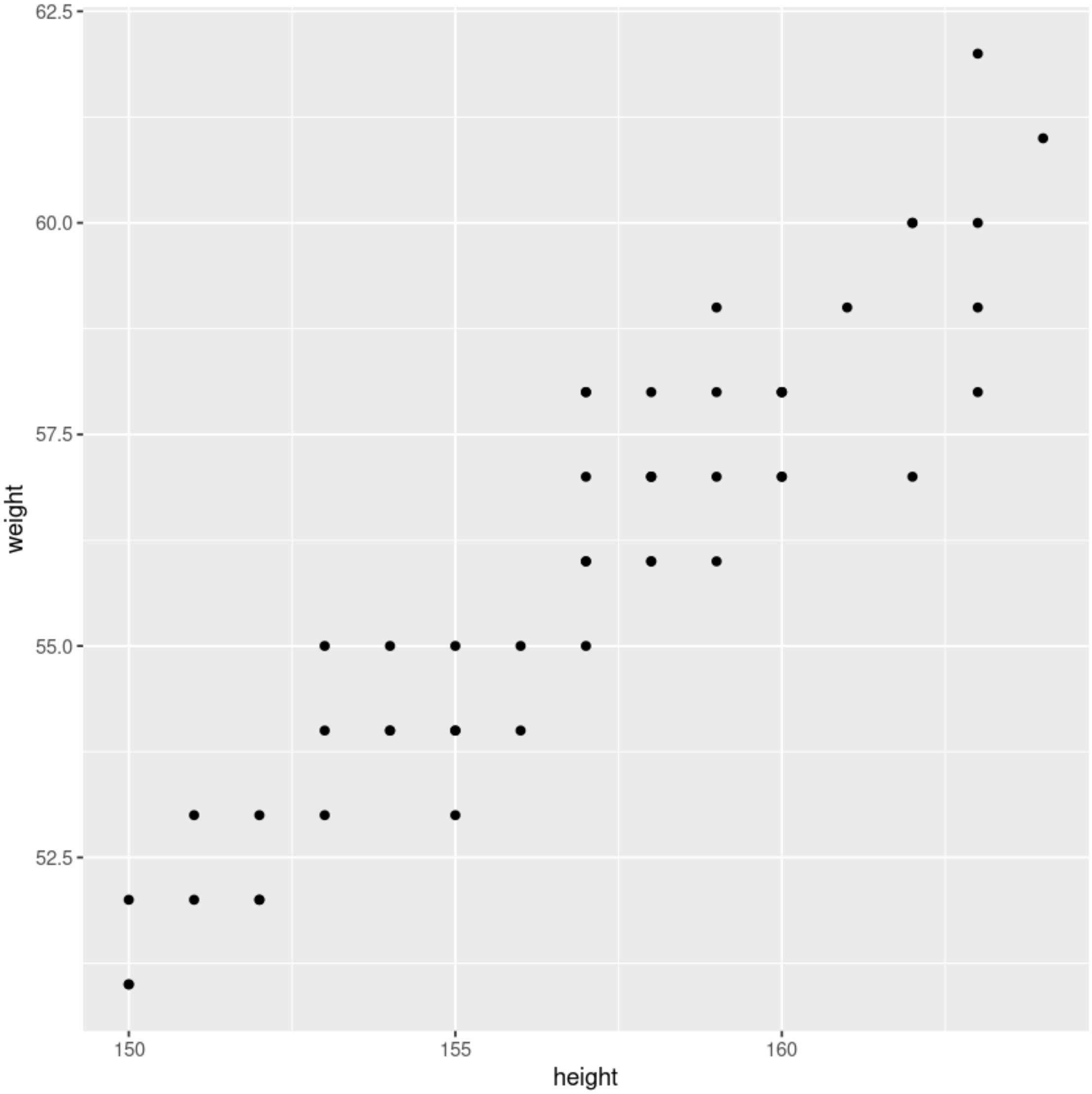
```
[ ]: # Reading The data file
     dataset <- read.csv('Assig1.csv')
```

```
[ ]: head(dataset)
```

A data.frame: 6 × 3

	respondent	height	weight
	<int>	<int>	<int>
1	1	158	58
2	2	153	55
3	3	152	53
4	4	163	59
5	5	157	55
6	6	158	56

```
[ ]: #  Plotting Scatter Plot using ggplot2 Library
     library(ggplot2)
     ggplot(dataset, aes(x=height, y=weight)) + geom_point()
```



Calculating Correlation

[ ]:

```
#  Calculating Correlation Between Height and Weight
corr = cor.test(dataset$weight, dataset$height)
corr
```

```

	Pearson's product-moment correlation

data:  dataset$weight and dataset$height
t = 17.528, df = 48, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.8792387 0.9598644
sample estimates:
      cor
0.9299869
```

Calculating Regression

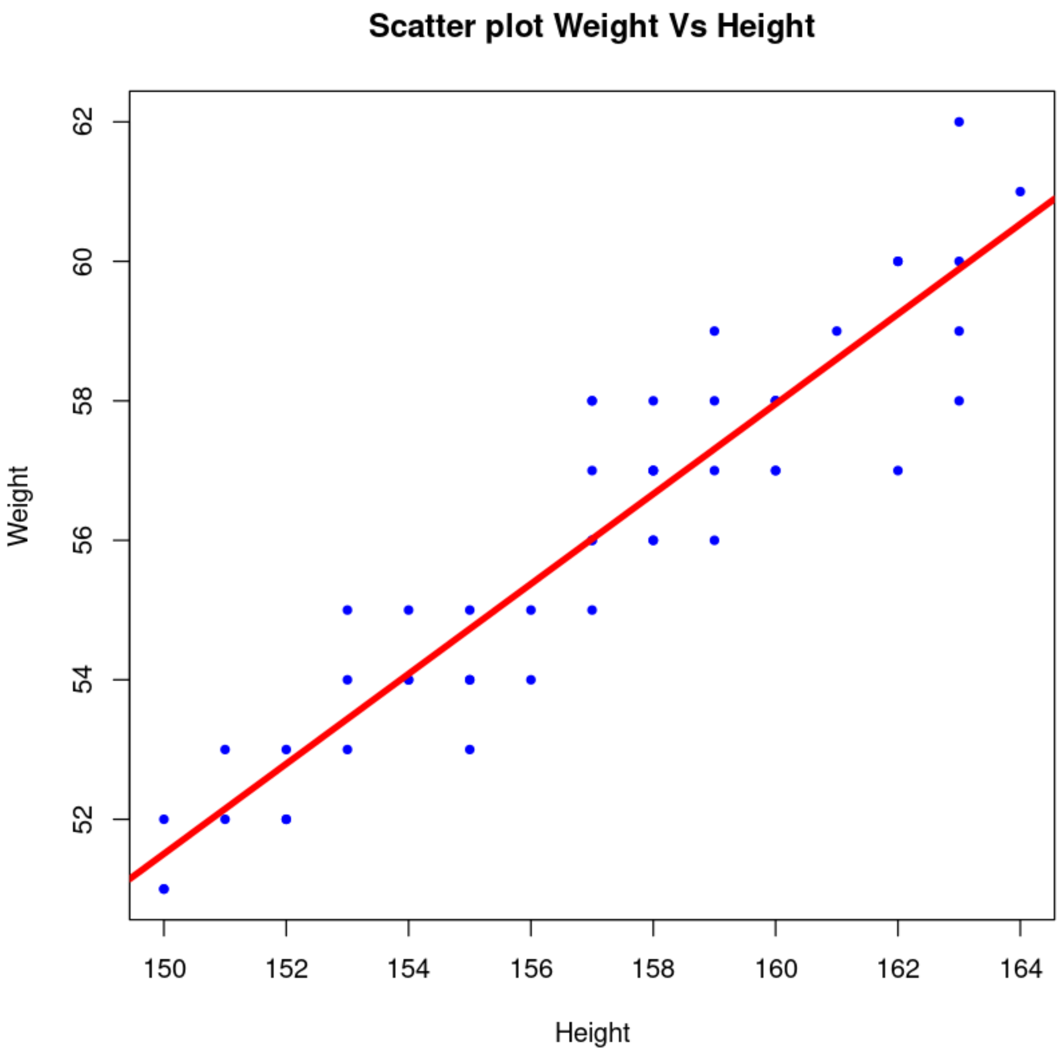
[ ]:

```
# Calculating Regression using lm module
regress = lm(dataset$weight~dataset$height)
summary(regress)
```

```
Call:
lm(formula = dataset$weight ~ dataset$height)

Residuals:
     Min       1Q   Median       3Q      Max
-2.24493 -0.71377  0.01252  0.54369  2.11008

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)    -45.2429     5.7790  -7.829    4e-10 ***
dataset$height   0.6450     0.0368  17.528   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.9997 on 48 degrees of freedom
Multiple R-squared:  0.8649,	Adjusted R-squared:  0.8621
F-statistic: 307.2 on 1 and 48 DF,  p-value: < 2.2e-16
```

[ ]:

```
# Plotting Regression Line
plot(dataset$height, dataset$weight, main = "Scatter plot Weight Vs Height",␣
 ↪xlab = "Height", ylab = "Weight", pch=20, col="blue")
abline(regress, col="red", lwd=4)
```



```
[ ]: shapiro.test(residuals(regress))
```

```
	Shapiro-Wilk normality test

data:  residuals(regress)
W = 0.98573, p-value = 0.8024
```

Testing using a Sample Test Data

```
[ ]: #  Function to Predict the Weight using Height
     PredictWeight <- function(height) { # create a function with the name␣
       ↪my_function
```

```
    return(0.6450 * height - 45.2429)

}
```

[ ]:

```
# Height in cm -> 160cm
PredictWeight(160)
```

```
57.9571
```