Name of Student: Pushkar Sane			
Roll Number: 45		**Lab Assignment Number: 10**	
Title of Lab Assignment: Implementation and analysis of Linear regression through graphical methods.			
DOP: 24-10-2023		**DOS: 27-10-2023**	
CO Mapped: **CO6**	**PO Mapped:** **PO1, PO2, PO3, PO4, PO5, PO7, PO12, PSO1, PSO2**		**Signature:**

Practical No. 10

Aim: Implementation and analysis of Linear regression through graphical methods.

Theory:

Linear Regression using R

Regression analysis is used to establish relationships between two variables.

Simple linear regression is used to estimate the relationship between **two** quantitative variables.

You can use simple linear regression when you want to know:

1. How strong the relationship is between two variables (e.g. the relationship between rainfall and soil erosion).
2. The value of the **dependent variable** at a certain value of the **independent variable**. (e.g. the amount of soil erosion at a certain level of rainfall).

Predictor or independent Variable: The **values** that are gathered through **experiments** is known as a predictor variable.

Response or dependent or predicted Variable: The **values that are derived from predictor variables** are known as response variables.

Linear regression is a regression model that uses a **straight line** to describe the relationship between variables. It finds the **line of best fit through your data** by searching for the value of the regression coefficient(s) that minimizes the total error of the model.

In **linear** regression predictor and response variables are related through an equation where exponent (power) of both these variables is **1**.

A **non-linear relationship** where the exponent of a variable is not equal to 1 creates a curve.

The equation for linear regression is **y=ax+b**.

Here,

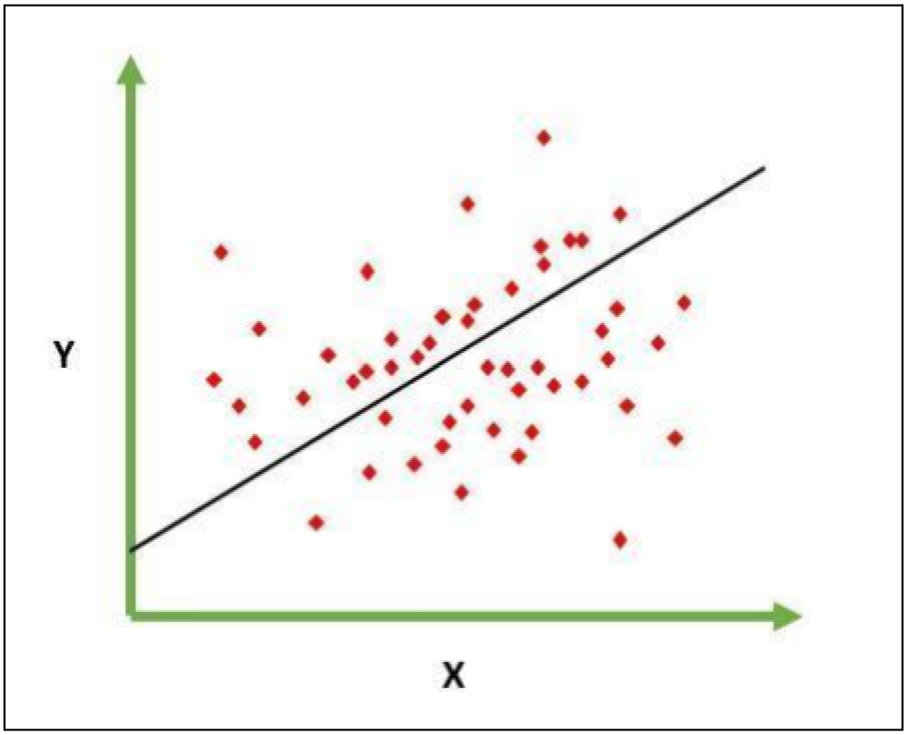
y is the response variable

x is the predictor variable

a is regression coefficient

b is y-intercept

The **linear regression** technique involves the continuous dependent variable and the independent variables can be continuous or discrete. By using best fit straight line linear regression sets up a relationship between dependent variable (Y) and one or more independent variables (X). In other words, there exists a linear relationship between independent and dependent variables.



In the above diagram, you see that the points can be anywhere in the plane of a graph in and around a straight line.

There are two main types of linear regression:

a. **Simple linear regression** uses only **one independent** variable.
b. **Multiple linear regression** uses **two or more independent** variables.

Simple linear regression:

Example :

You are a social researcher interested in the relationship between income and happiness. You survey 500 people whose incomes range from $15k to $75k and ask them to rank their happiness on a scale from 1 to 10. The income values are divided by 10,000 to make the

income data match the scale of the happiness scores (so a value of $2 represents $20,000, $3 is $30,000, etc.)
Your independent variable (income) and dependent variable (happiness) are both quantitative, so you can do a regression analysis to see if there is a linear relationship between them.

Multiple linear regression:
It is used to estimate the relationship between **two or more independent variables** and **one dependent variable.**

Equation can be of the form: **y= ax + bz + c**
y is the response variable
x and **z** are the predictor variable
a and **b** is regression coefficient
c is y-intercept

You can use multiple linear regression when you want to know:

1. How strong the relationship is between two or more independent variables and one dependent variable (e.g. how rainfall (1st independent variable), temperature (2nd independent variable), and amount of fertilizer added (3rd independent variable), affect crop growth (dependent variable).
2. The value of the dependent variable at a certain value of the independent variables (e.g. the expected yield of a crop at certain levels of rainfall, temperature, and fertilizer addition).

Example:
You are a public health researcher interested in social factors that influence heart disease. You survey 500 towns and gather data on the percentage of people in each town who smoke, the percentage of people in each town who bike to work, and the percentage of people in each town who have heart disease.
Because you have two independent variables and one dependent variable, and all your variables are quantitative, you can use multiple linear regression to analyze the relationship between them.

How to do the practical in R

install.packages("ggplot2")

library(ggplot2)

A system for 'declaratively' creating graphics, based on "The Grammar of Graphics". You provide the data, tell 'ggplot2' how to map variables to aesthetics, what graphical primitives to use, and it takes care of the details.

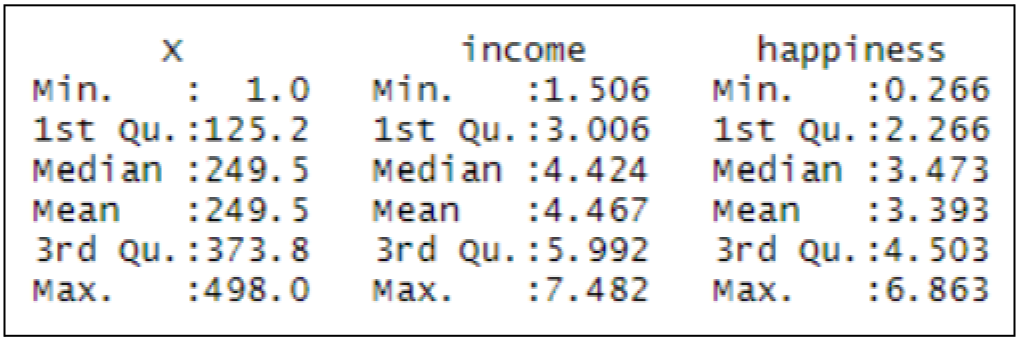
Step 1: Load the data into R

Simple regression

incomedata = read.csv("income.data for linear regression.csv")

summary(incomedata)

Because both our variables are quantitative, when we run this function we see a table in our console with a numeric summary of the data. This tells us the minimum, median, mean, and maximum values of the independent variable (income) and dependent variable (happiness):



```
       X              income         happiness    
 Min.   :  1.0   Min.   :1.506   Min.   :0.266  
 1st Qu.:125.2   1st Qu.:3.006   1st Qu.:2.266  
 Median :249.5   Median :4.424   Median :3.473  
 Mean   :249.5   Mean   :4.467   Mean   :3.393  
 3rd Qu.:373.8   3rd Qu.:5.992   3rd Qu.:4.503  
 Max.   :498.0   Max.   :7.482   Max.   :6.863  
```

Step 2: Make sure your data meet the assumptions

We can use R to check that our data meet the 3 main assumptions for linear regression.

Simple regression:

1. **Independence of observations:**

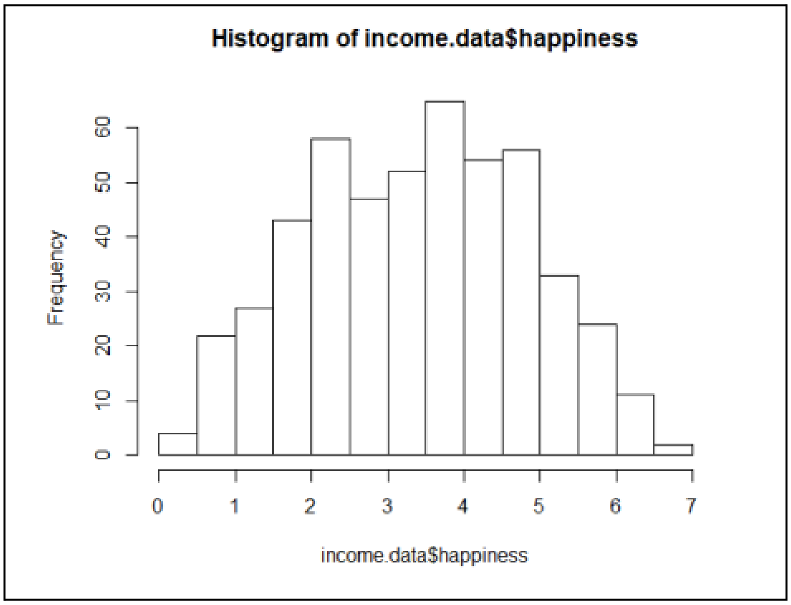
   Because we only have **one independent variable and one dependent variable**, we don’t need to test for any hidden relationships among variables.

   If you know that you have autocorrelation within variables (i.e. multiple observations of the same test subject), then do not proceed with a simple linear regression! Use a structured model, like a linear mixed-effects model, instead.

2. **Normality:**

   To check whether the **dependent variable** follows a normal distribution, use the hist() function.

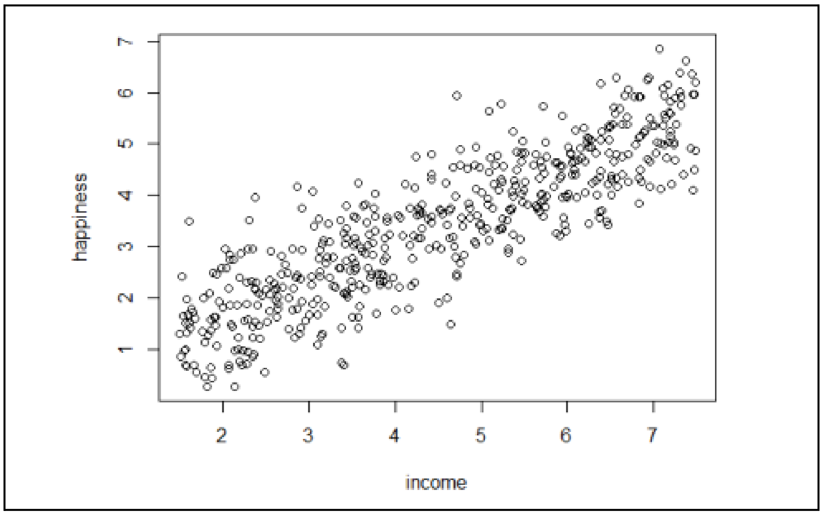
   hist(incomedata$happiness)



3. **Linearity:**

   The **relationship between the independent and dependent variable must be linear**. We can test this visually with a **scatter plot** to see if the **distribution of data points could be described with a straight line**.

   plot(happiness ~ income, data = incomedata)



The relationship looks **roughly linear**, so we can proceed with the linear model.

Step 3: Perform the linear regression analysis.

Simple regression: income and happiness

Let’s see if there’s a linear relationship between income and happiness in our survey of 500 people with incomes ranging from $15k to $75k, where happiness is measured on a scale of 1 to 10.

To perform a simple linear regression analysis and check the results, you need to run two lines of code. The first line of code makes the linear model, and the second line prints out the summary of the model:

The lm() function

In R, the lm(), or “linear model,” function can be used to create a simple regression model. The lm() function accepts a number of arguments. The following list explains the two most commonly used parameters.

- **formula:** describes the model

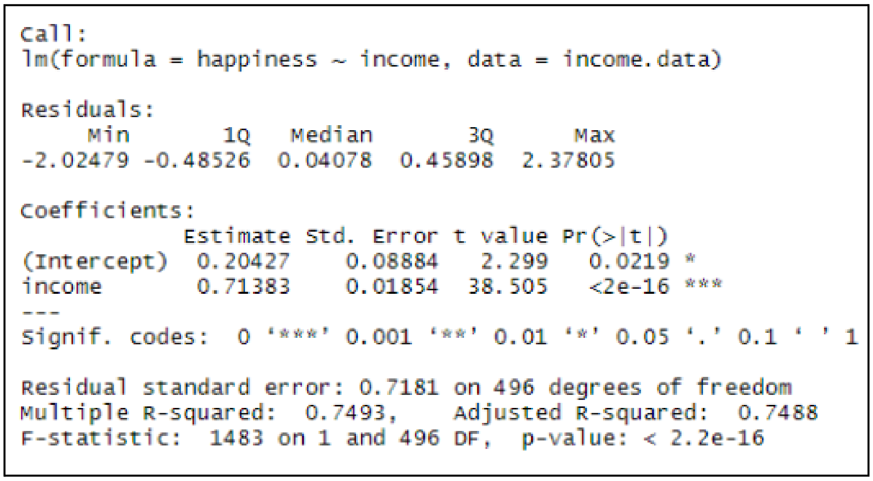
  Note that the formula argument follows a specific format. For simple linear regression, this is **“YVAR ~ XVAR”** where *YVAR is the dependent, or predicted or target variable and XVAR is the independent, or predictor variable*.
- **data:** the variable that contains the dataset

  lm([target variable] ~[predictor variables], data = [data source])

  income.happiness.lm <- lm(happiness ~ income, data = incomedata)

  summary(income.happiness.lm)

  The output looks like this:



```
Call:
lm(formula = happiness ~ income, data = income.data)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.02479 -0.48526  0.04078  0.45898  2.37805 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  0.20427    0.08884   2.299   0.0219 *  
income       0.71383    0.01854  38.505   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7181 on 496 degrees of freedom
Multiple R-squared:  0.7493,    Adjusted R-squared:  0.7488 
F-statistic:  1483 on 1 and 496 DF,  p-value: < 2.2e-16
```

This output table **first repeats the formula** that was used to generate the results (‘Call’), then **summarizes the model residuals** (‘Residuals’), which give an idea of how well the model fits the real data.

Next is the **‘Coefficients’** table.

The **first row gives the estimates of the y-intercept**, and the **second row gives the regression coefficient** of the model.

Row 1 of the table is labeled (Intercept). This is the **y-intercept** of the regression equation, with a value of 0.20. You can plug this into your regression equation if you want to predict happiness values across the range of income that you have observed:

*happiness = 0.20 + 0.71*income ± 0.018*

Row 2 in the ‘Coefficients’ table is **income**. This is the row that describes the estimated effect of income on reported happiness:

The Estimate column is the estimated **effect**, also called the **regression coefficient** or r2 value. The number in the table (0.713) tells us that for every one unit increase in income (where one unit of income = $10,000) there is a corresponding 0.71-unit increase in reported happiness (where happiness is a scale of 1 to 10).

The Std. Error column displays the **standard error** (*In statistics, a sample mean deviates from the actual mean of a population; this deviation is the standard error of the mean.*) of the estimate. This number shows how much variation there is in our estimate of the relationship between income and happiness.

The **Pr(>| t |)** column shows the **p-value**. Because the p-value is so low ($p < 0.001$), we can **reject the null hypothesis** and **conclude that income has a statistically significant effect on happiness**.

The last three lines of the model summary are statistics about the model as a whole.

The most important thing to notice here is the p-value of the model.

Here it is significant ($p < 0.001$), which means that this model is a good fit for the **observed data**.

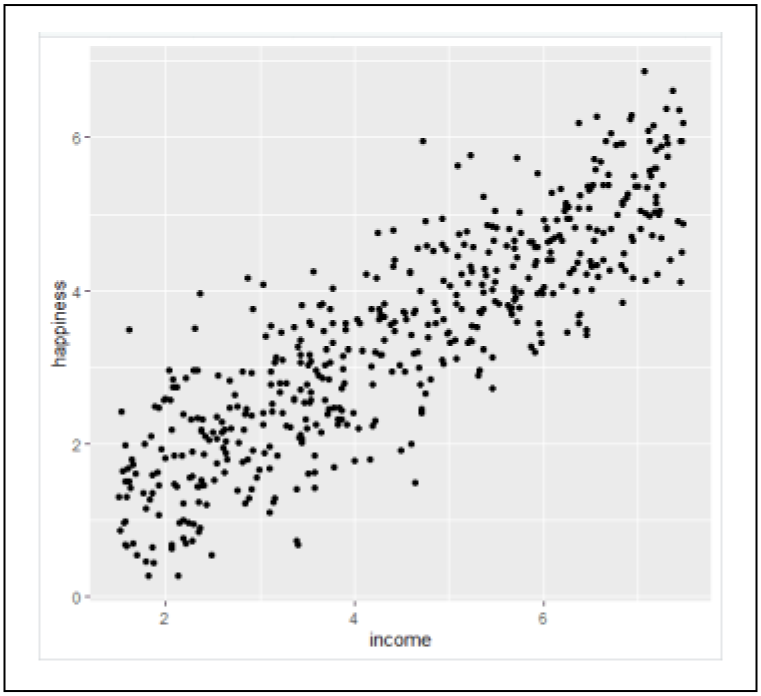
Step 4: Visualize the results with a graph

Simple regression

Follow 4 steps to visualize the results of your simple linear regression.

1. **Plot the data points on a graph**

```
income.graph <- ggplot(incomedata, aes(x=income, y=happiness))+
                    geom_point()
income.graph
```



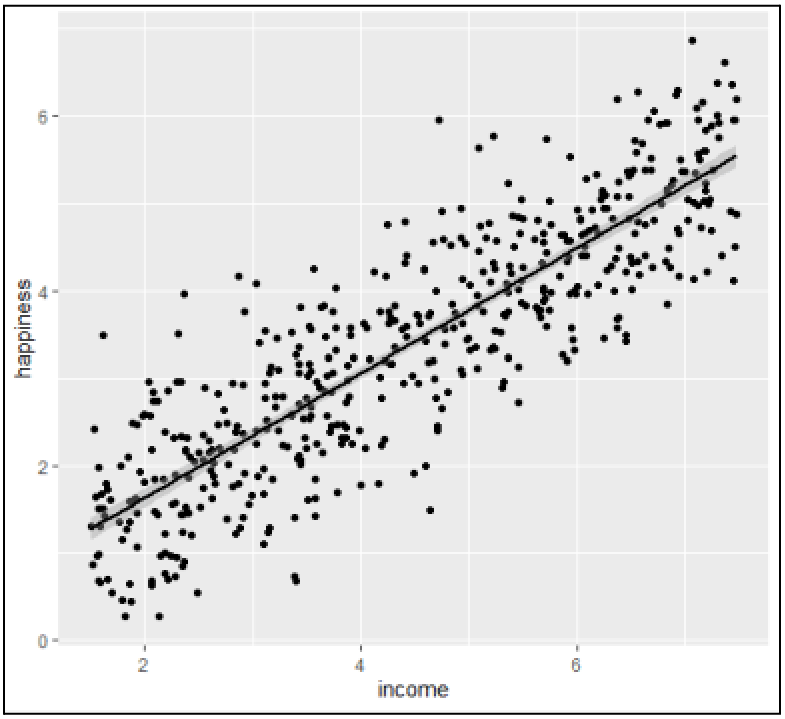
ggplot2 is a plotting package that makes it simple to create complex plots from data in a data frame.

geom_point() is used to create scatterplots. The scatterplot is most useful for displaying the relationship between two continuous variables.

2. **Add the linear regression line to the plotted data**

Add the regression line using geom_smooth() and typing in lm as your method for creating the line. This will add the line of the linear regression as well as the standard error of the estimate (in this case +/- 0.01) as a light gray stripe surrounding the line:

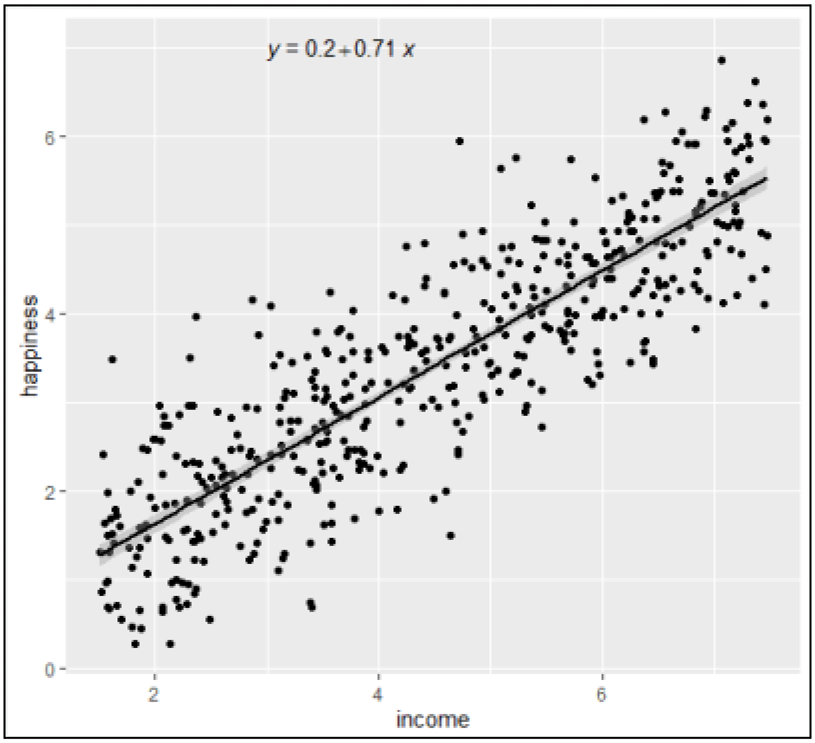
```
income.graph <- income.graph + geom_smooth(method="lm", col="black")
income.graph
```



3. **Add the equation for the regression line.**

```
install.packages("ggpubr") // to use stat_regline_equation() function
library(ggpubr)
income.graph <- income.graph +
                    stat_regline_equation(label.x = 3, label.y = 7)
income.graph
```

label.x, label.y	numeric coordinates (in data units) to be used for absolute positioning of the label.



Problem Statement: Implementation and analysis of Linear regression through graphical methods.

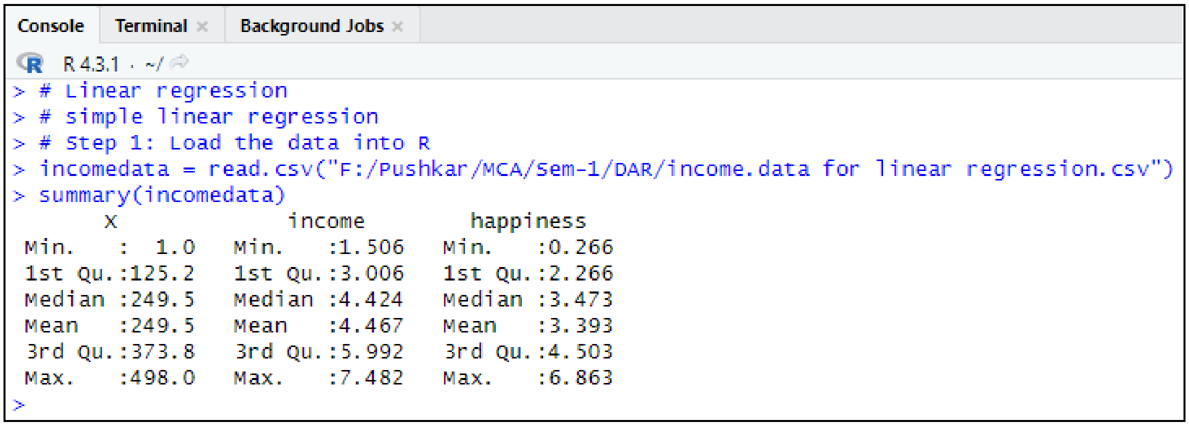
Code (Script):

```
# Linear regression
# simple linear regression
# Step 1: Load the data into R
incomedata = read.csv("C:\\Users\\ADMIN\\Desktop\\practicals\\dar\\practical 10\\income.data

for linear regression.csv")
summary(incomedata)

# Step 2: Make sure your data meet the assumptions
# Normality
hist(incomedata$happiness)

# Linearity
plot(happiness ~ income, data = incomedata)
```

```
# Step 3: Perform the simple linear regression analysis
income.happiness.lm <- lm(happiness ~ income, data = incomedata)
summary(income.happiness.lm)

# Step 4: Visualize the results with a graph
library(ggplot2)
# scatter plot
income.graph<-ggplot(incomedata, aes(x=income, y=happiness))+ geom_point()
income.graph

# Add the linear regression line to the plotted data
income.graph <- income.graph + geom_smooth(method="lm")
income.graph

# Add the equation for the regression line.
library(ggpubr)
income.graph <- income.graph +
  stat_regline_equation(label.x = 3, label.y = 7)
income.graph
```

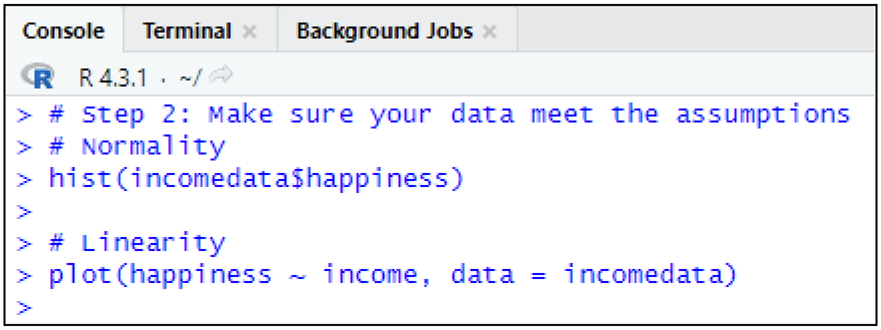
<u>Output:</u>

Step 1:

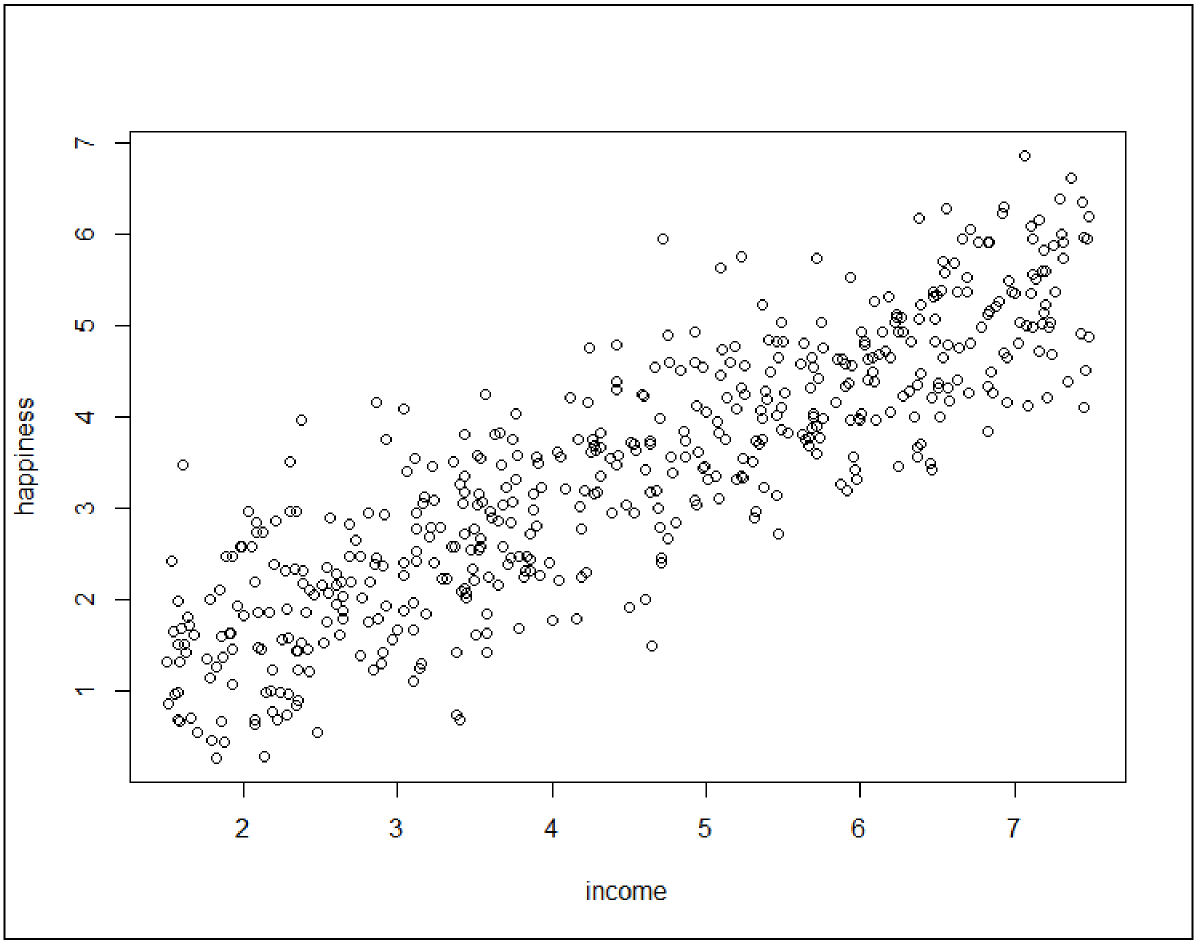


```
> # Linear regression
> # simple linear regression
> # Step 1: Load the data into R
> incomedata = read.csv("F:/Pushkar/MCA/Sem-1/DAR/income.data for linear regression.csv")
> summary(incomedata)
       X              income          happiness    
 Min.   :  1.0   Min.   :1.506   Min.   :0.266  
 1st Qu.:125.2   1st Qu.:3.006   1st Qu.:2.266  
 Median :249.5   Median :4.424   Median :3.473  
 Mean   :249.5   Mean   :4.467   Mean   :3.393  
 3rd Qu.:373.8   3rd Qu.:5.992   3rd Qu.:4.503  
 Max.   :498.0   Max.   :7.482   Max.   :6.863  
> 
```

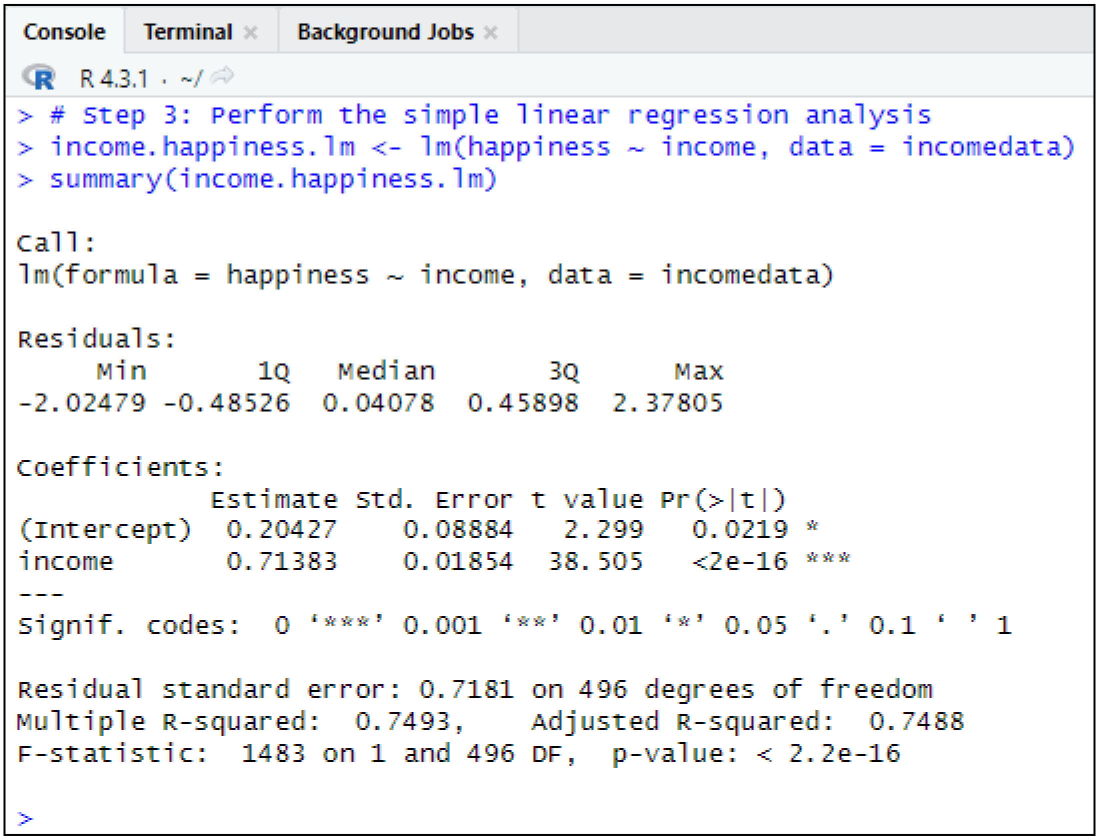
Step 2:



```
> # Step 2: Make sure your data meet the assumptions
> # Normality
> hist(incomedata$happiness)
>
> # Linearity
> plot(happiness ~ income, data = incomedata)
>
```

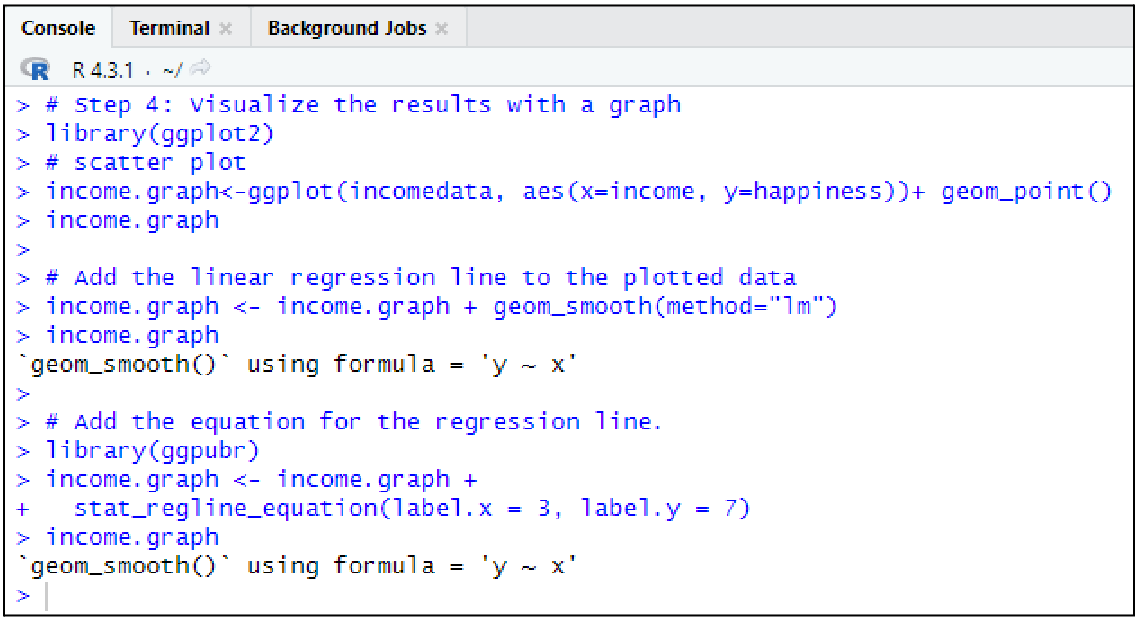


Step 3:

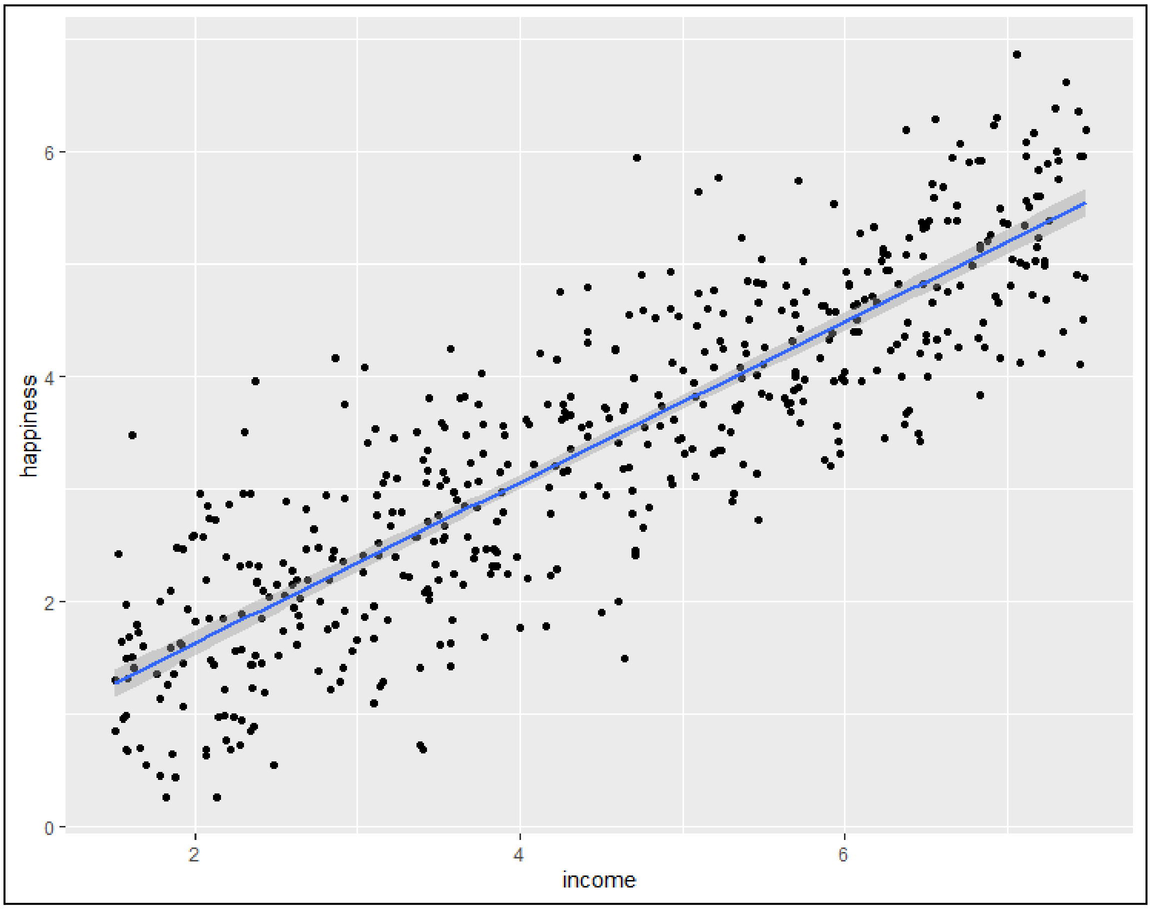


```
> # Step 3: Perform the simple linear regression analysis
> income.happiness.lm <- lm(happiness ~ income, data = incomedata)
> summary(income.happiness.lm)

Call:
lm(formula = happiness ~ income, data = incomedata)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.02479 -0.48526  0.04078  0.45898  2.37805 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  0.20427    0.08884   2.299   0.0219 *  
income       0.71383    0.01854  38.505   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7181 on 496 degrees of freedom
Multiple R-squared:  0.7493,    Adjusted R-squared:  0.7488 
F-statistic:  1483 on 1 and 496 DF,  p-value: < 2.2e-16

>
```

Step 4:



```
> # Step 4: Visualize the results with a graph
> library(ggplot2)
> # scatter plot
> income.graph<-ggplot(incomedata, aes(x=income, y=happiness))+ geom_point()
> income.graph
> 
> # Add the linear regression line to the plotted data
> income.graph <- income.graph + geom_smooth(method="lm")
> income.graph
`geom_smooth()` using formula = 'y ~ x'
> 
> # Add the equation for the regression line.
> library(ggpubr)
> income.graph <- income.graph +
+   stat_regline_equation(label.x = 3, label.y = 7)
> income.graph
`geom_smooth()` using formula = 'y ~ x'
> 
```



Conclusion: : We implemented commands for drawing various Correlation Plots and learnt the process of EDA.