**Exercise No -6**

**Title of Experiment: Statistical Analysis Using R.**

**College: V.P.Institute of Management Studies & Research, Sangli**

**Solution Provider Name: Pranita Anil Kamble**

**#R Version: 3.6.3**

**#R Studio Version: 1.3.1093**

**Abstract:**

**Statistical analysis is implemented by using:**

1) mean and mode

2) covariance

3) standard deviation

4) variance

5) linear regression

6) prediction

Install Necessary Packages:

1. install.packages("reader")

2. install.packages("modeest")

3. install.packages("ggpubr")

4. install.packages("ggplot2")

**Q1) Calculate mean and mode of StudyHours from given table.**

**Introduction:**

1. The **mean**is the [average](https://www.statisticshowto.com/arithmetic-mean/)of a data set.

x = ∑x

N

2. The **mode**is the most common number in a data set.

**Solution:**

> #R Version 3.6.3

> #R Studio Version 1.3.1093

> library(readr)

> library(modeest)

> library (ggpubr)

> library(ggplot2)

> rm(list=ls())

> studmark <- read\_csv("R/studmark.csv")#to load the data

-- -----------------Column specification -------------------------------------------------

cols(

studentNo = col\_double(),

StudyHours = col\_double(),

Marks = col\_double()

)

> cal1.mean <- mean(studmark$StudyHours)#mean function for Studyhours

> print(cal1.mean)

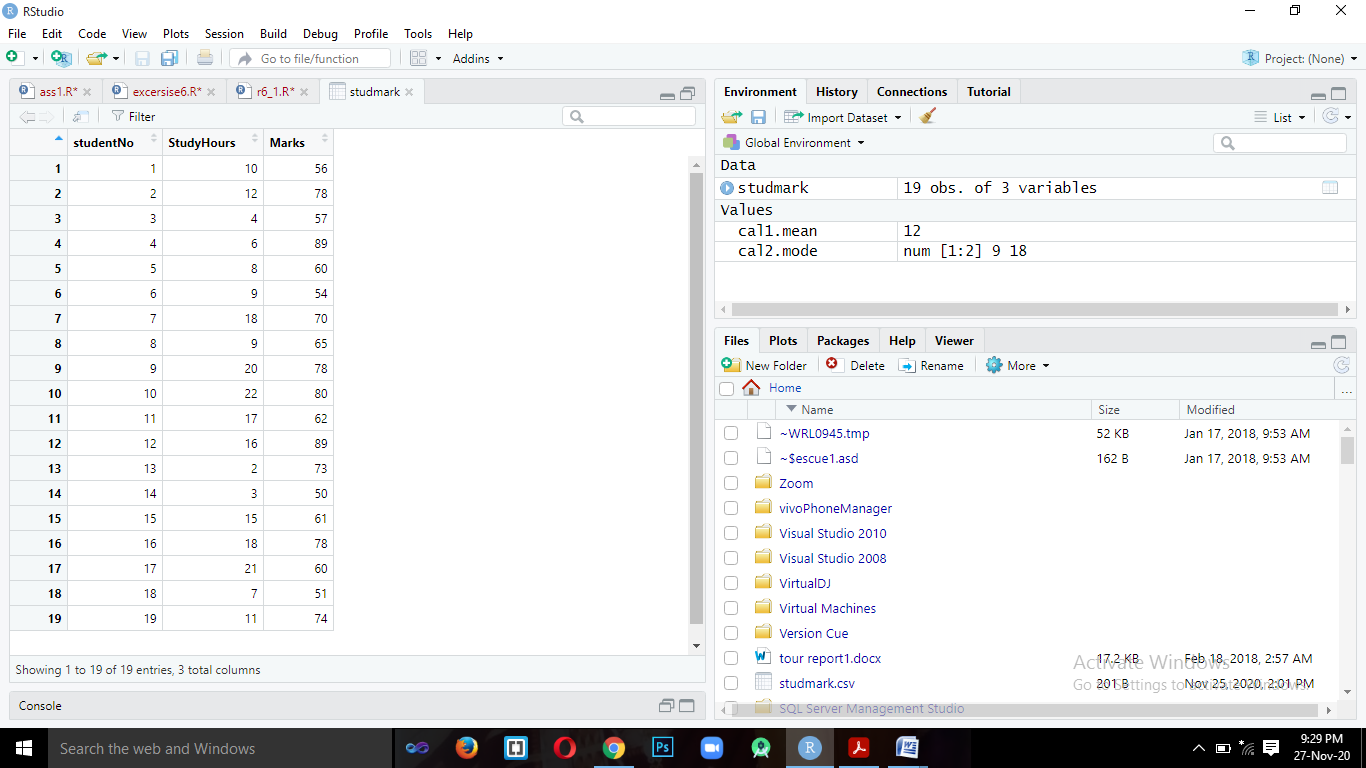
[1] 12

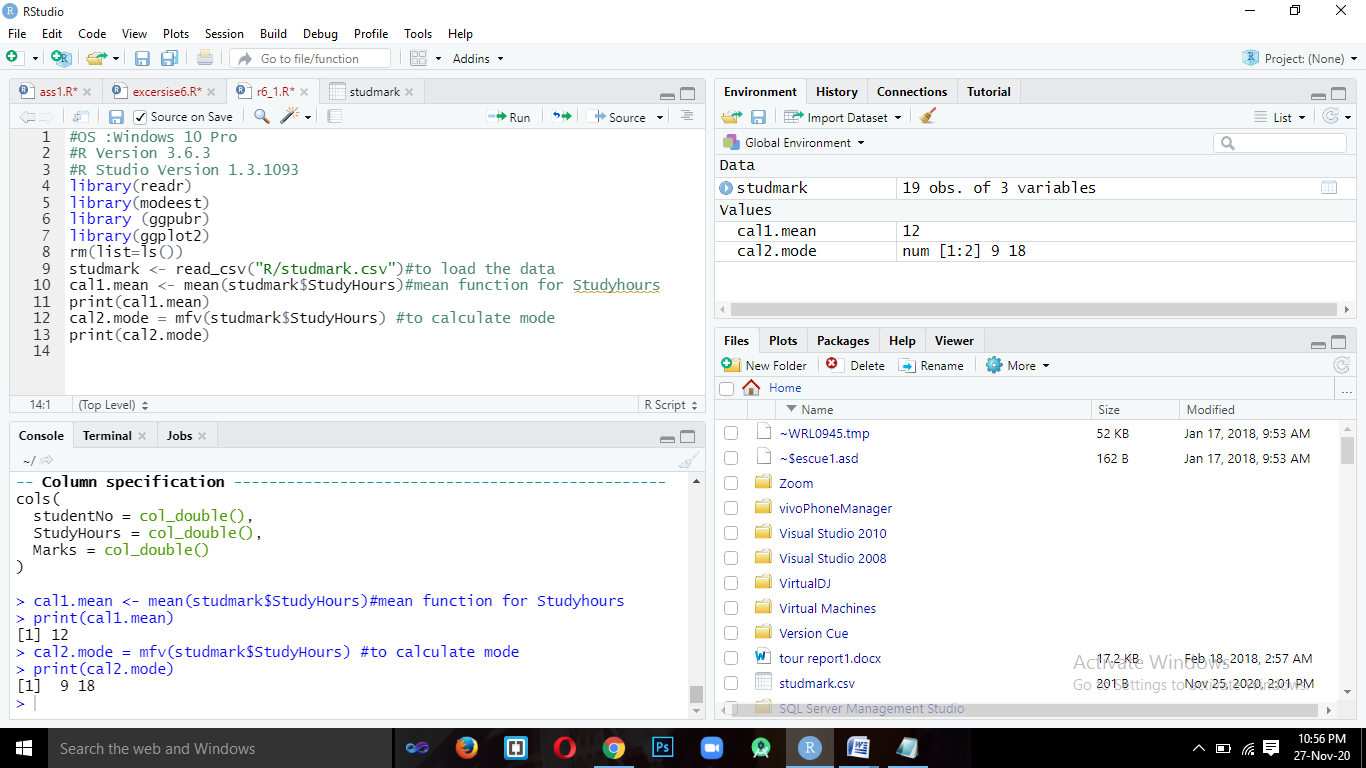
> cal2.mode = mfv(studmark$StudyHours) #to calculate mode

> print(cal2.mode)

[1] 9 18

**Screen:**





**Q2) Calculate covariance between StudyHours and Marks.**

**Explanation:**

Covariance is measure of relationship between two variables.

1) Positive Covariance: Tend to move in same direction.

2) Negative Covariance: Tend to move in inverse direction.

The formula is:  
Cov(X,Y) = Σ E((X-μ)E(Y-ν)) / n-1 where:  
X is a random variable  
E(X) = μ is the expected value (the [mean](https://www.statisticshowto.com/mean/)) of the random variable X and  
E(Y) = ν is the [expected value](https://www.statisticshowto.com/probability-and-statistics/expected-value/) (the mean) of the random variable Y  
n = the number of items in the data set

**Solution:**

> #OS :Windows 10 Pro

> #R Version 3.6.3

> #R Studio Version 1.3.1093

> library(readr)

> library(modeest)

> library (ggpubr)

> library(ggplot2)

> rm(list=ls())

> studmark <- read\_csv("R/studmark.csv") #to load data

-- ----------------------Column specification -------------------------------------------------

cols(

studentNo = col\_double(),

StudyHours = col\_double(),

Marks = col\_double()

)

> StudyHours = studmark$StudyHours

> Marks = studmark$Marks

> cov(StudyHours, Marks) #find covariance between StudyHours and Marks

[1] 27.38889

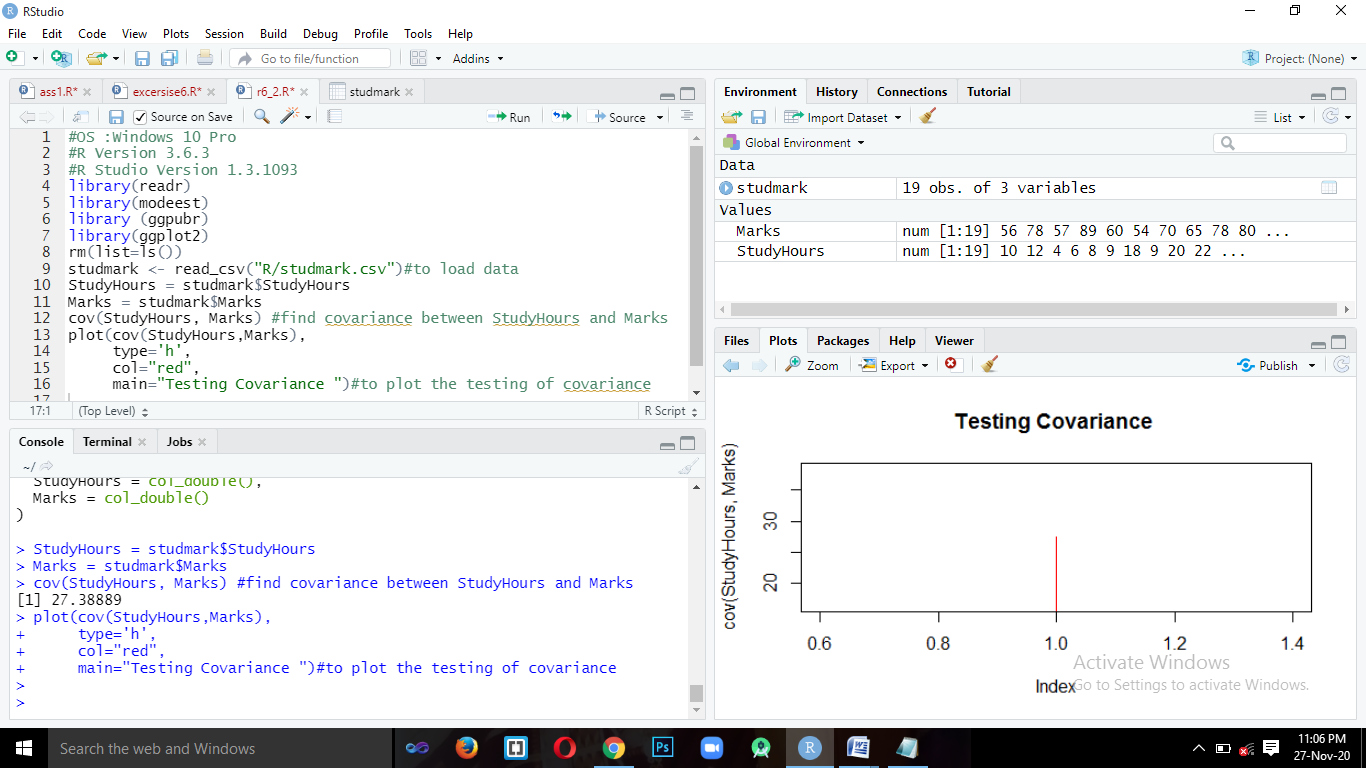
> plot (cov(StudyHours,Marks),

+ type='h',

+ col="red",

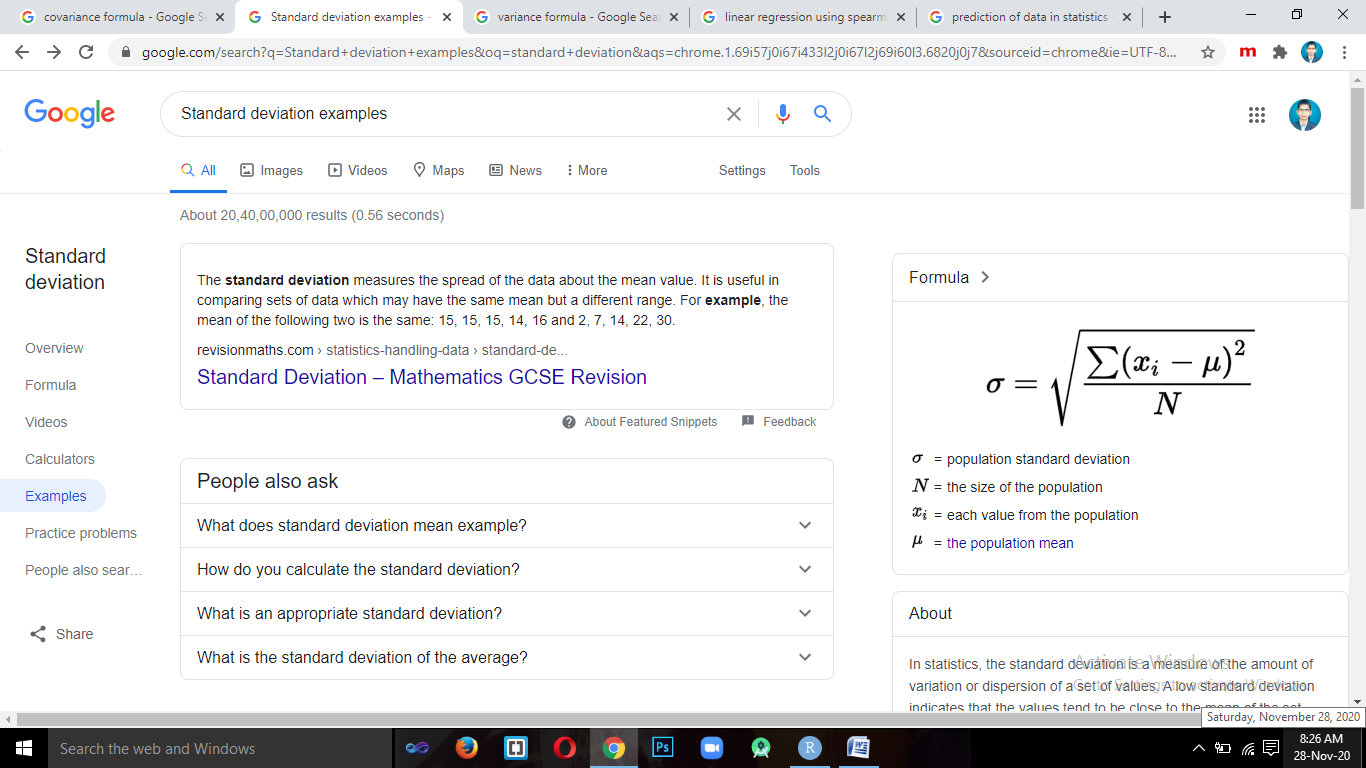
+ main="Testing Covariance ") #to plot the testing of covariance

**Screen:**

 **Q3)Calculate Standard Deviation of marks obtained by students.**

**Explanation:** The standard deviation measures the spread of the data about the mean value. It is useful in comparing sets of data which may have the same mean but a different range.

**Formula:**

****

**Solution:**

#OS :Windows 10 Pro

> #R Version 3.6.3

> #R Studio Version 1.3.1093

> library(readr)

> library(modeest)

> library (ggpubr)

> library(ggplot2)

> rm(list=ls())

> studmark <- read\_csv("R/studmark.csv") #to load data

-- Column specification -------------------------------------------------

cols(

studentNo = col\_double(),

StudyHours = col\_double(),

Marks = col\_double()

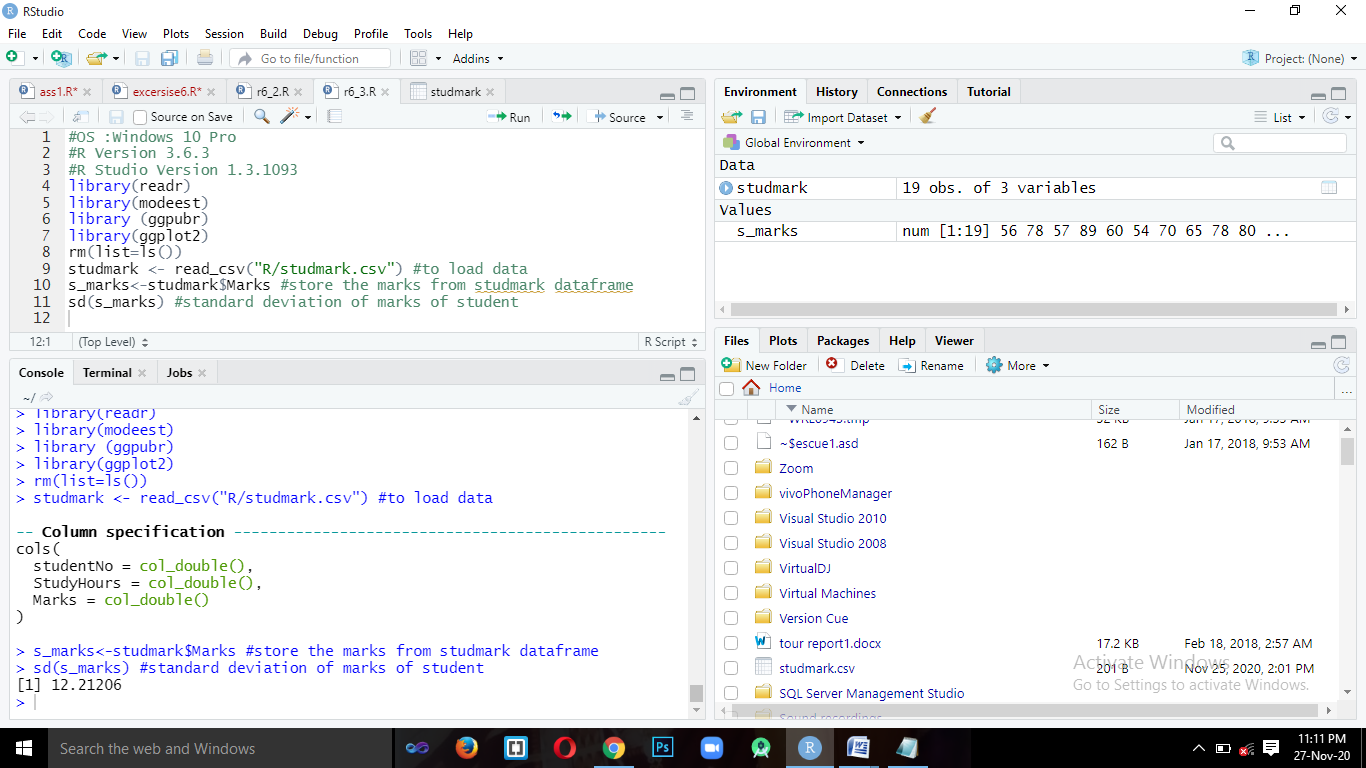
)

> s\_marks<-studmark$Marks #store the marks from studmark dataframe

> sd(s\_marks) #standard deviation of marks of student

[1] 12.21206

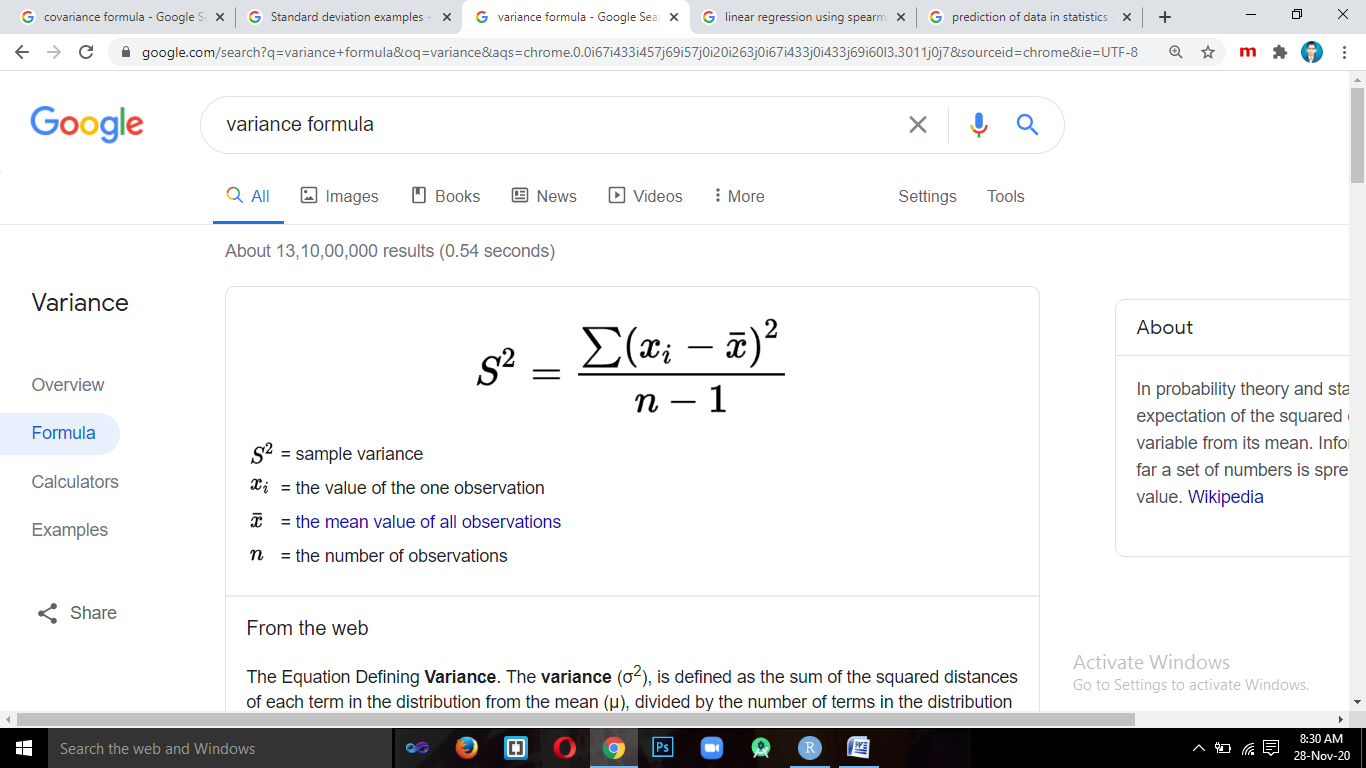
**Screen:**



**Q4) Calculate variance of StudyHours from above data.**

**Explanation:** Variance is used to measure of how data is spread in the dataset. It is calculated as the average squared deviation of each number from the mean of a data set.

**Formula:**

****

**Solution:**

> #OS :Windows 10 Pro

> #R Version 3.6.3

> #R Studio Version 1.3.1093

> library(readr)

> library(modeest)

> library (ggpubr)

> library(ggplot2)

> rm(list=ls())

> studmark <- read\_csv("R/studmark.csv") #to load data

----------- Column specification -------------------------------------------------

cols(

studentNo = col\_double(),

StudyHours = col\_double(),

Marks = col\_double()

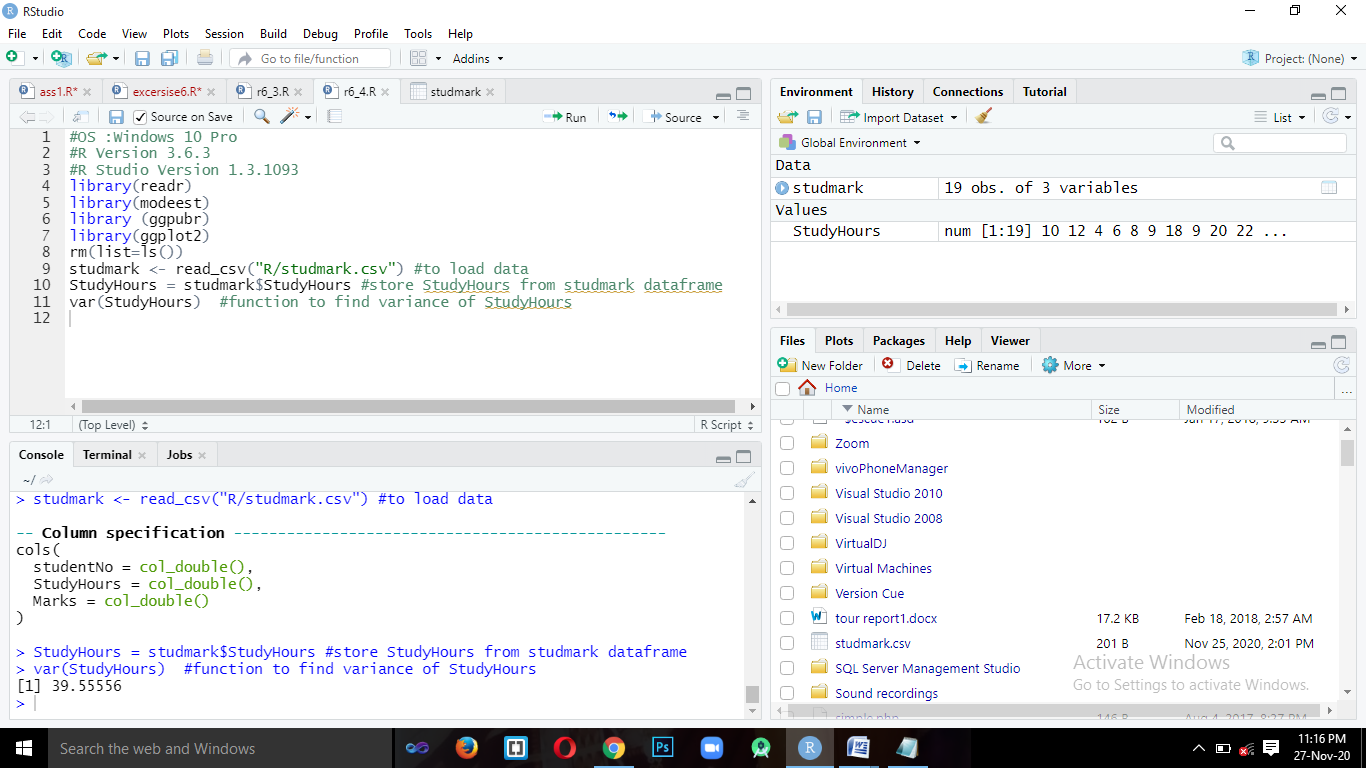
)

> StudyHours = studmark$StudyHours #store StudyHours from studmark dataframe

> var(StudyHours) #function to find variance of StudyHours

[1] 39.55556

**Screen:**



**Q5) Determine the relation between StudyHours and marks obtained by students using linear regression.**

**Explanation:** Regression is a statistical method used in finance, investing, and other disciplines that attempts to determine the strength and character of the relationship between one dependent variable (usually denoted by Y) and a series of other variables (known as independent variables).

The formula for a regression line is

Y' = bX + A

where Y' is the predicted score, b is the slope of the line, and A is the Y intercept.

 A non-parametric procedure, due to Spearman, is to replace the observations by their ranks in the calculation of the correlation coefficient.

https://www.bmj.com/sites/default/files/imagecache/highwire_fragment_medium/attachments/resources/2011/08/figure1114.jpg

**Solution:**

#OS :Windows 10 Pro

#R Version 3.6.3

#R Studio Version 1.3.1093

>library(readr)

>library(modeest)

>library (ggpubr)

>library(ggplot2)

>rm(list=ls())

>studmark <- read\_csv("R/studmark.csv") #to load data

>ggplot(studmark,aes(x=StudyHours, y = Marks))+

>geom\_point() #plot graph of StudyHours Vs Marks

>abline(lm(studmark$Marks~ studmark$StudyHours)) #to add regression line

>lm(studmark$Marks~studmark$StudyHours) #to create a regression model

>plot(StudyHours,Marks)

>cal3.correlation <-cor.test(studmark$StudyHours, studmark$Marks, method = "spearman")

#calculating the correlation between StudyHours and Marks using spearman method

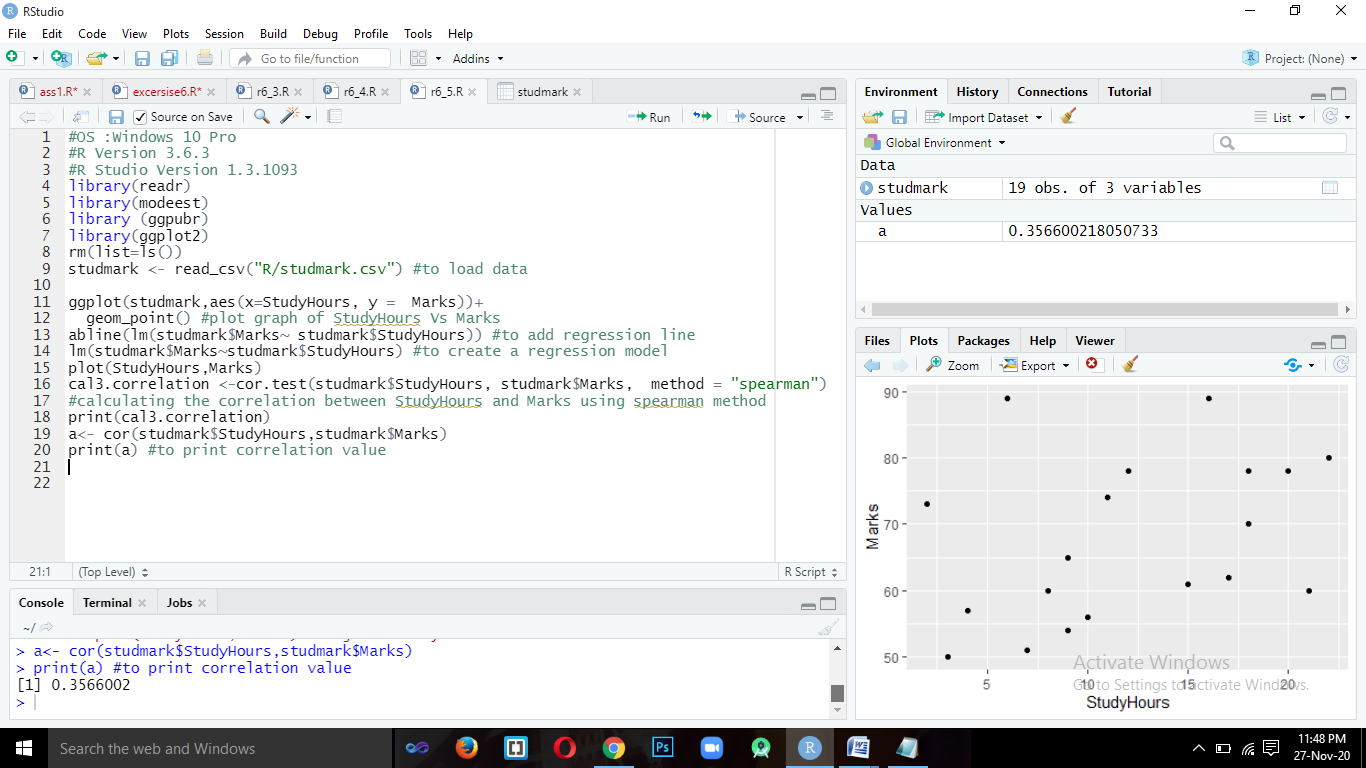
>print(cal3.correlation)

>a<- cor(studmark$StudyHours,studmark$Marks)

>print(a) #to print correlation value

>[1] 0.3566002

**Screen:**



**Q6) Predict the marks of students who study for 14 hours in a week.**

**Explanation:**

Predictions are precise when the observed values cluster close to the predicted values. Regression predictions are for the mean of the dependent variable. If you think of any mean, you know that there is variation around that mean. The same applies to the predicted mean of the dependent variable.

**Solution:**

> #OS :Windows 10 Pro

> #R Version 3.6.3

> #R Studio Version 1.3.1093

> library(readr)

> library(modeest)

> library (ggpubr)

> library(ggplot2)

> #rm(list=ls())

> studmark <- read\_csv("R/studmark.csv") #to load data

-- Column specification -----------------------------------------------------------------

cols(

studentNo = col\_double(),

StudyHours = col\_double(),

Marks = col\_double()

)

> # Load the data

> #data("studmark", package = "dataframes")

> # Build the model

> model <- lm(Marks~ StudyHours, data = studmark)#summary of dataset

> model

Call:

lm(formula = Marks ~ StudyHours, data = studmark)

Coefficients:

(Intercept) StudyHours

59.3226 0.6924

> new.StudyHours <- data.frame(

+ StudyHours = c(14)

+ )#new data for prediction

> predict(model, newdata = new.StudyHours)

1

69.01641

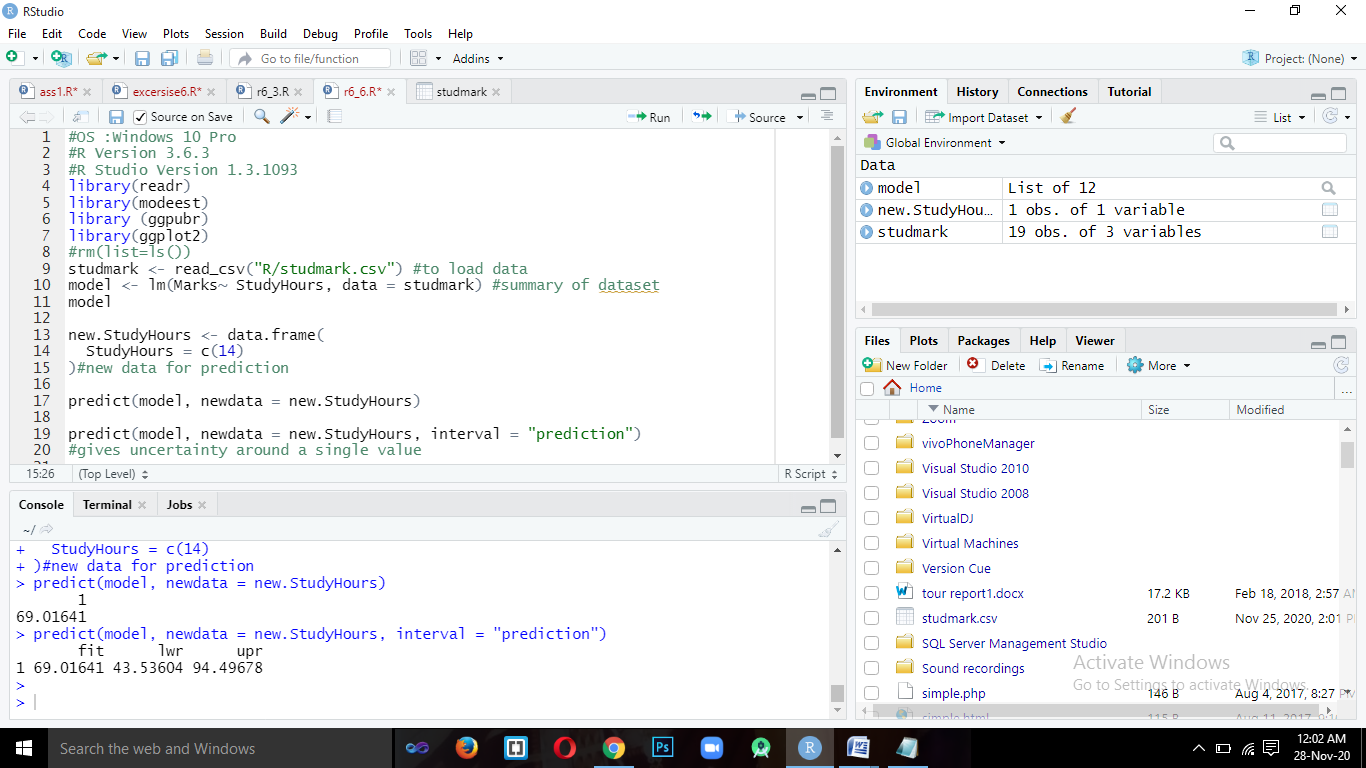
> predict(model, newdata = new.StudyHours, interval = "prediction")

fit lwr upr

1 69.01641 43.53604 94.49678

>

**Screen:**



xxxxxxxxxxxxxxxxxx