Angeles City Science High School

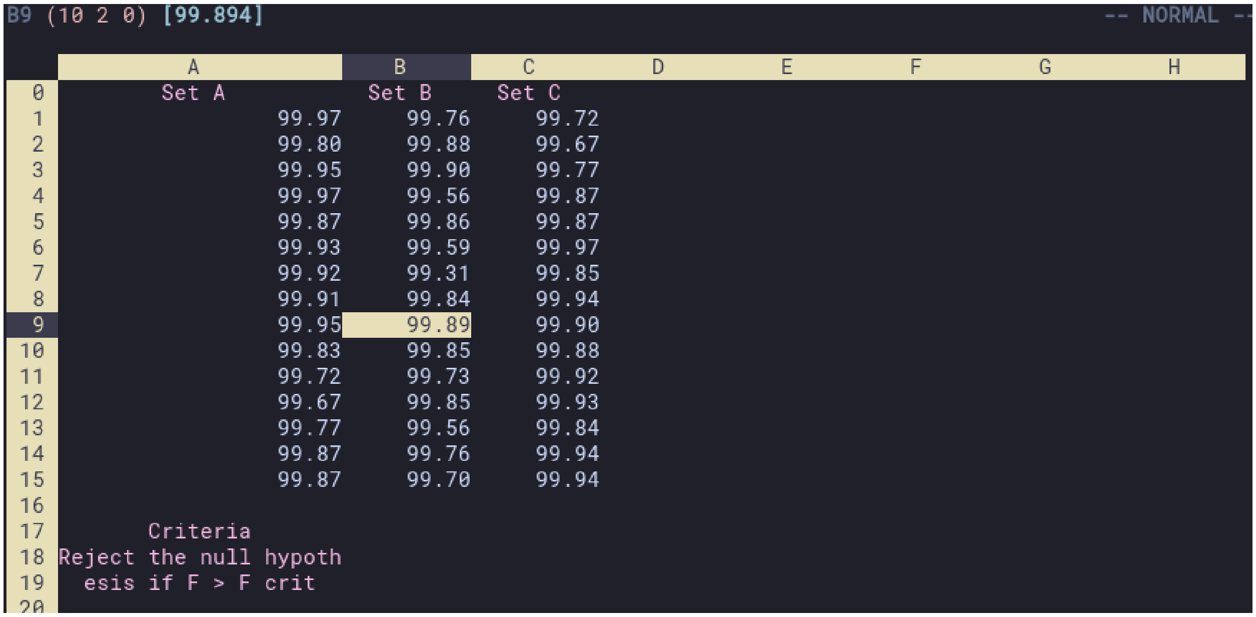
Research 10

Name: Paul Gerald D. Pare **Section:** 10-Hawking

Ma'am sorry po pero hindi po ako makapag-install ng excel since nasa linux po ako :D. Ginawa ko nalang po siya sa R instead of excel po.

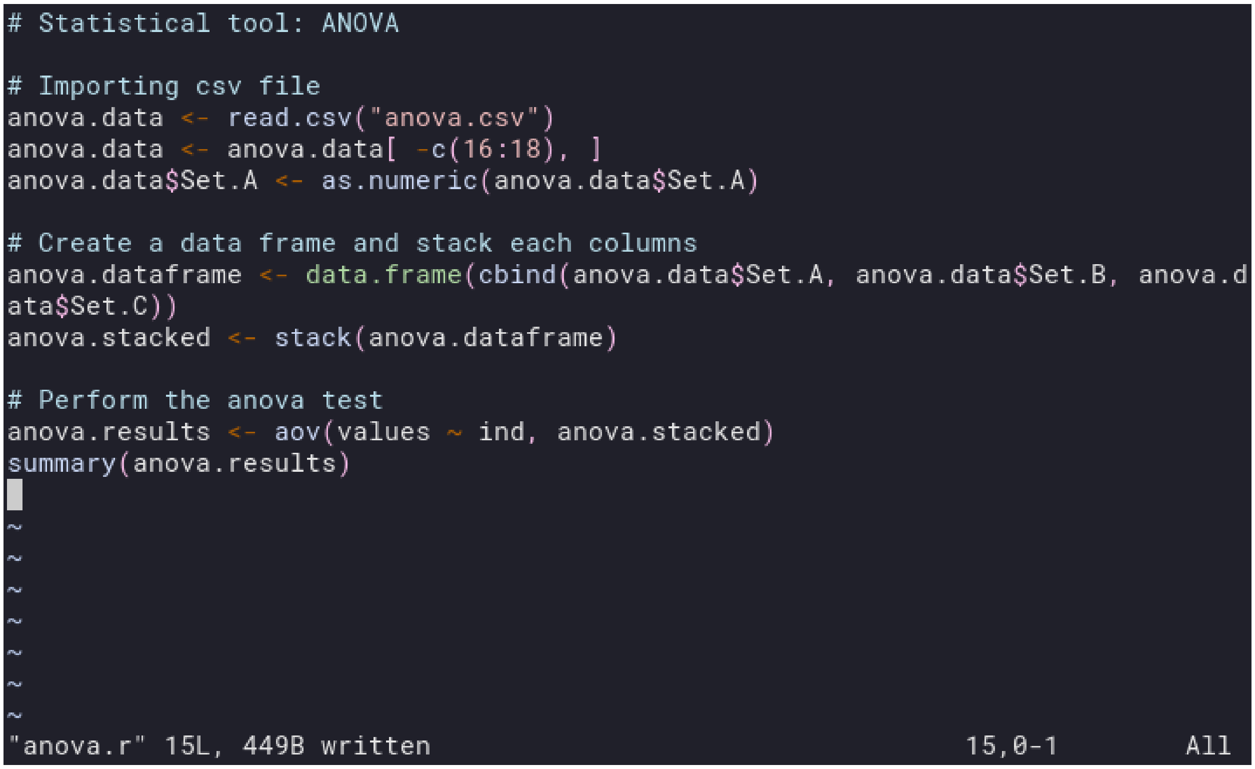
Anova:

Data in a spreadsheet:



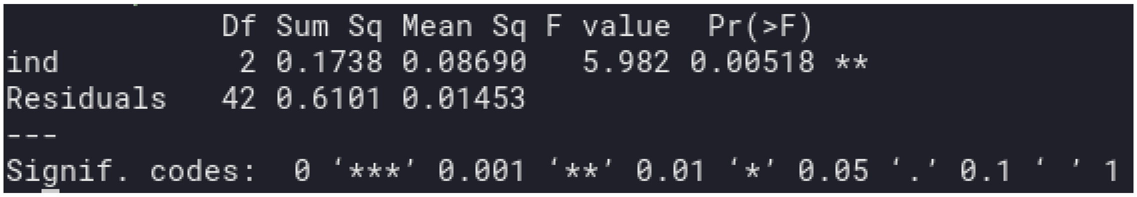
```
B9 (10 2 0) [99.894]                                                                        -- NORMAL --

                      A                 B          C          D          E          F          G          H
 0             Set A                  Set B      Set C
 1                       99.97      99.76      99.72
 2                       99.80      99.88      99.67
 3                       99.95      99.90      99.77
 4                       99.97      99.56      99.87
 5                       99.87      99.86      99.87
 6                       99.93      99.59      99.97
 7                       99.92      99.31      99.85
 8                       99.91      99.84      99.94
 9                       99.95      99.89      99.90
10                       99.83      99.85      99.88
11                       99.72      99.73      99.92
12                       99.67      99.85      99.93
13                       99.77      99.56      99.84
14                       99.87      99.76      99.94
15                       99.87      99.70      99.94
16
17           Criteria
18   Reject the null hypoth
19     esis if F > F crit
20
```

Code:



```
# Statistical tool: ANOVA

# Importing csv file
anova.data <- read.csv("anova.csv")
anova.data <- anova.data[ -c(16:18), ]
anova.data$Set.A <- as.numeric(anova.data$Set.A)

# Create a data frame and stack each columns
anova.dataframe <- data.frame(cbind(anova.data$Set.A, anova.data$Set.B, anova.d
ata$Set.C))
anova.stacked <- stack(anova.dataframe)

# Perform the anova test
anova.results <- aov(values ~ ind, anova.stacked)
summary(anova.results)

~
~
~
~
~
~
~
"anova.r" 15L, 449B written                                   15,0-1        All
```

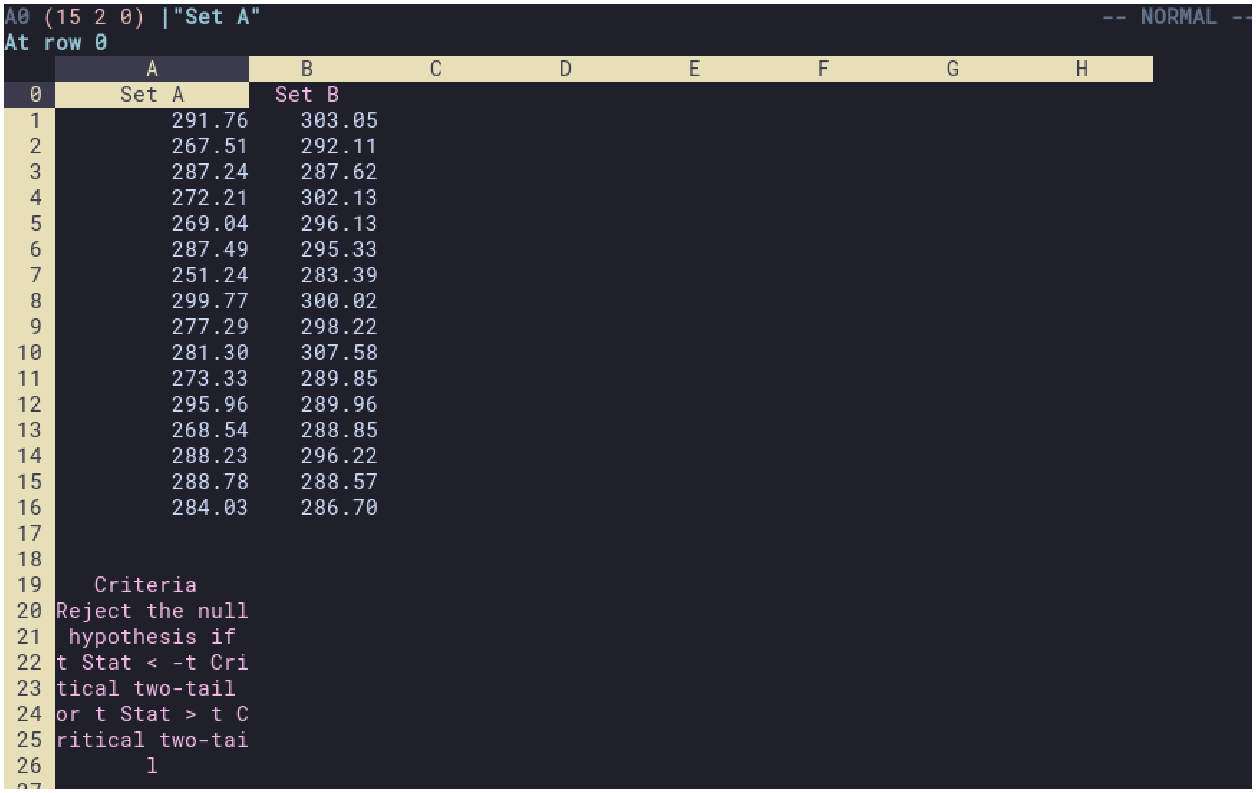
Result:



```
            Df Sum Sq Mean Sq F value  Pr(>F)
ind          2 0.1738 0.08690   5.982 0.00518 **
Residuals   42 0.6101 0.01453
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

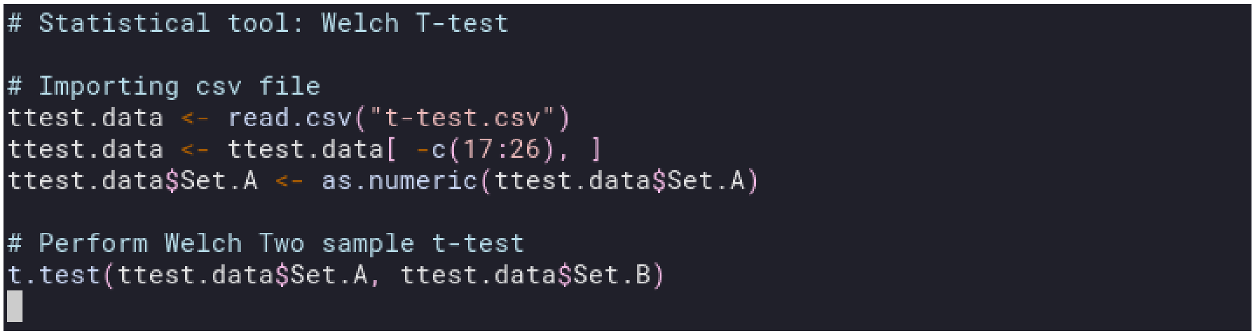
T-test:

Data in a spreadsheet:



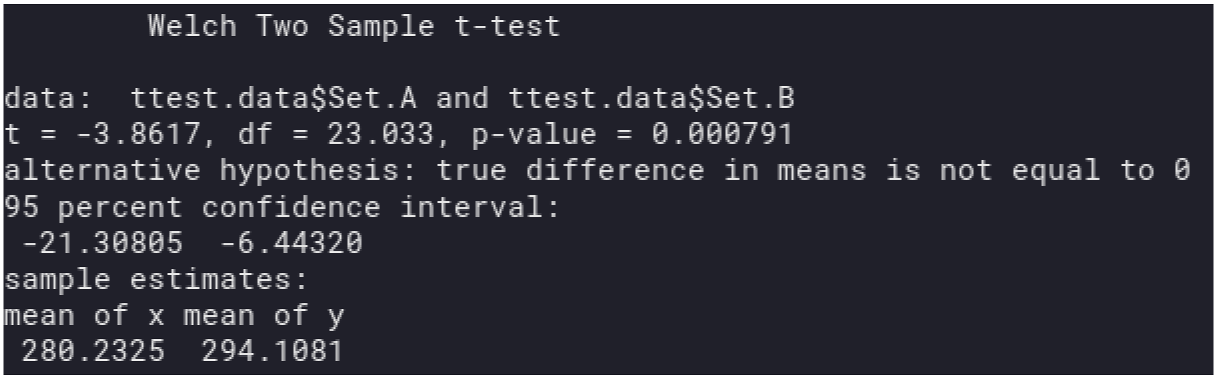
```
A0 (15 2 0) |"Set A"                                                                    -- NORMAL --
At row 0
                A           B          C          D          E          F          G          H
 0           Set A       Set B
 1          291.76    303.05
 2          267.51    292.11
 3          287.24    287.62
 4          272.21    302.13
 5          269.04    296.13
 6          287.49    295.33
 7          251.24    283.39
 8          299.77    300.02
 9          277.29    298.22
10          281.30    307.58
11          273.33    289.85
12          295.96    289.96
13          268.54    288.85
14          288.23    296.22
15          288.78    288.57
16          284.03    286.70
17
18
19     Criteria
20  Reject the null
21   hypothesis if
22  t Stat < -t Cri
23  tical two-tail
24  or t Stat > t C
25  ritical two-tai
26         l
```

Code:



```r
# Statistical tool: Welch T-test

# Importing csv file
ttest.data <- read.csv("t-test.csv")
ttest.data <- ttest.data[ -c(17:26), ]
ttest.data$Set.A <- as.numeric(ttest.data$Set.A)

# Perform Welch Two sample t-test
t.test(ttest.data$Set.A, ttest.data$Set.B)
```

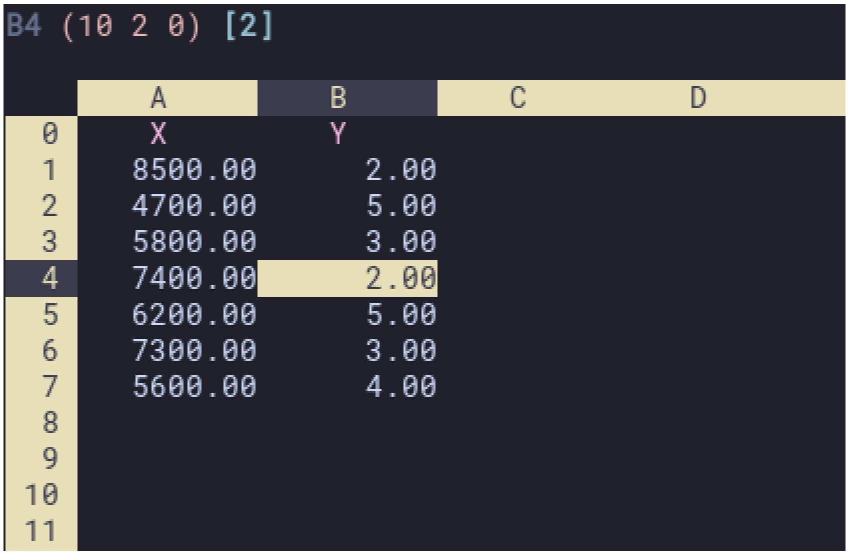
Result:



```
        Welch Two Sample t-test

data:  ttest.data$Set.A and ttest.data$Set.B
t = -3.8617, df = 23.033, p-value = 0.000791
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -21.30805  -6.44320
sample estimates:
mean of x mean of y
 280.2325  294.1081
```

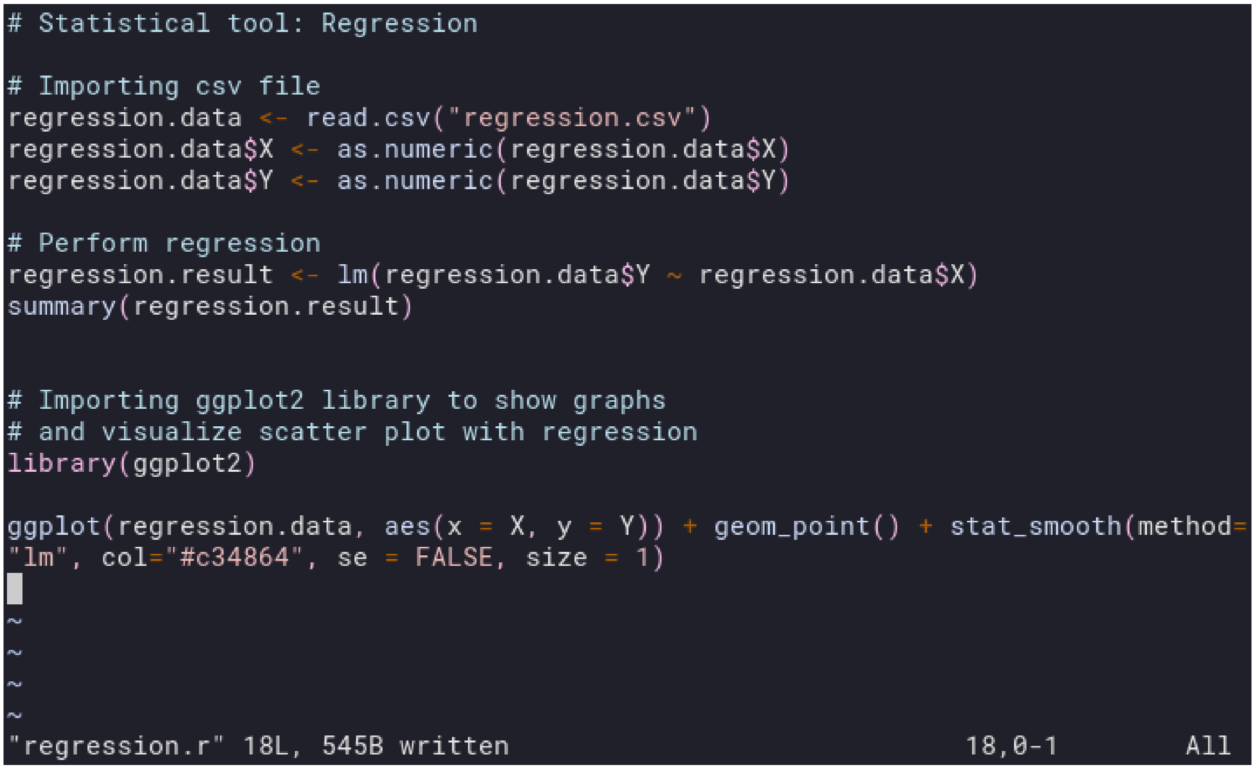
Regression:

Data in a spreadsheet:



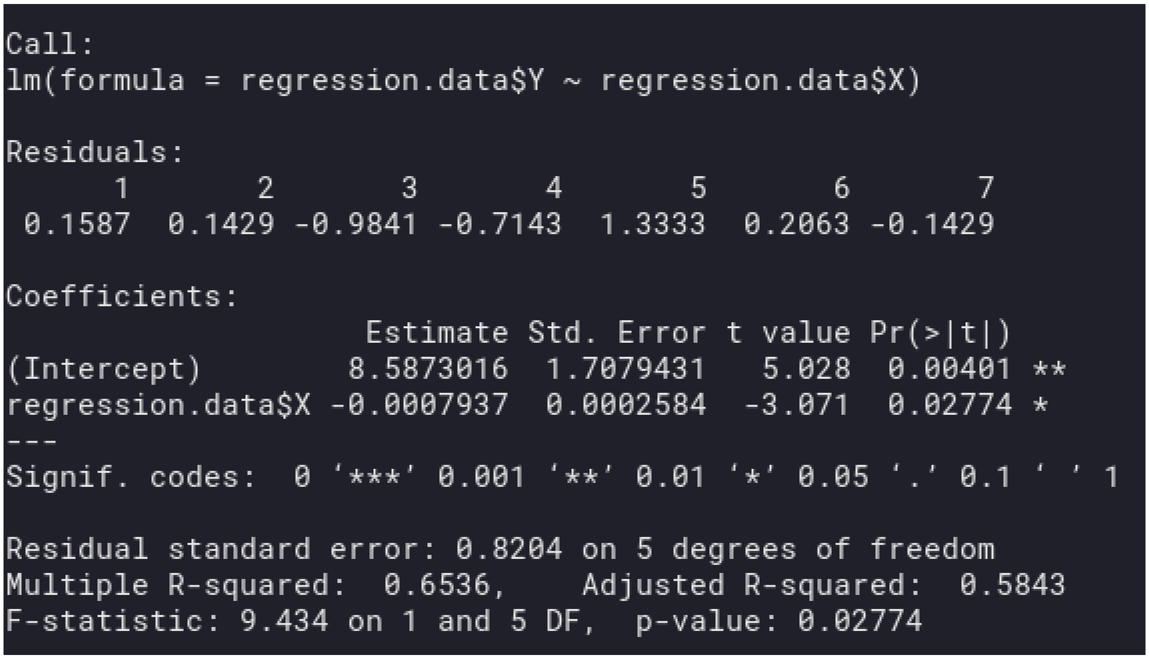
```
B4 (10 2 0) [2]

         A         B         C         D
 0       X         Y
 1    8500.00      2.00
 2    4700.00      5.00
 3    5800.00      3.00
 4    7400.00      2.00
 5    6200.00      5.00
 6    7300.00      3.00
 7    5600.00      4.00
 8
 9
10
11
```

Code:



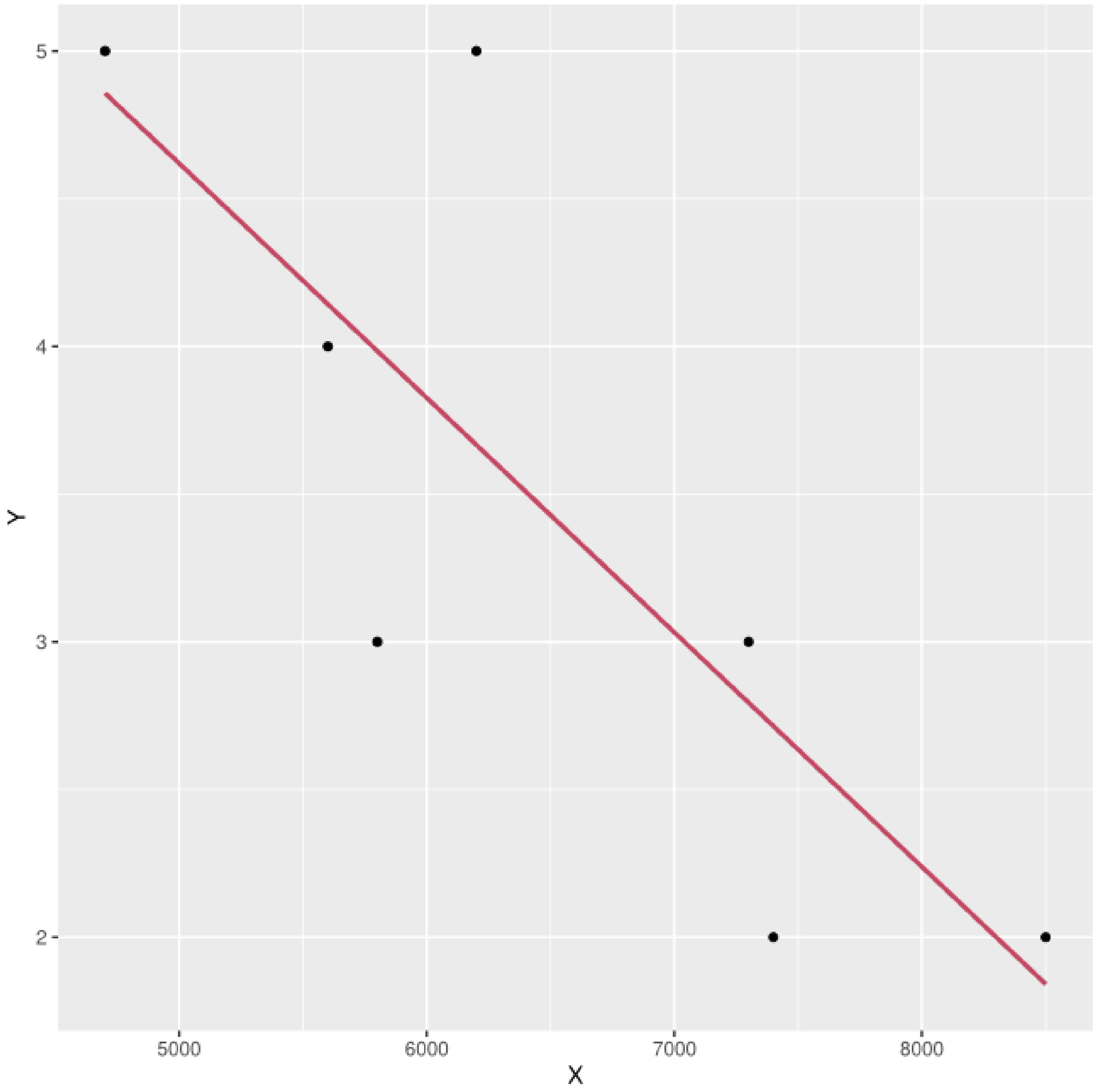
```
# Statistical tool: Regression

# Importing csv file
regression.data <- read.csv("regression.csv")
regression.data$X <- as.numeric(regression.data$X)
regression.data$Y <- as.numeric(regression.data$Y)

# Perform regression
regression.result <- lm(regression.data$Y ~ regression.data$X)
summary(regression.result)


# Importing ggplot2 library to show graphs
# and visualize scatter plot with regression
library(ggplot2)

ggplot(regression.data, aes(x = X, y = Y)) + geom_point() + stat_smooth(method=
"lm", col="#c34864", se = FALSE, size = 1)

~
~
~
~
"regression.r" 18L, 545B written                                18,0-1        All
```

Result (w/o graph):



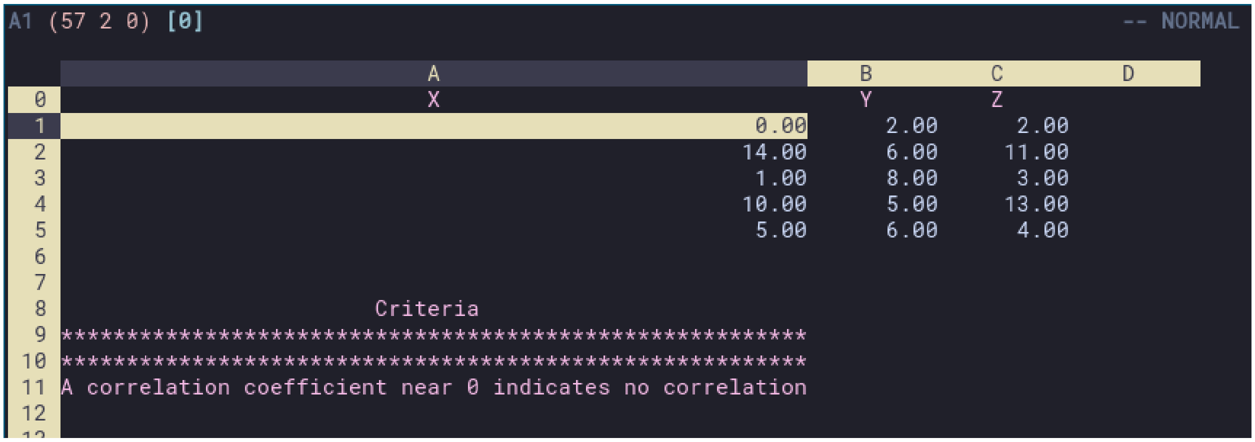
```
Call:
lm(formula = regression.data$Y ~ regression.data$X)

Residuals:
      1       2       3       4       5       6       7 
 0.1587  0.1429 -0.9841 -0.7143  1.3333  0.2063 -0.1429 

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)   
(Intercept)        8.5873016  1.7079431   5.028  0.00401 **
regression.data$X -0.0007937  0.0002584  -3.071  0.02774 * 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.8204 on 5 degrees of freedom
Multiple R-squared:  0.6536,    Adjusted R-squared:  0.5843 
F-statistic: 9.434 on 1 and 5 DF,  p-value: 0.02774
```

Graph:



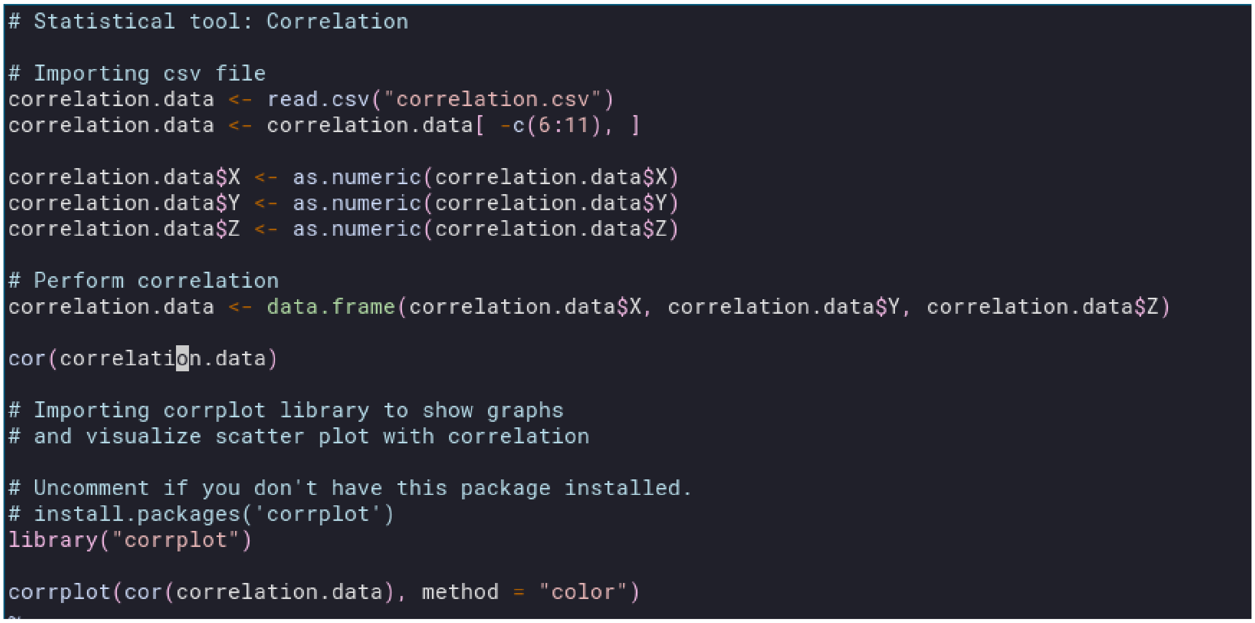
Correlation:

Data in a spreadsheet:



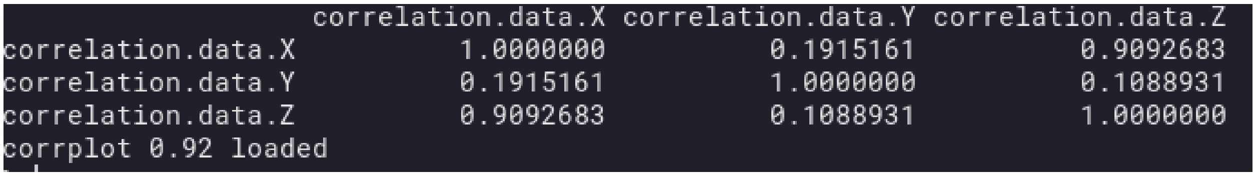
```
A1 (57 2 0) [0]                                                                                       -- NORMAL

                                A                                  B          C          D
 0                              X                                  Y          Z
 1                                                      0.00      2.00       2.00
 2                                                     14.00      6.00      11.00
 3                                                      1.00      8.00       3.00
 4                                                     10.00      5.00      13.00
 5                                                      5.00      6.00       4.00
 6
 7
 8                          Criteria
 9 **********************************************************
10 **********************************************************
11 A correlation coefficient near 0 indicates no correlation
12
```

Code:



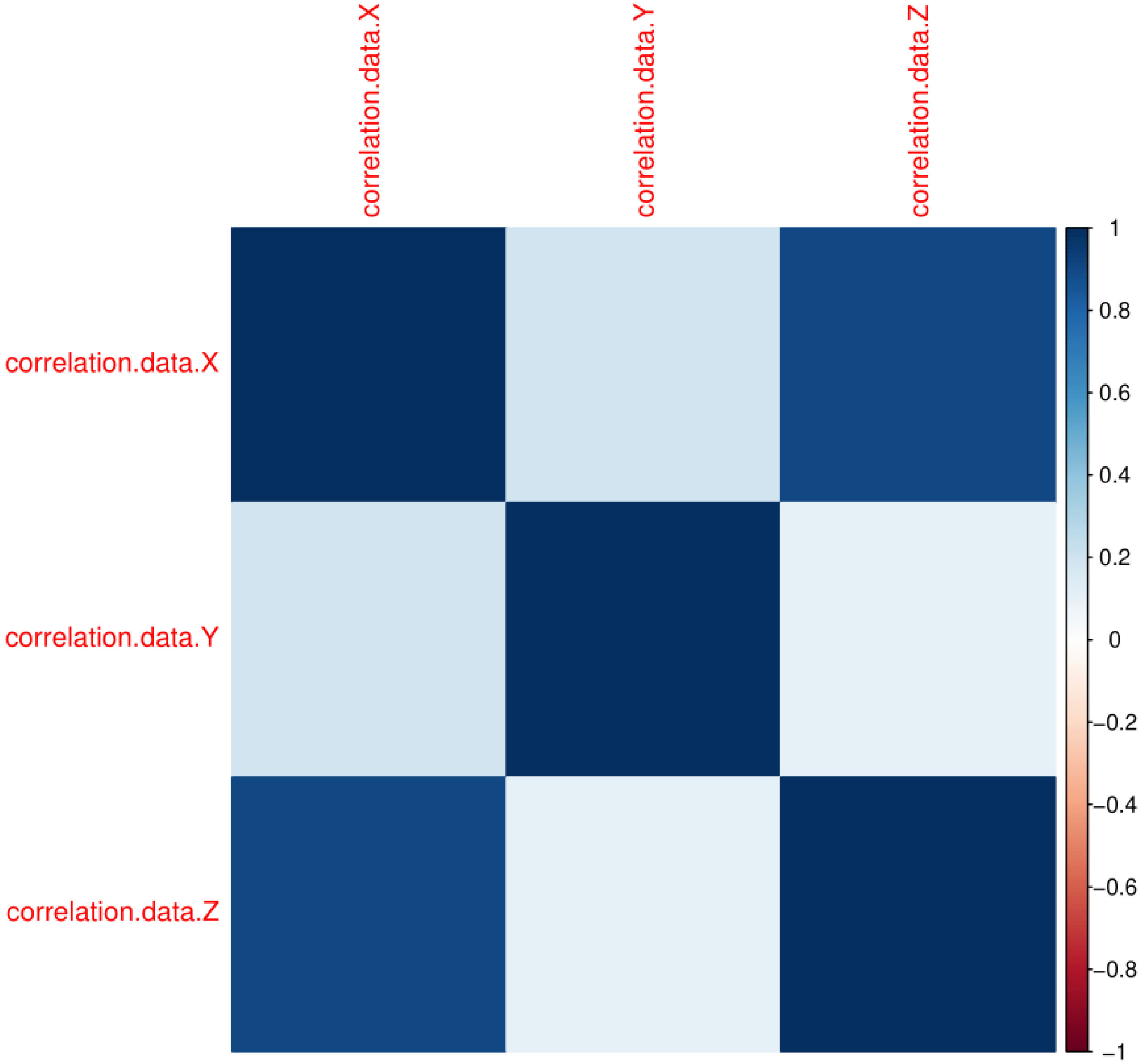
```
# Statistical tool: Correlation

# Importing csv file
correlation.data <- read.csv("correlation.csv")
correlation.data <- correlation.data[ -c(6:11), ]

correlation.data$X <- as.numeric(correlation.data$X)
correlation.data$Y <- as.numeric(correlation.data$Y)
correlation.data$Z <- as.numeric(correlation.data$Z)

# Perform correlation
correlation.data <- data.frame(correlation.data$X, correlation.data$Y, correlation.data$Z)

cor(correlation.data)

# Importing corrplot library to show graphs
# and visualize scatter plot with correlation

# Uncomment if you don't have this package installed.
# install.packages('corrplot')
library("corrplot")

corrplot(cor(correlation.data), method = "color")
```

Result (w/o graph):



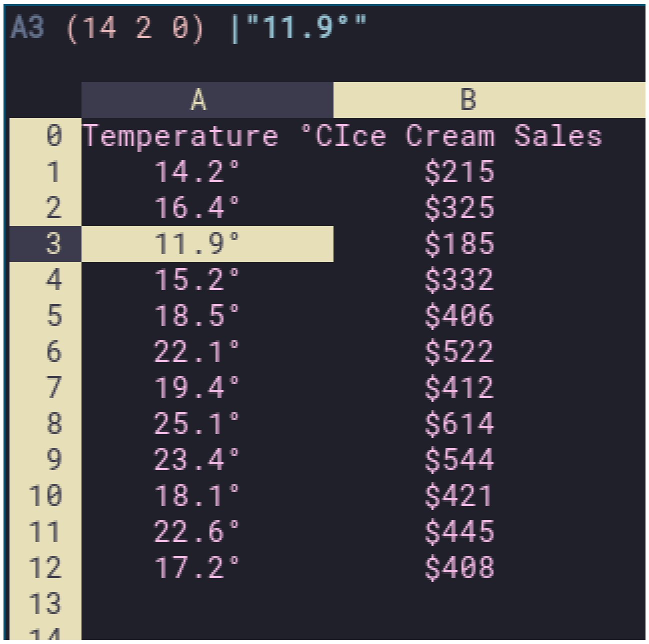
```
                   correlation.data.X correlation.data.Y correlation.data.Z
correlation.data.X          1.0000000          0.1915161          0.9092683
correlation.data.Y          0.1915161          1.0000000          0.1088931
correlation.data.Z          0.9092683          0.1088931          1.0000000
corrplot 0.92 loaded
```

Graph:


correlation.data.X
correlation.data.Y
correlation.data.Z
correlation.data.X
correlation.data.Y
correlation.data.Z
1
0.8
0.6
0.4
0.2
0
−0.2
−0.4
−0.6
−0.8
−1

Scatter Plot:

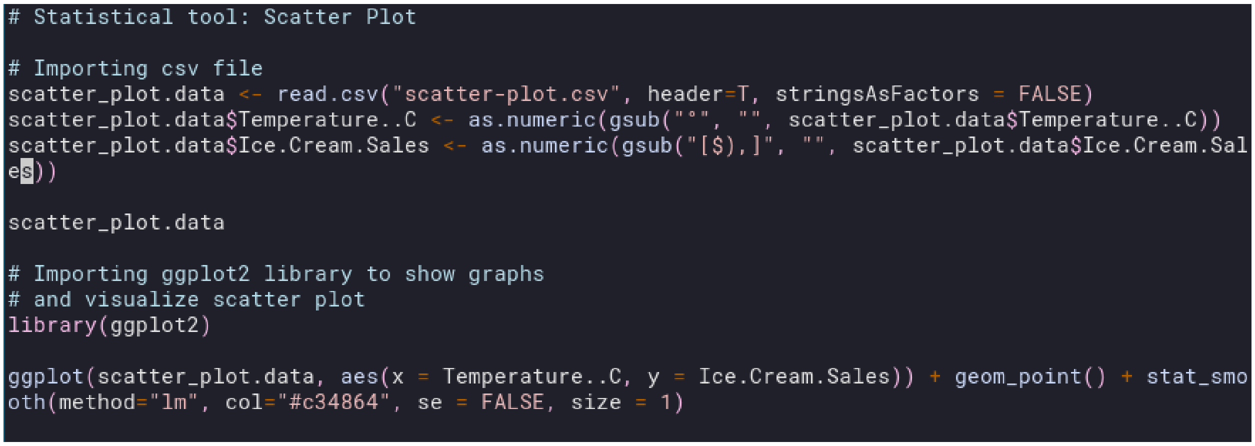
Data in a spreadsheet:



A3 (14 2 0) |"11.9°"

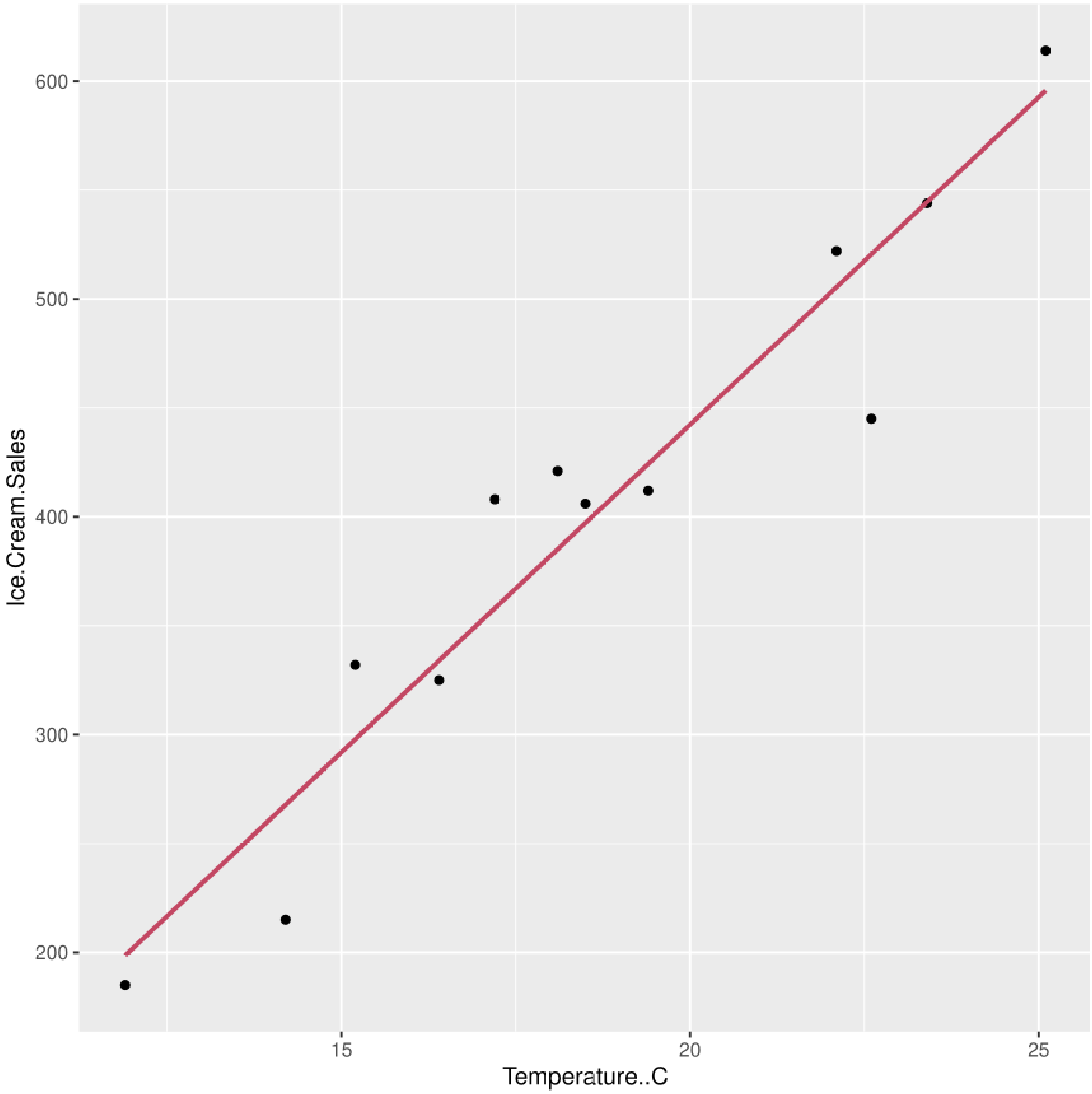
	A	B
0	Temperature °C	Ice Cream Sales
1	14.2°	$215
2	16.4°	$325
3	11.9°	$185
4	15.2°	$332
5	18.5°	$406
6	22.1°	$522
7	19.4°	$412
8	25.1°	$614
9	23.4°	$544
10	18.1°	$421
11	22.6°	$445
12	17.2°	$408
13		

Code:



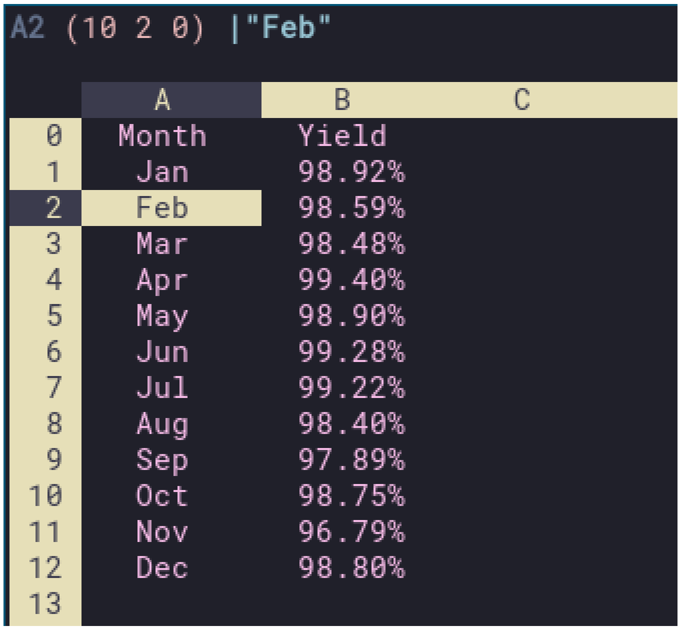
```
# Statistical tool: Scatter Plot

# Importing csv file
scatter_plot.data <- read.csv("scatter-plot.csv", header=T, stringsAsFactors = FALSE)
scatter_plot.data$Temperature..C <- as.numeric(gsub("°", "", scatter_plot.data$Temperature..C))
scatter_plot.data$Ice.Cream.Sales <- as.numeric(gsub("[$),]", "", scatter_plot.data$Ice.Cream.Sal
es))

scatter_plot.data

# Importing ggplot2 library to show graphs
# and visualize scatter plot
library(ggplot2)

ggplot(scatter_plot.data, aes(x = Temperature..C, y = Ice.Cream.Sales)) + geom_point() + stat_smo
oth(method="lm", col="#c34864", se = FALSE, size = 1)
```

Graph:



Run plot:

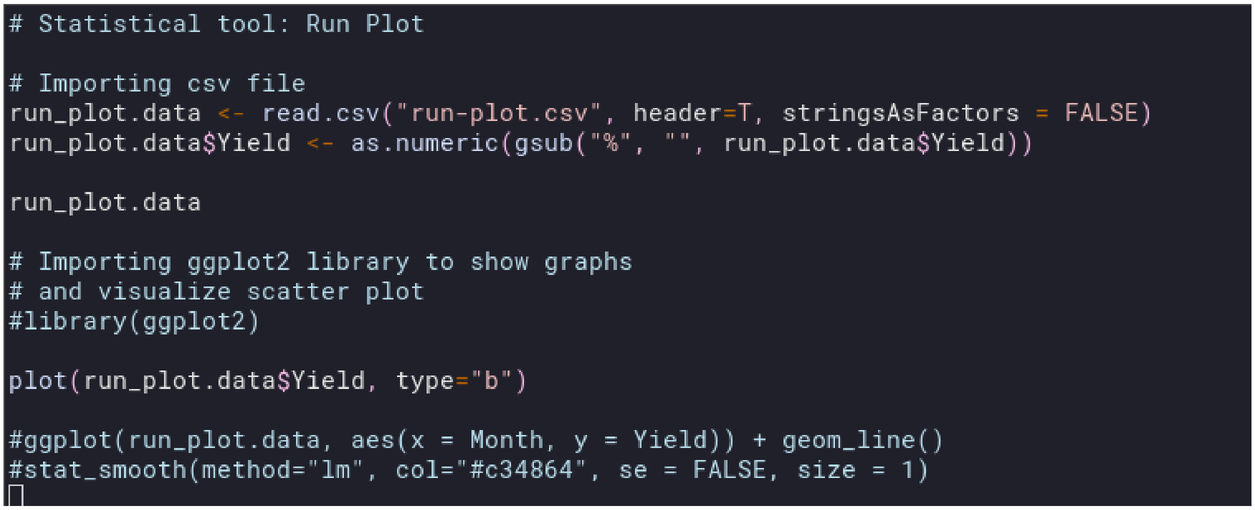
Data in a spreadsheet:



A2 (10 2 0) |"Feb"

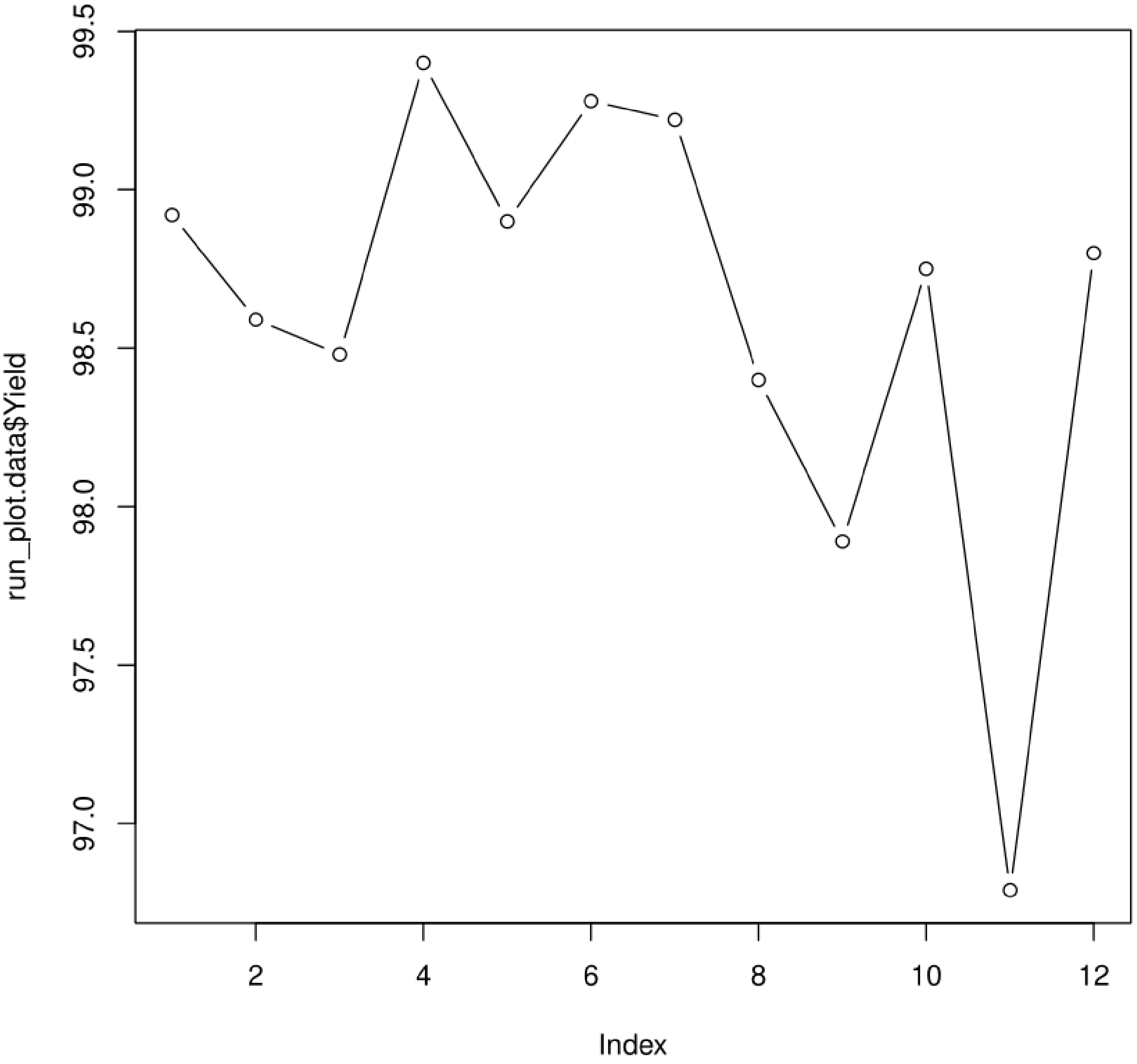
	A	B	C
0	Month	Yield	
1	Jan	98.92%	
2	Feb	98.59%	
3	Mar	98.48%	
4	Apr	99.40%	
5	May	98.90%	
6	Jun	99.28%	
7	Jul	99.22%	
8	Aug	98.40%	
9	Sep	97.89%	
10	Oct	98.75%	
11	Nov	96.79%	
12	Dec	98.80%	
13			

Code:



```
# Statistical tool: Run Plot

# Importing csv file
run_plot.data <- read.csv("run-plot.csv", header=T, stringsAsFactors = FALSE)
run_plot.data$Yield <- as.numeric(gsub("%", "", run_plot.data$Yield))

run_plot.data

# Importing ggplot2 library to show graphs
# and visualize scatter plot
#library(ggplot2)

plot(run_plot.data$Yield, type="b")

#ggplot(run_plot.data, aes(x = Month, y = Yield)) + geom_line()
#stat_smooth(method="lm", col="#c34864", se = FALSE, size = 1)
```

Graph:


99.5
99.0
98.5
98.0
97.5
97.0
run_plot.data$Yield
2
4
6
8
10
12
Index