**STAT 40001/STAT 50001 Statistical Computing Fall 2024**

**Lab -14**

1. In order to find the relationship between the number of hours student study outside the class and their test score we have collected a sample of 10 students and observe their test score and the number of hours they spend outside the classroom.

Student Score # of hours/Week

Andrea 63 3

Ben 79 11

Randy 53 8

Jamie 77 10

Emily 89 13

Darryl 66 5

Joseph 64 4

Eleanor 91 14

Jacob 71 5

Sharon 74 9

***Shift + Alt -> then ctrl + c***

1. Calculate the Pearson correlation coefficient.
2. Calculate the Spearman correlation Coefficient
3. Calculate the Kendall’ s Tau.
4. Using each method test the hypothesis that the correlation is nonzero.

> #Q1

> y = scan()

1: 3 11 8 10 13 5 4 14 5 9

11:

Read 10 items

> # 3 11 8 10 13 5 4 14 5 9

> y

[1] 3 11 8 10 13 5 4 14 5 9

> x = scan()

1: 63 79 53 77 89 66 64 91 71 74

11:

Read 10 items

> # 63 79 53 77 89 66 64 91 71 74

> x

[1] 63 79 53 77 89 66 64 91 71 74

> x;y

[1] 63 79 53 77 89 66 64 91 71 74

[1] 3 11 8 10 13 5 4 14 5 9

> plot(x,y, pch = 17, lwd = 2)

> cor(x,y)

[1] 0.7989697

> # or cor(x,y,method = p) or cor(x,y, method = "pearson")

> cor(x,y,method = "s")

[1] 0.875384

> cor(x,y,method = "k")

[1] 0.8090398

> cor.test(x,y, method = "p")$p.value

[1] 0.005561874

> cat("Reject the null hypothesis i.e. We have enough evidence to say that they are correleated")

Reject the null hypothesis i.e. We have enough evidence to say that they are correleated

> shapiro.test(y)

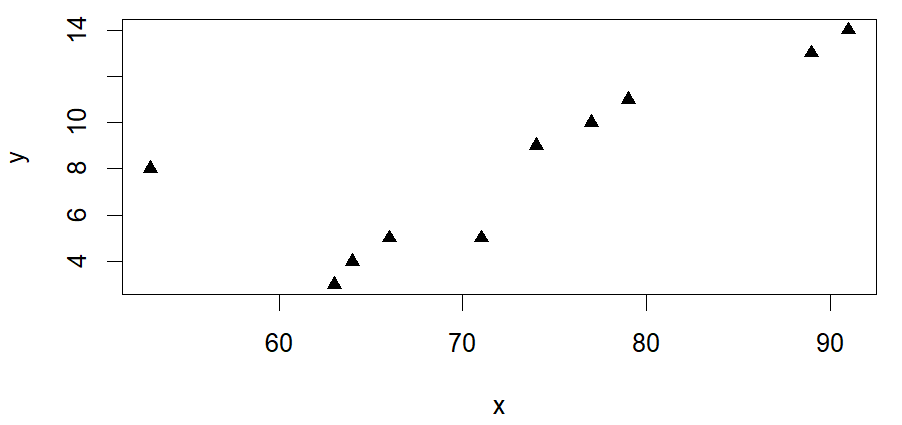
Shapiro-Wilk normality test

data: y

W = 0.94089, p-value = 0.563

> cat("There are normally distributed, best suit is pearson!!")

There are normally distributed, best suit is pearson!!



1. Link below provides the dataset related to breast cancer (please note that the variable names are not listed.)

<https://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/wdbc.data>

1. Import the dataset in R and determine its dimension.
2. Select column 3 through 8 and create a correlation matrix.
3. Display the correlation matrix using correlogram.

> #Q2

> Q2 = read.csv("https://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/wdbc.data", header = F)

> dim(Q2)

[1] 569 32

> data <- Q2[, 3:8]

> cor(data)

V3 V4 V5 V6 V7 V8

V3 1.0000000 0.32378189 0.9978553 0.9873572 0.17058119 0.5061236

V4 0.3237819 1.00000000 0.3295331 0.3210857 -0.02338852 0.2367022

V5 0.9978553 0.32953306 1.0000000 0.9865068 0.20727816 0.5569362

V6 0.9873572 0.32108570 0.9865068 1.0000000 0.17702838 0.4985017

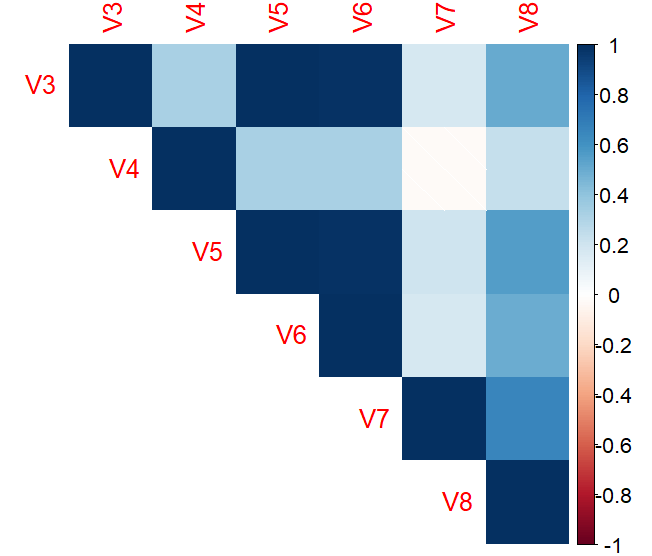
V7 0.1705812 -0.02338852 0.2072782 0.1770284 1.00000000 0.6591232

V8 0.5061236 0.23670222 0.5569362 0.4985017 0.65912322 1.0000000

> install.packages("corrplot")

> library(corrplot)

> corrplot(cor(data), method = "shade", type = "upper")



1. The data frame **Grades** in the **PASWR** package contains the information about the GPA and SAT scores of second semester freshman students.
2. Create a scatterplot of the data to investigate the relationship between gpa and sat scores
3. Obtain the least squares estimates for  and  . State the estimated regression model.
4. Display the regression model along with the scatterplot.

> #Q3

> install.packages("PASWR")

> library(PASWR)

> data(Grades)

> head(Grades)

sat gpa

1 1410 3.67

2 1100 2.43

3 1130 1.95

4 980 2.05

5 1070 2.34

6 1030 2.04

> dim(Grades)

[1] 200 2

> attach(Grades)

> plot(sat,gpa, pch = 17, col = 2, main = "GPA vs SAT")

> m = lm(gpa~sat)

> m

Call:

lm(formula = gpa ~ sat)

Coefficients:

(Intercept) sat

-1.192064 0.003094

> cat("The equation of the fitted model is: gpa = -1.192064 + 0.003094\*sat")

The equation of the fitted model is: gpa = -1.192064 + 0.003094\*sat

> abline(m, lwd = 2, col = 4)

