**LAB EXERCISE - 7**

AIM ::

Exercises to present the data as a frequency table in SPSS.

Scilab Code ::

### ****data = [23, 45, 23, 45, 23, 56, 56, 23, 45, 56, 23, 45];****

Output ::

### ****[frequencies, edges] = hist(data, unique(data));****

### ****disp("Frequency Table using hist():");****

### ****disp(frequencies);****

### ****unique\_values = unique(data);****

### ****disp("Unique values:");****

### ****disp(unique\_values);****

### ****frequency\_table = zeros(1, length(unique\_values));****

### ****for i = 1:length(unique\_values)****

### ****frequency\_table(i) = sum(data == unique\_values(i));****

### ****value****

### ****end****

### ****disp("Frequency Table using unique() and count():");****

### ****disp(frequency\_table);****

### ****data <- c(23, 45, 23, 45, 23, 56, 56, 23, 45, 56, 23, 45)frequency\_table <- table(data)****

R Code ::

Output ::

### ****cat("Frequency Table using table():\n")****

### ****print(frequency\_table)****

### ****cat("\nSummary using summary():\n")****

### ****print(summary(data))****

**LAB EXERCISE - 8**

AIM ::

Exercises to find the outliers in a dataset in SPSS.

Scilab Code ::

### ****data = [100, 102, 104, 120, 130, 130, 150, 200, 250, 300];****

Output ::

### ****boxplot(data);****

### ****mean\_data = mean(data);****

### ****std\_data = stdev(data);****

### ****z\_scores = (data - mean\_data) / std\_data;****

### ****outliers\_z = find(abs(z\_scores) > 3);****

### ****disp("Outliers based on Z-scores:");****

### ****disp(data(outliers\_z));****

### ****Q1 = median(data(1:round(end/2)));****

### ****Q3 = median(data(round(end/2)+1:end));****

### ****IQR = Q3 - Q1;****

### ****lower\_bound = Q1 - 1.5 \* IQR;****

### ****upper\_bound = Q3 + 1.5 \* IQR;****

### ****outliers\_iqr = find(data < lower\_bound | data > upper\_bound);disp("Outliers based on IQR:");****

### ****disp(data(outliers\_iqr));****

### ****data <- c(23, 45, 23, 45, 23, 56, 56, 23, 45, 56, 23, 45)frequency\_table <- table(data)****

R Code ::

Output ::

### ****cat("Frequency Table using table():\n")****

### ****print(frequency\_table)****

### ****cat("\nSummary using summary():\n")****

### ****print(summary(data))data <- c(100, 102, 104, 120, 130, 130, 150, 200, 250, 300)****

### ****boxplot(data, main="Boxplot to detect outliers") # Outliers are shown as dots outside****

### ****the whiskers****

### ****z\_scores <- (data - mean(data)) / sd(data)****

### ****outliers\_z <- data[abs(z\_scores) > 3]****

### ****cat("Outliers based on Z-scores:", outliers\_z, "\n")****

### ****Q1 <- quantile(data, 0.25)****

### ****Q3 <- quantile(data, 0.75)****

### ****IQR <- Q3 - Q1****

### ****lower\_bound <- Q1 - 1.5 \* IQR****

### ****upper\_bound <- Q3 + 1.5 \* IQR****

### ****outliers\_iqr <- data[data < lower\_bound | data > upper\_bound]****

### ****cat("Outliers based on IQR:", outliers\_iqr, "\n")****

**LAB EXERCISE - 9**

AIM ::

Exercises to find the most risky project out of two mutually exclusive projects in SPSS

Scilab Code ::

### ****cash\_flow\_A = [100000, 150000, 200000];****

Output ::

### ****cash\_flow\_B = [90000, 120000, 180000];****

### ****prob\_A = [0.3, 0.4, 0.3];****

### ****prob\_B = [0.4, 0.3, 0.3];****

### ****EV\_A = sum(cash\_flow\_A .\* prob\_A);****

### ****EV\_B = sum(cash\_flow\_B .\* prob\_B);****

### ****disp("Expected Value of Project A:");****

### ****disp(EV\_A);****

### ****disp("Expected Value of Project B:");****

### ****disp(EV\_B);****

### ****variance\_A = sum((cash\_flow\_A - EV\_A).^2 .\* prob\_A);****

### ****SD\_A = sqrt(variance\_A);****

### ****disp("Standard Deviation of Project A:");****

### ****disp(SD\_A);****

### ****variance\_B = sum((cash\_flow\_B - EV\_B).^2 .\* prob\_B);****

### ****SD\_B = sqrt(variance\_B);disp("Standard Deviation of Project B:");****

### ****disp(SD\_B);****

### ****if SD\_A > SD\_B then****

### ****disp("Project A is riskier");****

### ****else****

### ****disp("Project B is riskier");****

### ****end****

### ****cash\_flow\_A <- c(100000, 150000, 200000)****

R Code ::

Output ::

### ****cash\_flow\_B <- c(90000, 120000, 180000)****

### ****prob\_A <- c(0.3, 0.4, 0.3)****

### ****prob\_B <- c(0.4, 0.3, 0.3)****

### ****EV\_A <- sum(cash\_flow\_A \* prob\_A)****

### ****EV\_B <- sum(cash\_flow\_B \* prob\_B)****

### ****cat("Expected Value of Project A:", EV\_A, "\n")****

### ****cat("Expected Value of Project B:", EV\_B, "\n")****

### ****variance\_A <- sum((cash\_flow\_A - EV\_A)^2 \* prob\_A)****

### ****SD\_A <- sqrt(variance\_A)****

### ****cat("Standard Deviation of Project A:", SD\_A, "\n")****

### ****variance\_B <- sum((cash\_flow\_B - EV\_B)^2 \* prob\_B)****

### ****SD\_B <- sqrt(variance\_B)****

### ****cat("Standard Deviation of Project B:", SD\_B, "\n")if (SD\_A > SD\_B) {****

### ****cat("Project A is riskier\n")****

### ****} else {****

### ****cat("Project B is riskier\n")****

### ****}****

**LAB EXERCISE - 10**

AIM ::

Exercises to draw a scatter diagram, residual plots, outliers leverage and influential data points in R

Code ::

**library(car)**

data(mtcars)

plot(mtcars$wt, mtcars$mpg, main="Scatter Plot of mpg vs wt", xlab="Weight of Car

(wt)", ylab="Miles per Gallon (mpg)", pch=**19**, col="blue")

lm\_model <- lm(mpg ~ wt, data = mtcars)

residuals <- residuals(lm\_model)

plot(lm\_model$fitted.values, residuals, main="Residual Plot", xlab="Fitted Values",

ylab="Residuals", pch=**19**, col="red")

abline(h = **0**, col="blue") # Add **a** horizontal line at zero

outliers <- outlierTest(lm\_model)

cat("Outliers:\n")

print(outliers)

plot(lm\_model$fitted.values, residuals(lm\_model), main="Residuals and Outliers",

xlab="Fitted Values", ylab="Residuals", pch=**19**, col="blue")

abline(h = **0**, col="red")

leverage <- hatvalues(lm\_model)

cooks\_d <- cooks.distance(lm\_model)

plot(leverage, residuals(lm\_model), main="Leverage vs Residuals", xlab="Leverage

(Hat Values)", ylab="Residuals", pch=**19**, col="green")

plot(cooks\_d, main="Cook's Distance", ylab="Cook's Distance", pch=**19**, col="purple")

abline(h = **4**/nrow(mtcars), col="red") # Threshold **for** influence (**4**/n)

influencePlot(lm\_model, main="Influence Plot for mpg vs wt")

high\_leverage <- which(leverage > **2** \* mean(leverage))

plot(mtcars$wt, mtcars$mpg, main="Scatter Plot with High Leverage Points",

xlab="Weight (wt)", ylab="Miles per Gallon (mpg)", pch=**19**, col="blue")

points(mtcars$wt[high\_leverage], mtcars$mpg[high\_leverage], col="red", pch=**19**)

### 

Output ::

**LAB EXERCISE - 11**

AIM ::

Exercises to calculate correlation using R

Code ::

Output ::

**clc; clear;**

// Correlation manually

// Data

X = [**1**, **2**, **3**, **4**, **5**];

Y = [**2**, **4**, **5**, **4**, **6**];

// means of X and Y

mean\_X = mean(X);

mean\_Y = mean(Y);

//Calculate the numerator and denominator

// Covariance part

numerator = sum((X - mean\_X).\* (Y - mean\_Y));

// Product of standard deviations

denominator = sqrt(sum((X - mean\_X).^**2**) sum((Y – mean\_Y).^**2**));

// correlation coefficient

r\_manual = numerator / denominator;

// Display the result

mprintf("Manually Calculated Correlation Coefficient (r) = %.2f\n", r\_manual);

//Interpretation

**if** r\_manual > **0** then

mprintf("=> Positive Correlation\n");

**elseif** r\_manual <**0** then

mprintf("=> Negative Correlation\n");

**else**

mprintf("=> No Correlation\n");

**end**

### 

**LAB EXERCISE - 12**

AIM ::

Exercises to implement Time Series using R

Code ::

Output ::

**data("AirPassengers")**

plot(AirPassengers, main="Airline Passenger Numbers (1949-1960)",

xlab="Year", ylab="Number of Passengers", col="blue", type="o")

### 

**LAB EXERCISE - 13**

AIM ::

Exercises to implement Linear Regression using R

Code ::

Output ::

**data(mtcars)**

lm\_model <- lm(mpg ~ wt, data = mtcars)

summary(lm\_model)

plot(mtcars$wt, mtcars$mpg, main = "Simple Linear Regression: MPG vs Weight",

xlab = "Weight (wt)", ylab = "Miles per Gallon (mpg)", pch = **19**, col = "blue")

abline(lm\_model, col = "red")

### 

**LAB EXERCISE - 14**

AIM ::

Exercises to implement concepts of probability and distributions R

Code ::

Output ::

### **normal\_samples <- rnorm(**1000**, mean = **0**, sd = **1**)**

pdf\_1 <- dnorm(**1**, mean = **0**, sd = **1**)

cat("PDF at x = 1:", pdf\_1, "\n")

hist(normal\_samples, main="Histogram of Normal Distribution", xlab="Values",

col="lightgreen", breaks=**20**)

cdf\_1 <- pnorm(**1**, mean = **0**, sd = **1**)

cat("CDF at x = 1:", cdf\_1, "\n")

quantiles <- qnorm(c(**0.25**, **0.50**, **0.75**), mean = **0**, sd = **1**)

cat("Quantiles at 0.25, 0.50, 0.75:", quantiles, "\n")

### 