

# R Programming – Statistical Exercises (Day 2)

## 1. Covariance and Correlation

### Aim:

To study the relationship between variables using covariance and correlation.

### Algorithm:

Step 1: Input preference data.

Step 2: Use cov() and cor() functions.

Step 3: Display covariance and correlation matrices.

### R Program:

```
B <- c(22,28,10)
C <- c(20,40,40)
cov(B,C)
cov(cbind(B,C))
cor(B,C)
cor(cbind(B,C))
```

### Output:

The screenshot shows the RStudio interface with the following details:

- Code Editor:** The file is named "exp12.R". The code is identical to the one provided above.
- Environment Tab:**
  - Global Environment: Shows two vectors, B and C, both of length 3.
  - Values table:
    - B: num [1:3] 22 28 10
    - C: num [1:3] 20 40 40
- Plots Tab:** No plots are present.
- Packages Tab:** No packages are listed.
- Help Tab:** No help pages are listed.
- Viewer Tab:** No output is shown.
- Presentation Tab:** Buttons for Present, Print, Edit, and View are visible.

## 2. Data Partitioning and Smoothing

### Aim:

To perform equal-frequency partitioning and apply data smoothing techniques.

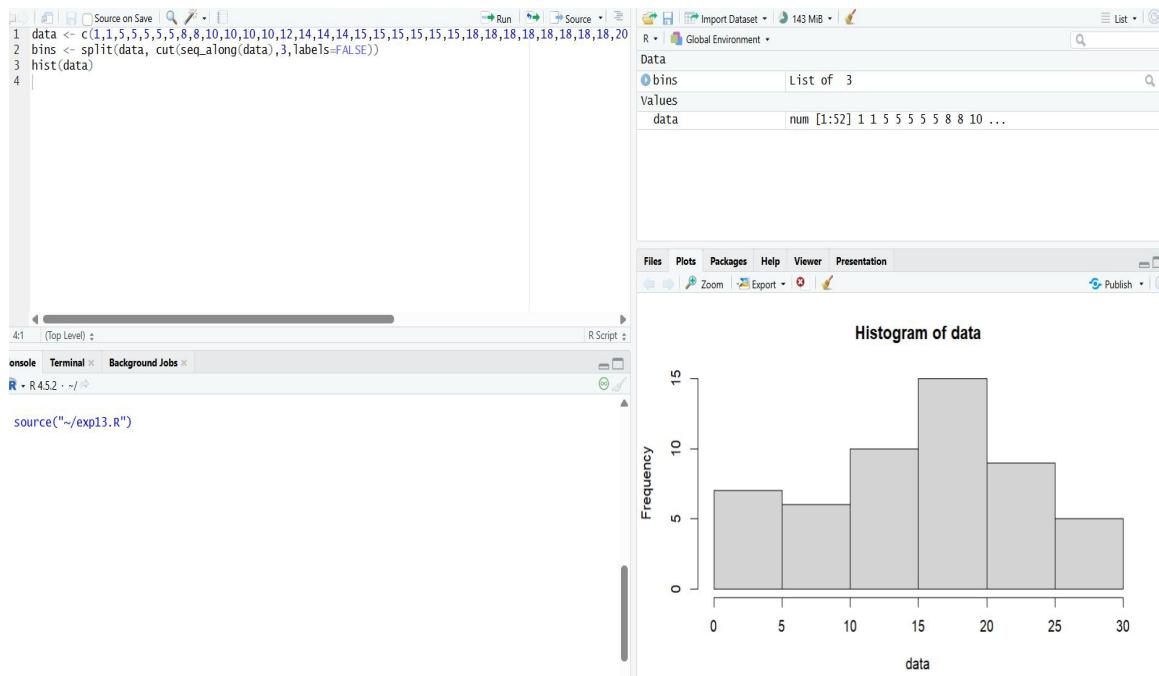
### Algorithm:

- Step 1: Sort dataset.
- Step 2: Divide into equal-frequency bins.
- Step 3: Apply bin means and boundaries.
- Step 4: Plot histogram.

### R Program:

```
data <-  
c(1,1,5,5,5,5,5,8,8,10,10,10,10,12,14,14,14,15,15,15,15,15,15,15,18,18,18,18,18,18,18,18,20,20,  
20,20,20,20,21,21,21,21,25,25,25,25,25,25,28,28,30,30)  
bins <- split(data, cut(seq_along(data),3,labels=FALSE))  
hist(data)
```

### Output:



### 3. Comparison of Two Classes Performance

#### Aim:

To compare statistical measures of two classes using descriptive statistics.

#### Algorithm:

Step 1: Input marks.

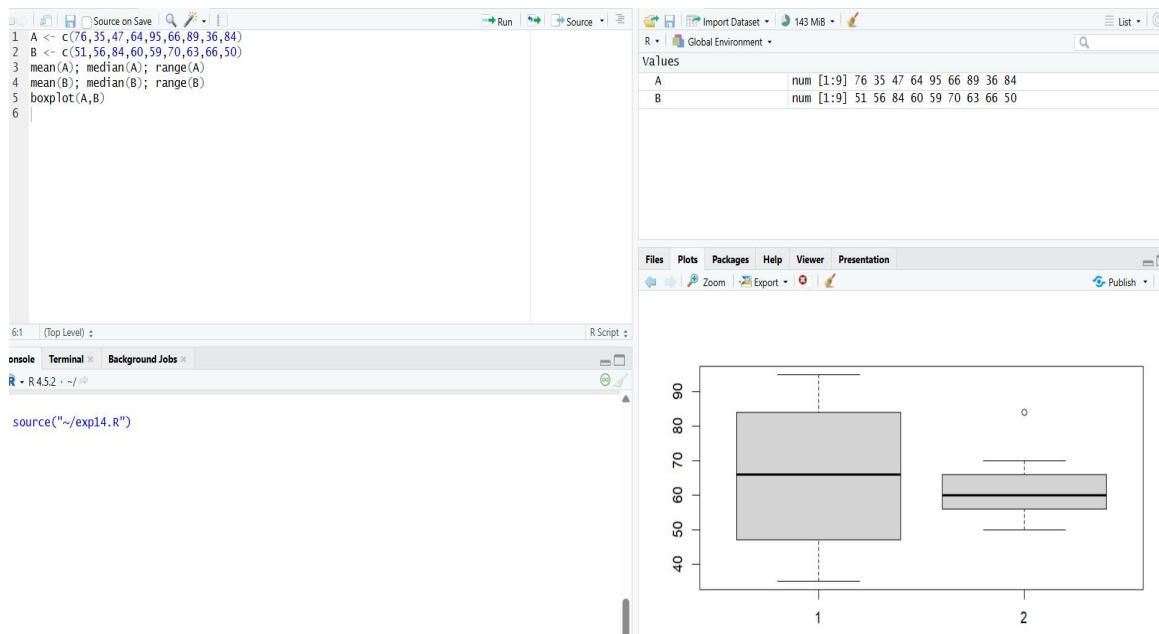
Step 2: Compute mean, median, range.

Step 3: Plot boxplot and infer.

#### R Program:

```
A <- c(76,35,47,64,95,66,89,36,84)
B <- c(51,56,84,60,59,70,63,66,50)
mean(A); median(A); range(A)
mean(B); median(B); range(B)
boxplot(A,B)
```

#### Output:



## 4. Data Normalization Techniques

### Aim:

To normalize data using min-max and z-score normalization techniques.

### Algorithm:

- Step 1: Identify min and max.
- Step 2: Apply min-max formula.
- Step 3: Apply z-score formula.

### R Program:

```
data <- c(200,300,400,600,1000)
(data-min(data))/(max(data)-min(data))
scale(data)
```

### Output:

The screenshot shows the RStudio interface. The top-left pane displays the R script code. The top-right pane shows the Global Environment with variables mm, x, and z. The bottom-left pane is the Console, and the bottom-right pane is the Viewer.

Script Editor (Top Left):

```
1 x <- c(200,300,400,600,1000)
2
3 mm <- (x - min(x)) / (max(x) - min(x))
4 z <- (x - mean(x)) / sd(x)
5
6 mm
7 z
8
9
```

Global Environment (Top Right):

Values	x	z
mm	[1:5] 0 0.125 0.25 0.5 1	
x	[1:5] 200 300 400 600 1000	
z	[1:5] -0.949 -0.632 -0.316 0.316 1.581	

Console (Bottom Left):

```
8:1 (Top Level) R Script
Console Terminal Background Jobs
R 4.5.2 · ~/R
> source("~/exp15.R")
> source("~/exp15.R")
```

Viewer (Bottom Right):

## 5. Histogram for AirPassengers Dataset

### Aim:

To visualize the distribution of AirPassengers data using histogram.

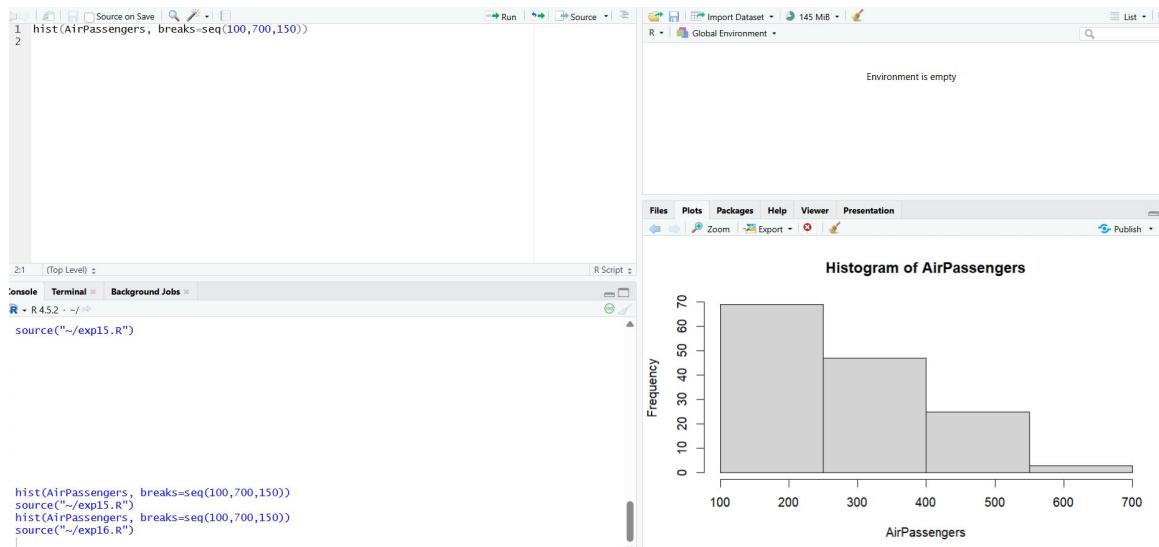
### Algorithm:

- Step 1: Load dataset.
- Step 2: Define breaks.
- Step 3: Plot histogram.

### R Program:

```
hist(AirPassengers, breaks=seq(100,700,150))
```

### Output:



## **6. Multiple Line Plot using mtcars Dataset**

### **Aim:**

To plot multiple attributes using a single plot function in R.

### **Algorithm:**

Step 1: Load mtcars dataset.

Step 2: Plot mpg and qsec using plot().

### **R Program:**

```
plot(mtcars$mpg, type='l')
lines(mtcars$qsec, col=2)
```

### **Output:**

## **7. Linear Regression on Water Dataset**

### **Aim:**

To identify linear relationship and predict values using regression.

### **Algorithm:**

Step 1: Load water dataset.

Step 2: Plot hardness vs mortality.

Step 3: Fit linear model.

Step 4: Predict value.

### **R Program:**

```
data(water)
plot(water$hardness, water$mortality)
model <- lm(mortality~hardness, data=water)
predict(model, data.frame(hardness=88))
```

### **Output:**

## 8. Boxplot for mpg vs cyl (mtcars)

### Aim:

To analyze relationship between mileage and cylinder count using boxplot.

### Algorithm:

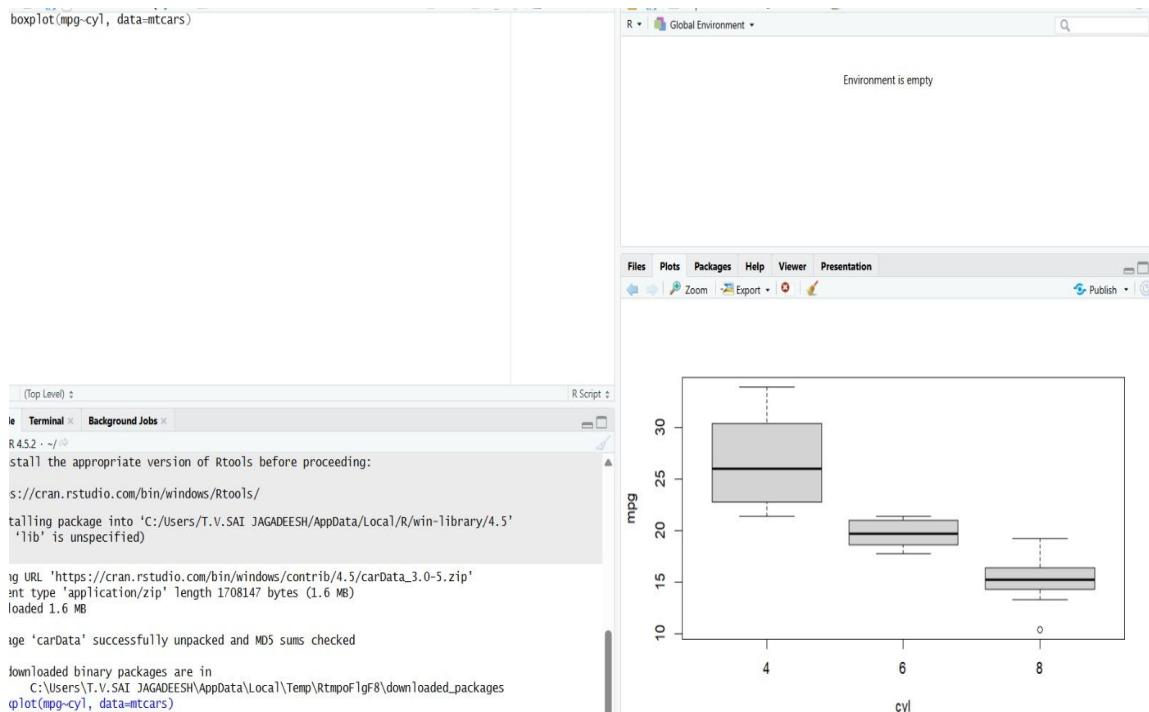
Step 1: Load mtcars.

Step 2: Plot mpg vs cyl using boxplot.

### R Program:

```
boxplot(mpg~cyl, data=mtcars)
```

### Output:



## 9. Outlier Detection using Boxplot

**Aim:**

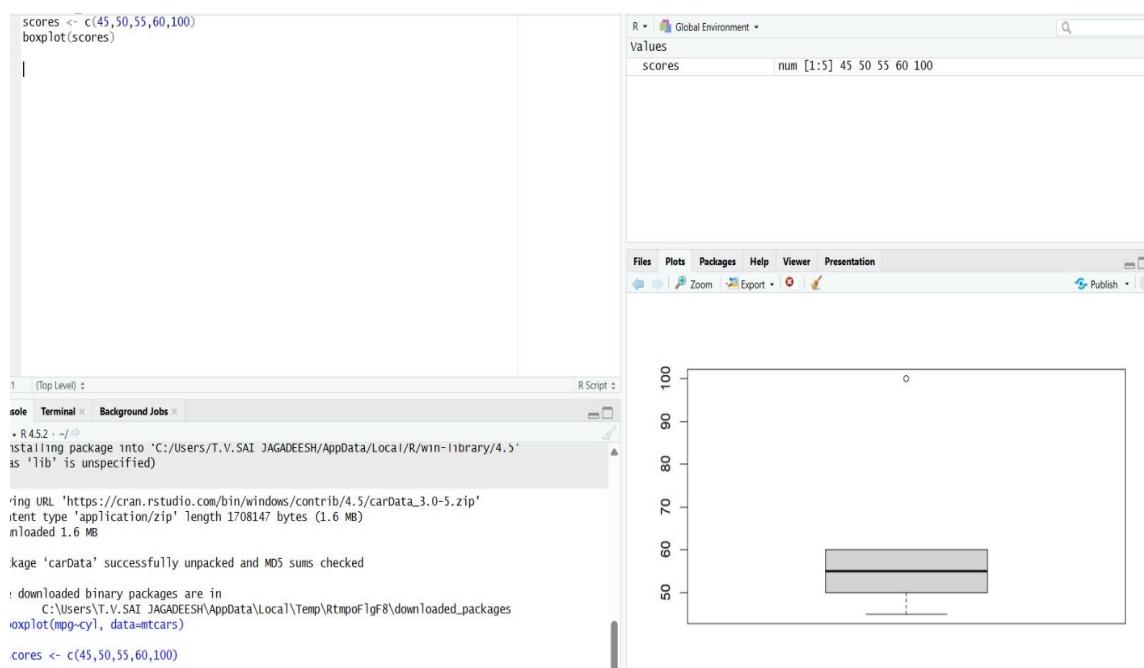
To detect outliers using boxplot visualization.

**Algorithm:**

- Step 1: Collect player scores.
- Step 2: Plot boxplot.
- Step 3: Identify outliers.

**R Program:**

```
scores <- c(45,50,55,60,100)
boxplot(scores)
```

**Output:**

## **10. Blood Pressure vs Age Analysis (Diabetes Dataset)**

### **Aim:**

To analyze age group affected by blood pressure using diabetes data.

### **Algorithm:**

Step 1: Load diabetes dataset.

Step 2: Plot scatterplot and bar chart.

### **R Program:**

```
data <- read.csv('diabetes.csv')
plot(data$Age, data$BloodPressure)
barplot(data$BloodPressure)
```

### **Output:**