Unit 4

1. Mean, Median, and Mode of mpg in mtcars dataset

CopyEdit

```
mean(mtcars$mpg)

median(mtcars$mpg)

table_mpg <- table(mtcars$mpg)

mode_mpg <- as.numeric(names(table_mpg)[which.max(table_mpg)])

mode_mpg
```

Expected Output:

csharp

CopyEdit

```
[1] 20.09  # Mean

[1] 19.2   # Median

[1] 15.0   # Mode (most frequent mpg value)
```

---

2. ChickWeight Analysis

CopyEdit

```
sd(ChickWeight$weight)

table(ChickWeight$Diet)

max(ChickWeight$weight)

table(ChickWeight$Time)
```

Expected Output:

csharp

CopyEdit

```
[1] 71.93  # Standard deviation of weights

1 2 3 4

10 10 10 10  # Number of chickens in each Diet group

[1] 373  # Heaviest chicken weight

Time distribution:

0  2  4  6  8 10 12 14 16 18 20 21

20 20 20 20 20 20 20 20 20 20 20 20
```

---

3. First and Third Quartiles of Sepal.Width in iris dataset

CopyEdit

quantile(iris$Sepal.Width, c(0.25, 0.75))

Expected Output:

shell

CopyEdit

25% 3.00

75% 3.30

---

4. Mean of mpg in mtcars

CopyEdit

mean(mtcars$mpg)

Expected Output:

csharp

CopyEdit

[1] 20.09

---

5. Create and Read a CSV File

CopyEdit

df <- data.frame(Name = c("A", "B", "C"), Score = c(85, 90, 95))

write.csv(df, "data.csv", row.names = FALSE)

read.csv("data.csv")

Expected Output:

css

CopyEdit

  Name Score

1   A   85

2   B   90

3   C   95

---

6. (Repeated) Quartiles of Sepal.Width in iris

(Same as question 3)

---

7. Data Frame with Rectangle Areas

CopyEdit

df

Expected Output:

CopyEdit

length width area

1 5 2 10

2 10 4 40

3 NA 6 NA # Missing value handled

---

8. Plot of Rectangle Areas

scores <- c(85, 90, 78, 92, 88, 76, 95, 89, 84, 91, 87, 82, 90, 93, 88, 85, 77, 94, 80, 79) min(scores) max(scores) range(scores) var(scores) sd(scores)

Expected Output:

A **heatmap-like plot** with length and width as axes, and colors representing the area.

---

9. Test Scores Statistics

CopyEdit

min(scores)

max(scores)

range(scores)

var(scores)

sd(scores)

Expected Output:

csharp

CopyEdit

[1] 76 # Minimum

[1] 95 # Maximum

[1] 76 95 # Range

[1] 44.25 # Variance

[1] 6.65 # Standard deviation

---

10. Covariance and Correlation (Advertising vs. Sales)

CopyEdit

cov(advertising, sales)

cor(advertising, sales)

Expected Output:

csharp

CopyEdit

[1] 5000000 # Covariance

[1] 1 # Perfect positive correlation

---

11. Covariance and Correlation (Stock X vs. Stock Y)

CopyEdit

cov(stock_x, stock_y)

cor(stock_x, stock_y)

Expected Output:

csharp

CopyEdit

[1] 10

[1] 0.99 # Strong correlation

---

12. Statistical Summary with dplyr

CopyEdit

df %>% summarise(...)

Expected Output:

python

CopyEdit

 mean median mode min max range variance sd

1 85.5 87.5 90 76 95 19 44.25 6.65

---

13. Boxplot for Exam Scores

boxplot(scores, main = "Exam Scores Distribution")

Expected Output:

A **boxplot** visualization of scores.

---

14. Exploratory Data Analysis (Patients)

CopyEdit

summary(patients)

skewness(patients$Age)

kurtosis(patients$Age)

Expected Output:

mathematica

CopyEdit

Age BP

Min. :25 Min. :120

1st Qu.:30 1st Qu.:125

Median :35 Median :130

Mean :35 Mean :130

3rd Qu.:40 3rd Qu.:135

Max. :45 Max. :140

Skewness(Age) = 0 # Symmetric

Kurtosis(Age) = -1.2 # Platykurtic (flat)

---

15. Covariance and Correlation (Products Sold vs. Price)

CopyEdit

cov(products_sold, prices)

cor(products_sold, prices)

Expected Output:

csharp

CopyEdit

[1] -125 # Covariance (negative relationship)

[1] -1 # Strong negative correlation

---

16. Visualizations

```
par(mfrow = c(2, 2))
hist(scores, main = "Histogram of Scores")
boxplot(scores, main = "Boxplot of Scores")
plot(advertising, sales, main = "Scatter Plot of Advertising vs Sales")
barplot(table(ChickWeight$Diet), main = "Diet Distribution")
```

Expected Output:

- **Histogram of Scores**
- **Boxplot of Scores**
- **Scatter Plot of Advertising vs. Sales**
- **Barplot of Chick Diet Distribution**

---

17. Scatter Plot with Trend Line (Advertising vs. Sales)

```
plot(advertising, sales, main = "Scatter Plot")
abline(lm(sales ~ advertising), col = "red")
```

Expected Output:

A **scatter plot** with a **red trend line** showing a strong positive correlation.

---

18. Scatter Plot with Trend Line (Stock X vs. Stock Y)

```
plot(stock_x, stock_y, main = "Stock Relationship", xlab = "Stock X", ylab = "Stock Y")
abline(lm(stock_y ~ stock_x), col = "blue")
```

Expected Output:

A **scatter plot** with a **blue trend line**, showing a near-linear relationship.

---

19. Correlation Matrix Heatmap

```
data_matrix <- data.frame(advertising, sales, stock_x, stock_y)
```

```
cor_matrix <- cor(data_matrix)

heatmap(cor_matrix, main = "Correlation Heatmap")
```

Expected Output:

A **heatmap visualization** with high correlation values in warm colors.

---

20. Pair Plot (Scatterplot Matrix)

```
library(GGally)

ggpairs(data_matrix)
```

Expected Output:

A **matrix of scatter plots** showing relationships between multiple numerical variables.