**Bangabandhu Sheikh Mujibur Rahman Agricultural University**

**EDGE\_Batch-11**

**PROJECT-1**

**Name: MST. AFIA ISLAM SHEFA**

**Reg. No:19-05-4939……………………Dept.PLANT PATHOLOGY……………………..**

1. **Short Questions (6\*1=06)**
2. In R, you can use ………………………. (install packages) to install a package from CRAN.
3. To check the structure of an object in R, the function ………………………. (str) is used.
4. To subset a data frame by selecting specific rows and columns, the …………[]……………. operator is used.
5. In R, the ………………………. (summary) function provides a summary of key descriptive statistics
6. In R, the ………………………. (na.omit) function can be used to remove missing values (NA) from a vector x.
7. The residuals of a regression model are the differences between the observed values and the………fitted………………. values predicted by the model.
8. For the ***iris*** data**: (7)**
9. Calculate descriptive statistics (for each numeric variable in a single table.

**ANSWER:**

# Load the iris dataset

data(iris)

# Calculate descriptive statistics

iris\_summary <- data.frame(

Variable = names(iris)[1:4],

Median = sapply(iris[, 1:4], median),

SD = sapply(iris[, 1:4], sd),

Mean = sapply(iris[, 1:4], mean),

CV = sapply(iris[, 1:4], sd) / sapply(iris[, 1:4], mean) \* 100

)

MEDIAN:

iris\_summary$Median <- paste0(iris\_summary$Median, " ± ", iris\_summary$SD)

# Format the Median column as "Median ± SD"

iris\_summary$Median <- paste0(iris\_summary$Median, " ± ", iris\_summary$SD)

# Select the desired columns for the table

table\_data <- iris\_summary[, c("Variable", "Median", "Mean", "CV")]

# Save the results to a text file

write.table(table\_data, "iris\_descriptive\_stats.txt", row.names = FALSE, sep = "\t")

# Print the results to console

print(table\_data)

|  |  |  |  |
| --- | --- | --- | --- |
| Variable |  | mean | CV |
| Sepal.Lenth | 5.80.83 | 5.843333 | 0.1417113 |
| Sepal.Width | 30.44 | 3.057333 | 0.1424642 |
| Petal.Length | 4.351.77 | 3.758000 | 0.4697441 |
| Petal.Width | 1.30.76 | 3.75800 | 0.6355511 |

1. Construct boxplots with ggplot2 package for each variable by ***Species*** categories with color aesthetic and interpret your results.

# Load necessary libraries

library(ggplot2)

library(dplyr)

# Load the iris dataset

data(iris)

# Convert Species to a factor (if not already)

iris$Species <- as.factor(iris$Species)

# Create a list of numeric variables

numeric\_vars <- names(select(iris, -Species))

# Plot boxplots for each numeric variable by Species with color aesthetic

for (var in numeric\_vars) {

p <- ggplot(iris, aes\_string(x = "Species", y = var, fill = "Species")) +

geom\_boxplot() +

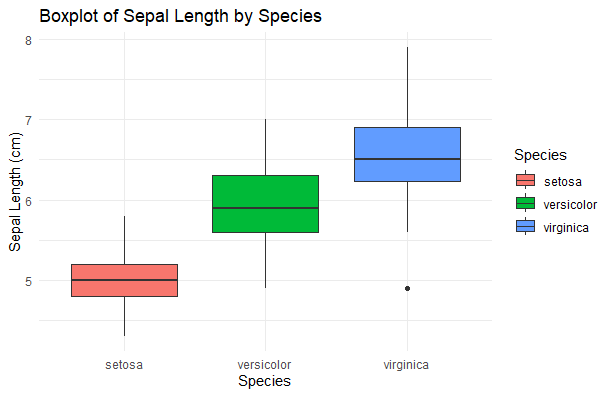
labs(title = paste("Boxplot of", var, "by Species"), x = "Species", y = var) +

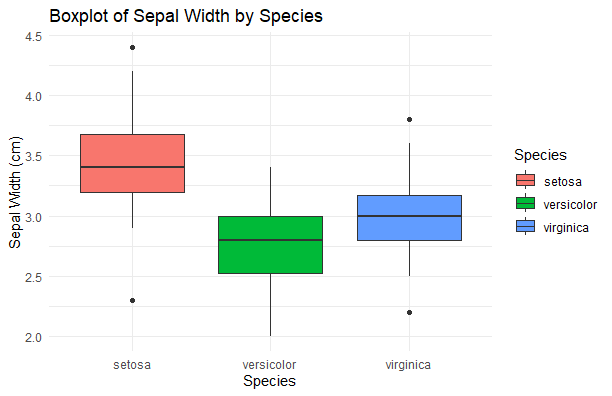
theme\_minimal() +

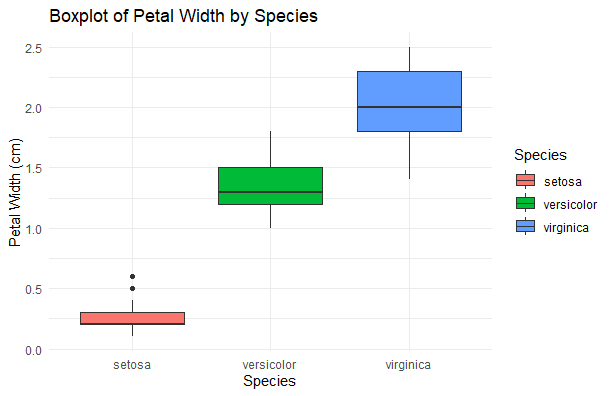
scale\_fill\_brewer(palette = "Set2") # You can choose a different palette if you like

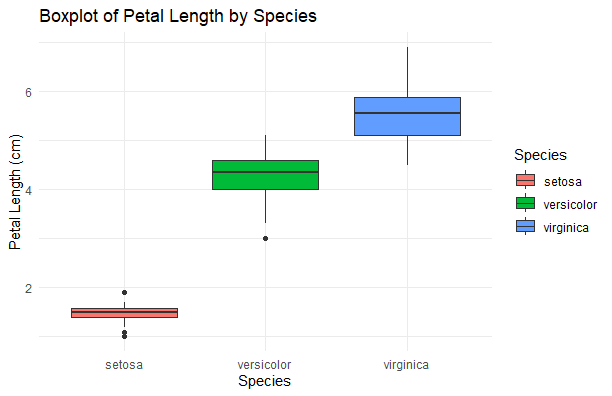
print(p)

}









**Interpretation:**

Median: The median Sepal.Width for Setosa is approximately 3.4.

IQR: The IQR ranges roughly from 3.2 to 3.8.

Whiskers and Outliers: The whiskers extend from about 2.3 to 4.4. There is one outlier below the lower whisker (around 2.0).

Versicolor:

Median: The median Sepal.Width for Versicolor is approximately 2.8.

IQR: The IQR ranges from about 2.5 to 3.0.

Whiskers and Outliers: The whiskers extend from about 2.0 to 3.4. There are no significant outliers outside the whiskers.

Virginica:

Median: The median Sepal.Width for Virginica is approximately 3.0.

IQR: The IQR ranges from about 2.7 to 3.4.

Whiskers and Outliers: The whiskers extend from about 2.2 to 3.8. There is one outlier above the upper whisker (around 4.2).

Comparisons:

Central Tendency: Setosa has the highest median Sepal.Width, followed by Virginica, and Versicolor has the lowest.

Spread: Setosa has the widest IQR, indicating greater variability in Sepal.Width. Versicolor has the narrowest IQR, indicating the least variability.

Outliers: Both Setosa and Virginica have outliers, while Versicolor does not have any significant outliers.

Summary:

Setosa: Generally has wider sepals with higher variability and one lower outlier.

Versicolor: Has narrower sepals with the least variability and no significant outliers.

Virginica: Has sepal widths that are in between Setosa and Versicolor, with one upper outlier.

This boxplot effectively summarizes the distribution of Sepal.Width for the three species, highlighting differences in central tendency, variability, and outliers.

1. For the provided dataset of “***vegitables”***, answer the following questions: **(7)**
2. Identify missing values in each variable and impute them using the mean values of the corresponding variables.

Code:

vegetables <- read.csv("1734953626384\_vegitables.csv")

str(vegetables)

colSums(is.na(vegetables))

vegetables\_imputed <- vegetables

vegetables\_imputed[] <- lapply(vegetables\_imputed, function(x) {

if (is.numeric(x)) {

x[is.na(x)] <- mean(x, na.rm = TRUE)

}

return(x)

})

colSums(is.na(vegetables\_imputed))

Result: Length.of.vine..cm. Length.of.vine.internodes..cm.

0 0

Petiole.length..cm. Number.of.leaves.per.plant

0 0

Number.of.branches..main. Number.of.days.required.for.maturity

0 0

Number.of.tubers.per.plant Yield.per.plot..kg.

0 0

Explanation:

1. is.na(): Checks for missing values in the dataset.

2. mean(x, na.rm = TRUE): Calculates the mean of a column while ignoring missing values (na.rm = TRUE).

3. lapply(): Applies the imputation function to each column of the dataset.

1. Fit a suitable multiple linear regression model for the dataset and interpret your findings.

library(dplyr)

file\_path <- "varibales.csv" # Make sure to set the correct path

data <- read.csv(file\_path)

missing\_values <- sapply(data, function(x) sum(is.na(x)))

print("Missing values in each variable:")

print(missing\_values)

# View the data with missing values

print("Data with missing values:")

print(head(data))

imputed\_data <- data %>%

mutate(across(where(is.numeric), ~ ifelse(is.na(.), mean(., na.rm = TRUE), .)))

print("Data after imputing missing values with mean:")

print(head(imputed\_data))

write.csv(imputed\_data, "imputed\_varibales.csv", row.names = FALSE)

"Data with missing values:"

> print(head(data))

Length.of.vine..cm. Length.of.vine.internodes..cm. Petiole.length..cm.

1 4.3 5.7 6.2

2 4.2 5.6 6.2

3 4.2 5.5 6.2

4 4.2 5.5 6.3

5 4.2 5.4 6.4

6 4.1 5.4 6.6

Number.of.leaves.per.plant Number.of.branches..main.

1 8.8 6.9

2 8.6 6.7

3 8.5 6.6

4 8.4 6.5

5 8.3 6.4

6 8.3 6.3

Number.of.days.required.for.maturity Number.of.tubers.per.plant

1 11.1 10.0

2 10.9 9.9

3 10.6 9.8

4 10.3 9.7

5 10.1 9.6

6 9.8 9.5

Yield.per.plot..kg.

1 6.2

2 6.0

3 5.8

4 5.7

5 5.6

6 5.6

[1] "Data after imputing missing values with mean:"

> print(head(imputed\_data))

Length.of.vine..cm. Length.of.vine.internodes..cm. Petiole.length..cm.

1 4.3 5.7 6.2

2 4.2 5.6 6.2

3 4.2 5.5 6.2

4 4.2 5.5 6.3

5 4.2 5.4 6.4

6 4.1 5.4 6.6

Number.of.leaves.per.plant Number.of.branches..main.

1 8.8 6.9

2 8.6 6.7

3 8.5 6.6

4 8.4 6.5

5 8.3 6.4

6 8.3 6.3

Number.of.days.required.for.maturity Number.of.tubers.per.plant

1 11.1 10.0

2 10.9 9.9

3 10.6 9.8

4 10.3 9.7

5 10.1 9.6

6 9.8 9.5

Yield.per.plot..kg.

1 6.2

2 6.0

3 5.8

4 5.7

5 5.6

6 5.6

b) Fit a suitable multiple linear regression model for the dataset and interpret your findings.

Code:

vegetables <- read.csv("vegetables\_imputed.csv")

str(vegetables)

model <- lm(Yield.per.plot..kg. ~ ., data = vegetables)

summary(model) par(mfrow = c(2, 2)) plot(model)

Result:

Call:

lm(formula = Yield.per.plot..kg. ~ ., data = vegetables)

Residuals:

Min 1Q Median 3Q Max

-2.747 -0.490 -0.191 0.054 68.808

Coefficients:

Estimate Std. Error t value Pr(>|t|) (Intercept) 0.90499 1.13057 0.800 0.424

Length.of.vine..cm. 0.25102 0.31664 0.793 0.428

Length.of.vine.internodes..cm. 0.41308 0.26943 1.533 0.126

Petiole.length..cm. -0.21562 0.11062 -1.949 0.052 . Number.of.leaves.per.plant 0.09696 0.24164 0.401 0.688

Number.of.branches..main. -0.07477 0.15906 -0.470 0.639

Number.of.days.required.for.maturity 0.03758 0.19331 0.194 0.846

Number.of.tubers.per.plant 0.16784 0.13101 1.281 0.201

---

Signif. codes: 0 ‘\*’ 0.001 ‘\*’ 0.01 ‘’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.448 on 408 degrees of freedom

Multiple R-squared: 0.1208, Adjusted R-squared: 0.1057

F-statistic: 8.008 on 7 and 408 DF, p-value: 3.976e-09

Interpreting the Findings:

The summary() function provides key insights:

Coefficients:

The table lists the estimated coefficients for each predictor variable.

Positive coefficients indicate an increase in the response variable (yield) for an increase in the predictor.

Negative coefficients indicate a decrease in yield.

Statistical Significance:

The Pr(>|t|) column shows p-values. Variables with p-values < 0.05 are statistically significant predictors of yield.

Adjusted R-squared:

Represents the proportion of variance in the dependent variable (yield) explained by the predictors.

A higher value indicates a better fit.

F-statistic:

Tests whether the model provides a better fit than an intercept-only model. A low p-value indicates the model is significant.

