STUDENT PERFORMANCE DATA ANALYSIS

Submitted To
Dr. Ashraf Uddin
Assistant Professor, CS, AIUB

AI Usage Declaration

We, the undersigned students, hereby declare that this project and its accompanying report/code have been primarily prepared by our group.

We acknowledge that the use of Artificial Intelligence (AI) tools such as ChatGPT, GitHub Copilot, Grammarly, or similar systems was permitted only to assist in learning, idea generation, code debugging, or language improvement.

We further declare that:

1. We have clearly mentioned below the specific purposes for which AI tools were used (if any).
2. The core design, implementation, analysis, and conclusions are our own original work.
3. We collectively take full academic responsibility for the content of this submission.

AI Usage Details:

☐ No AI tools were used.

☐ AI tools were used for the following purposes (please specify clearly):

	Name	Student ID	Signature with Date
	Md Sifat Hosen	22-47001-1	

Table of Contents

List of Table and Figures

Student Performance Data Analysis

Introduction

This project applies to a complete data science workflow to analyze a real-world student performance dataset. The primary objective is to understand how academic, demographic, and behavioral factors influence student outcomes. The project includes data loading, data type inspection, exploratory data analysis, missing value handling, outlier treatment, encoding of categorical variables, feature transformation, and feature selection.

By following a structured data preprocessing pipeline, the dataset becomes cleaner, more consistent, and better suited for future machine learning applications. This workflow helps uncover trends, patterns, and relationships within the data while ensuring that all variables are processed properly for deeper analysis. The project demonstrates practical knowledge of data manipulation techniques commonly required in real-world analytical tasks.

1. Data Understanding

This stage focuses on loading the dataset, examining its structure, and identifying the types of features present. These steps help determine the preprocessing and analytical techniques that follow.

R Code for Data Understanding

```
# Load libraries

library(dplyr)


# Load dataset

student_data <- read.csv("student_data.csv", header = TRUE)
```

```
# First few rows
head(student_data)


# Dataset shape
cat("Rows: ", nrow(student_data), "\n")
cat("Columns: ", ncol(student_data), "\n")


# Data structure
str(student_data)


# Descriptive statistics
summary(student_data)


# Mode function
get_mode <- function(v) {
  uniq <- unique(v)
  uniq[which.max(tabulate(match(v, uniq)))]
}


sapply(student_data, get_mode)


# Identify categorical and numerical features
categorical_features <- names(student_data)[sapply(student_data, is.factor) | sapply(student_data, is.character)]
```

```
numeric_features <- names(student_data)[sapply(student_data, is.numeric)]
```

Student id	**Weekly self-study hours**	**attendance percentage**	**Class participation**	**Total score**	**grade**
Min: 1	Min : 0.00	Min. : 50.00	Min. : 0.000	Min. : 9.40	Length:1000000
1stQu: 250001	1st Qu:10.30	1st Qu.: 78.30	1st Qu.: 4.700	1st Qu.: 73.90	Class: character
Median: 500001	Median :15.00	Median : 85.00	Median : 6.000	Median: 87.50	Mode: character
Mean: 500001	Mean :15.03	Mean : 84.71	Mean : 5.985	Mean: 84.28	
3rdQu: 750000	3rd Qu:19.70	3rd Qu.: 91.80	3rd Qu.: 7.300	3rd Qu.:100.00	
Max :1000000	Max. :40.00	Max. :100.00	Max. :10.000	Max. :100.00	

Table 1 — Summary Statistics

2. Exploratory Data Analysis (EDA)

The dataset was explored to uncover trends and relationships.
Univariate analysis included:
• Histograms (distribution)
• Boxplots (outliers and spread)
• Bar charts (frequencies of categorical variables)

Bivariate analysis included:
• Correlation matrix
• Scatterplots
• Boxplots between categorical & numerical variables

R Code & Plot for EDA

```
library(ggplot2)

library(corrplot)
```

A. Histograms

```
for (col in numeric_features) {
  print(
    ggplot(student_data, aes_string(col)) +
      geom_histogram(bins = 25) +
      ggtitle(paste("Histogram of", col))
  )
}
```

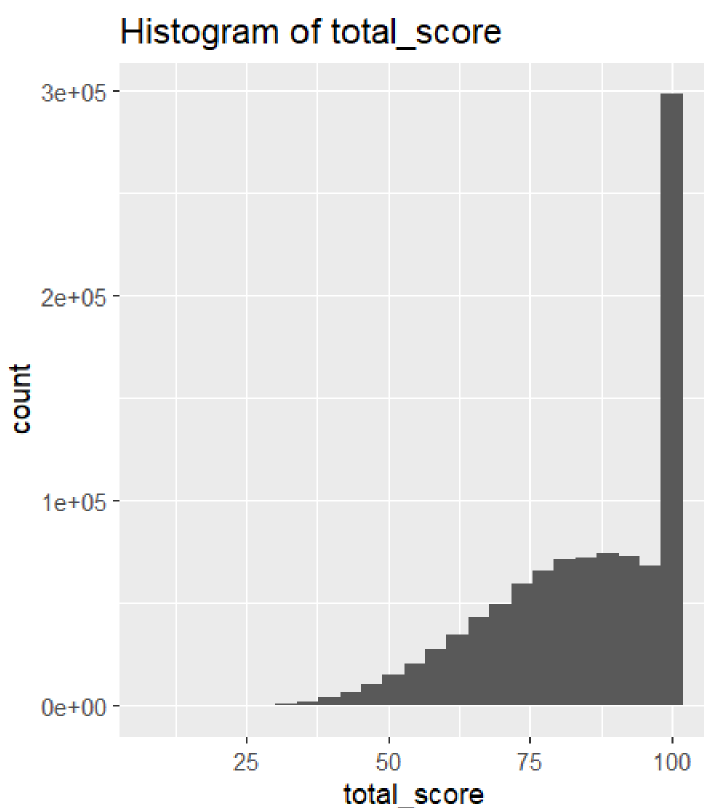

Fig. 1: Distribution of Student Scores

A. Boxplots

```
for (col in numeric_features) {
  print(
    ggplot(student_data, aes_string(y = col)) +
      geom_boxplot() +
      ggtitle(paste("Boxplot of", col))
  )
}
```

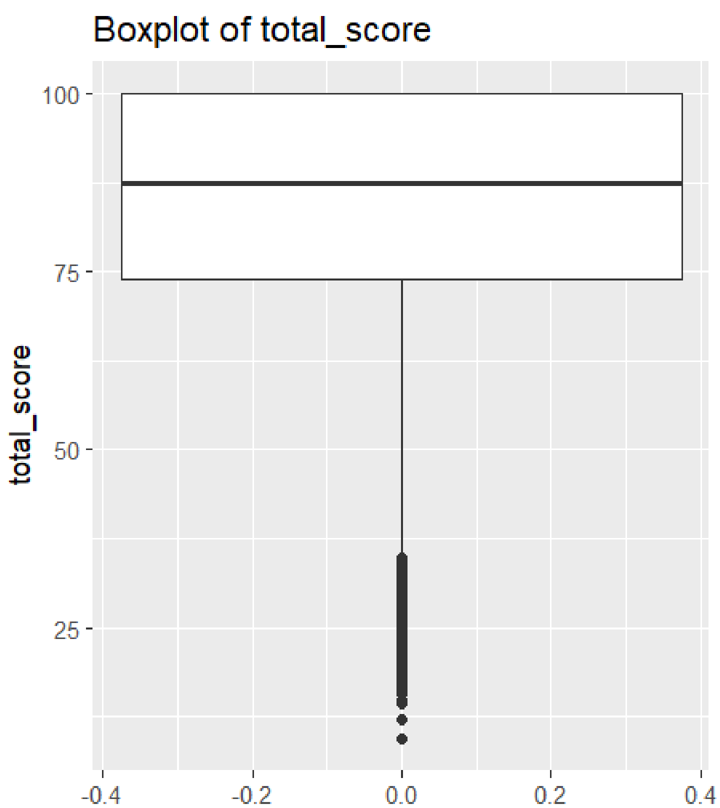

Fig. 2: Boxplot of Student Scores

B. Bar charts for categorical variables

```
for (col in categorical_features) {
 print(
   ggplot(student_data, aes_string(col)) +
     geom_bar() +
     ggtitle(paste("Frequency of", col))
 )
}
```

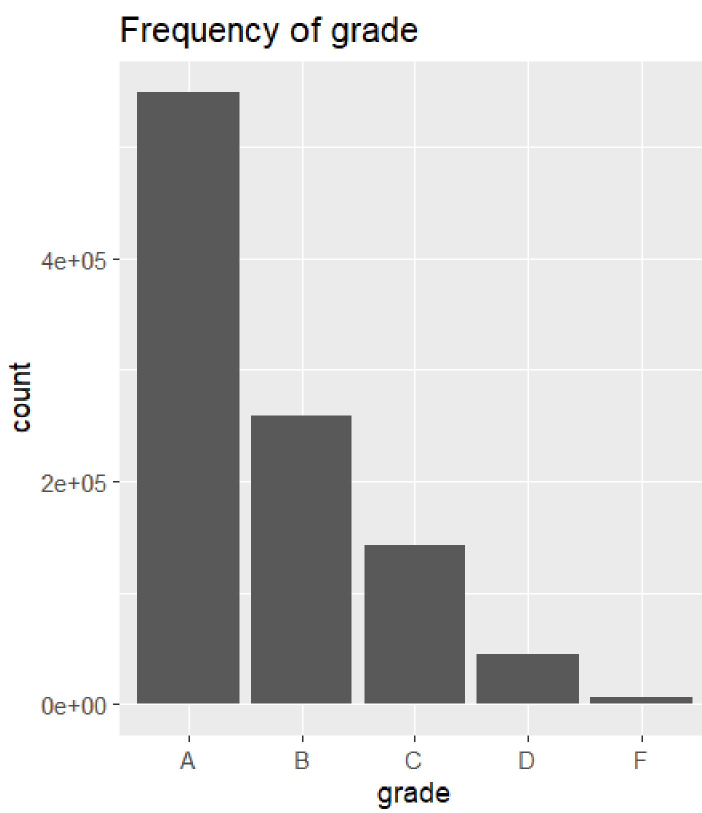

Fig. 3: Frequency Distribution of Gender

C. Correlation matrix

```
numeric_df <- student_data[, numeric_features]
```

```
cor_matrix <- cor(numeric_df, use = "complete.obs")

corrplot(cor_matrix, method = "color", type = "upper")
```

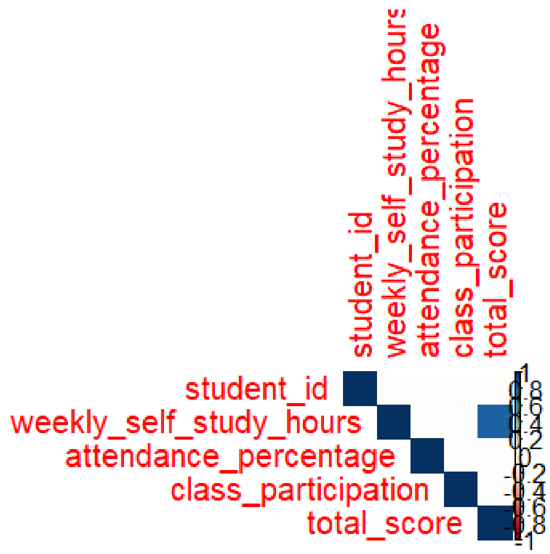

Fig. 4: Correlation Matrix of Numerical Variables

D. Scatterplot matrix

```
pairs(numeric_df)


# Boxplots (categorical vs numeric)

for (cat in categorical_features) {

  for (num in numeric_features) {

    print(

      ggplot(student_data, aes_string(x = cat, y = num)) +

        geom_boxplot() +

        ggtitle(paste("Boxplot of", num, "by", cat))

    )

  }
```

}

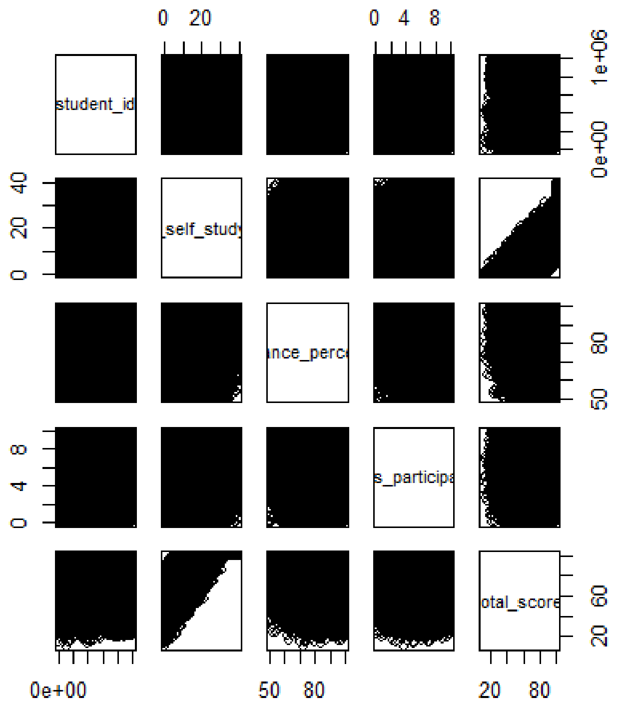

Fig. 5: Scatterplot Matrix

3. Data Preprocessing

This step ensures that the dataset is cleaned and ready for modeling.

3.1 Handling Missing Values

Approach

- Numerical values → mean
- Categorical values → mode

Student id	**Weekly self study hours**	**Attendance percentage**	**Class participation**	**Total score**
0	0	0	0	0
grade				
0				

Table 2 — Missing Value CountR Code

```
# Check missing values
colSums(is.na(student_data))


# Impute values
for (col in names(student_data)) {
  if (is.numeric(student_data[[col]])) {
    student_data[[col]][is.na(student_data[[col]])] <- mean(student_data[[col]], na.rm = TRUE)
  } else {
    student_data[[col]][is.na(student_data[[col]])] <- get_mode(student_data[[col]])
  }
}
```

3.2 Handling Outliers (IQR Method)

Approach

- Identify and cap outliers using the IQR range.

R Code

```
num_cols <- names(student_data)[sapply(student_data, is.numeric)]


for (col in num_cols) {
  Q1 <- quantile(student_data[[col]], 0.25)
  Q3 <- quantile(student_data[[col]], 0.75)
  I <- Q3 - Q1
```

```
  lower <- Q1 - 1.5 * I

  upper <- Q3 + 1.5 * I



  student_data[[col]][student_data[[col]] < lower] <- lower

  student_data[[col]][student_data[[col]] > upper] <- upper

}
```

3.3 Data Conversion (Encoding)

R Code

```
# Label Encoding
student_label <- student_data
for (col in categorical_features) {
  student_label[[col]] <- as.numeric(as.factor(student_label[[col]]))
}
# One-hot Encoding
library(caret)
dummies <- dummyVars("~ .", data = student_data)
student_onehot <- data.frame(predict(dummies, newdata = student_dat
```

3.4 Data Transformation (Scaling & Log Transform)

R Code

```
# Standardization
student_scaled <- student_data
student_scaled[num_cols] <- scale(student_scaled[num_cols])

# Log transformation where possible
student_log <- student_data
for (col in num_cols) {
  if (all(student_log[[col]] > 0)) {
    student_log[[col]] <- log(student_log[[col]])
  }
}
```

3.5 Feature Selection

R Code

```
# Correlation-based selection
cor_mat <- cor(student_onehot)
high_cor <- findCorrelation(cor_mat, cutoff = 0.85)
student_fs <- student_onehot[, -high_cor]

# Variance Thresholding
nzv <- nearZeroVar(student_onehot)
student_fs2 <- student_onehot[, -nzv]
```

Conclusion

This project successfully applied a full data preprocessing pipeline to a student performance dataset. Through exploratory analysis, important insights were discovered about the structure and relationships within the data. Missing values were imputed appropriately, and outliers were managed using a systematic IQR-based approach. Categorical variables were encoded using industry-standard methods, and numerical features were standardized to improve consistency. Feature selection techniques helped reduce redundancy and improve the dataset's usability. Overall, this project prepared the dataset for future predictive modeling and demonstrated a strong understanding of core data preprocessing principles essential for data science work.