

DM PROJECT REPORT



Huzaif Bin Tariq 20F-0272

Muhammad Hayyan 20F-0131

moiz Ghouri 20F-0261

Muhammad Ahmad 20F-0288

Contents

[Introduction 2](#_Toc165207161)

[Data Preprocessing 2](#_Toc165207162)

[Combining Data from Multiple Sheets: 2](#_Toc165207163)

[Creation of Two Separate Datasets: 3](#_Toc165207164)

[Handling Missing Values: 3](#_Toc165207165)

[Summary Statistics 3](#_Toc165207166)

[Before Mid-II Exam: 3](#_Toc165207167)

[Assignment Scores: 3](#_Toc165207168)

[Quiz Scores: 4](#_Toc165207169)

[Mid-I Score: 4](#_Toc165207170)

[Before Final Exam: 5](#_Toc165207171)

[Assignment Scores: 5](#_Toc165207172)

[Model Evaluation 5](#_Toc165207173)

[Predicting Grades before Mid-II Exam 5](#_Toc165207174)

[Nearest Neighbor Classifier: 5](#_Toc165207175)

[Predicting Grades after Mid-II Exam 6](#_Toc165207176)

[Nearest Neighbor Classifier: 6](#_Toc165207177)

[Codes 7](#_Toc165207178)

[Pre-processing codes: 7](#_Toc165207179)

[Before final data extraction: 7](#_Toc165207180)

[Before mid 2 data extraction 8](#_Toc165207181)

[Classifiers: 8](#_Toc165207182)

[Decision Tree Classifier: 8](#_Toc165207183)

[Nearest Neighbor Classifier: 9](#_Toc165207184)

[Visualizations: 10](#_Toc165207185)

[Scatter Plot and Histograms: 10](#_Toc165207186)

[Box Plot: 11](#_Toc165207187)

[Correlation Heatmap: 12](#_Toc165207188)

[Conclusion 12](#_Toc165207189)

# Introduction

In the modern educational landscape, understanding student performance and predicting their academic outcomes is vital for educators and institutions alike. With the rise of data-driven methodologies, utilizing datasets containing student assessment results has become essential for this purpose.

This initiative aims to use such datasets to forecast students' grades before significant exams, such as the Mid-II and Final exams, early in their academic journey.

The provided dataset comprises students' assessment scores, including assignments, quizzes, and Mid-I and Mid-II exams. Additionally, it includes a predictor variable indicating the final grade. To facilitate analysis, the data is spread across seven sheets, each varying in the number of assignments and quizzes included. Notably, before calculating grades, only the best five assignments and quizzes for each student have been considered, as specified in the dataset.

Exploratory Data Analysis (EDA) plays a pivotal role in understanding and preprocessing the dataset. It serves as the foundation for subsequent phases of the project. Through EDA, the goal is to uncover hidden patterns, detect anomalies, and gain insights into the relationships between different assessment types and the final grades.

By thoroughly exploring the data, informed decisions can be made regarding preprocessing steps and model selection. This, in turn, enhances the accuracy of grade predictions, making it an indispensable part of the project.

# Data Preprocessing

In preparation for model training and analysis, the dataset underwent several preprocessing steps to ensure its suitability for predictive modeling. Below, we outline the preprocessing steps performed, along with the rationale behind each step:

## Combining Data from Multiple Sheets:

The dataset provided was divided across seven sheets, each containing different subsets of assessment scores.

To streamline analysis and modeling, data from all sheets were extracted and consolidated into a single CSV file.

This consolidation ensures consistency in data handling and facilitates seamless processing during subsequent phases of the project.

## Creation of Two Separate Datasets:

Given the project's objectives to predict students' grades before and after the Mid-II exam, two distinct datasets were created.

The first dataset is intended for predicting grades before the Mid-II exam and includes the following attributes: 'As:1', 'As:2', 'As:3', 'As:4', 'Qz:1', 'Qz:2', 'Qz:3', 'Qz:4', and 'S-I'.

The second dataset is tailored for predicting grades after the Mid-II exam and contains attributes: 'As', 'Qz', 'S-I', and 'S-II'.

This separation allows for focused analysis and model training based on the specific requirements of each prediction scenario.

## Handling Missing Values:

Missing values in the dataset can pose challenges during analysis and modeling.

For this project, missing values were identified and addressed by imputing zeros ('0').

Since the goal is to predict students' grades, alternative methods such as mean value imputation were considered unsuitable, as they could potentially distort the predictive accuracy by artificially inflating or deflating scores.

Imputing zeros maintains the integrity of the dataset while ensuring that missing values do not adversely affect grade predictions.

# Summary Statistics

## Before Mid-II Exam:

### Assignment Scores:

As:1:

Mean = 66.12, Standard Deviation = 33.07

Min = 0, Max = 127, IQR = 48.375

As:2:

Mean = 59.40, Standard Deviation = 25.83

Min = 0, Max = 100, IQR = 32.750

As:3:

Mean = 84.62, Standard Deviation = 34.11

Min = 0, Max = 140, IQR = 39.875

As:4:

Mean = 54.09, Standard Deviation = 31.70

Min = 0, Max = 130, IQR = 36.000

### Quiz Scores:

Qz:1:

Mean = 5.55, Standard Deviation = 4.25

Min = 0, Max = 20, IQR = 4.500

Qz:2:

Mean = 6.77, Standard Deviation = 6.39

Min = 0, Max = 30.5, IQR = 7.875

Qz:3:

Mean = 3.39, Standard Deviation = 3.48

Min = 0, Max = 17.5, IQR = 4.400

Qz:4:

Mean = 3.42, Standard Deviation = 3.82

Min = 0, Max = 10, IQR = 7.000

### Mid-I Score:

Mean = 5.72, Standard Deviation = 2.348

Min = 0, Max = 13.87, IQR = 2.885

## Before Final Exam:

### Assignment Scores:

Mean = 11.08, Standard Deviation = 2.52

Min = 0, Max = 14.87, IQR = 2.88

Quiz Scores:

Mean = 5.60, Standard Deviation = 1.97

Min = 0.2, Max = 10, IQR = 2.665

Mid-II Score:

Mean = 4.89, Standard Deviation = 2.71

Min = 0, Max = 12.37, IQR = 4.108

# Model Evaluation

## Predicting Grades before Mid-II Exam

For this prediction task, we utilized the first four assignments, the first four quizzes, and the Mid-I score as features.

### Nearest Neighbor Classifier:

Accuracy: 69.64%

**Precision:**

Class 0: 67%

Class 1: 72%

**Recall:**

Class 0: 69%

Class 1: 70%

**F1-score:**

Class 0: 68%

Class 1: 71%

### Decision Tree Classifier:

Accuracy: 76.79%

**Precision:**

Class 0: 72%

Class 1: 81%

**Recall:**

Class 0: 81%

Class 1: 73%

**F1-score:**

Class 0: 76%

Class 1: 77%

## Predicting Grades after Mid-II Exam

For this prediction task, we used the best five assignments, the best five quizzes, and the Mid-I score as features.

### Nearest Neighbor Classifier:

Accuracy: 87.5%

**Precision:**

Class 0: 88%

Class 1: 87%

**Recall:**

Class 0: 85%

Class 1: 90%

**F1-score:**

Class 0: 86%

Class 1: 89%

### Decision Tree Classifier:

Accuracy: 87.5%

**Precision:**

Class 0: 100%

Class 1: 81%

**Recall:**

Class 0: 73%

Class 1: 100%

**F1-score:**

Class 0: 84%

Class 1: 90%

# Codes

## Pre-processing codes:

### Before final data extraction:

import pandas as pd  
  
# Read the Excel file  
excel\_file = "your\_file.xlsx"  
xls = pd.ExcelFile(excel\_file)  
  
# Dictionary to store data frames for each sheet  
sheet\_dict = {}  
  
# Loop through each sheet and extract required columns  
for sheet\_name in xls.sheet\_names:  
 df = pd.read\_excel(excel\_file, sheet\_name=sheet\_name)  
 sheet\_dict[sheet\_name] = df[['As:1', 'Qz', 'S-I', 'S-II', 'Grade']]  
  
# Write the extracted data to a new CSV file  
output\_csv = "extracted\_data.csv"  
with open(output\_csv, 'w', newline='') as csvfile:  
 for sheet\_name, df in sheet\_dict.items():  
 df.to\_csv(csvfile, mode='a', index=False, header=True)

### Before mid 2 data extraction

import pandas as pd  
  
# Read the Excel file  
excel\_file = "D.xlsx"  
xls = pd.ExcelFile(excel\_file)  
  
# Dictionary to store data frames for each sheet  
sheet\_dict = {}  
  
# Loop through each sheet and extract required columns  
for sheet\_name in xls.sheet\_names:  
 df = pd.read\_excel(excel\_file, sheet\_name=sheet\_name)  
 sheet\_dict[sheet\_name] = df[['As:1','As:2','As:3','As:4','Qz:1','As:2','As:3','Qz:4','S-I','Grade']]  
  
# Write the extracted data to a new CSV file  
output\_csv = "model\_train\_before\_mid2.csv"  
with open(output\_csv, 'w', newline='') as csvfile:  
 for sheet\_name, df in sheet\_dict.items():  
 df.to\_csv(csvfile, mode='a', index=False, header=True)

## Classifiers:

### Decision Tree Classifier:

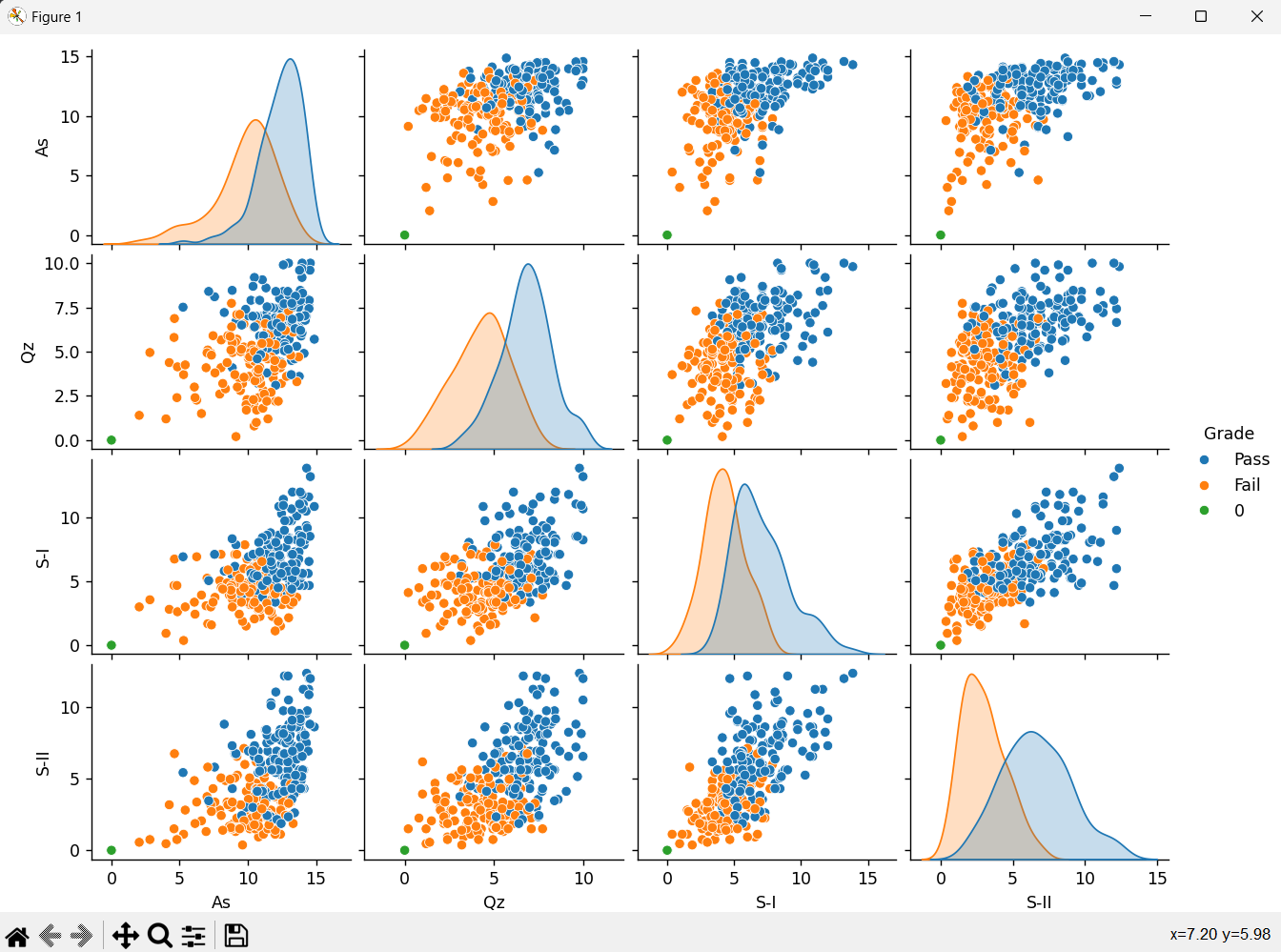
import pandas as pd  
from sklearn.model\_selection import train\_test\_split  
from sklearn.tree import DecisionTreeClassifier  
from sklearn.metrics import accuracy\_score  
  
# Read the cleaned CSV file  
csv\_file = "DM Project/Training Data/Model Train Data Bef Mid 2.csv"  
df = pd.read\_csv(csv\_file)  
  
# Calculate weighted scores for assignments and quizzes  
for i in range(1, 5): # Assuming there are 4 assignments and quizzes  
 df[f'As:{i}'] \*= 3 # Assignments weighted by 3  
 df[f'Qz:{i}'] \*= 2 # Quizzes weighted by 2  
  
# S1 weighted by 15  
df['S-I'] \*= 15  
  
# Split the data into features (X) and target variable (y)  
X = df.drop(columns=['Grade']) # Features  
y = df['Grade'] # Target variable  
  
# Split data into training and testing sets  
X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)  
  
# Initialize the DecisionTreeClassifier  
clf = DecisionTreeClassifier(random\_state=42)  
  
# Train the classifier  
clf.fit(X\_train, y\_train)  
  
# Make predictions  
y\_pred = clf.predict(X\_test)  
  
# Evaluate the model  
accuracy = accuracy\_score(y\_test, y\_pred)  
print("Accuracy:", accuracy)  
  
# Assuming you have a new data point represented as a dictionary  
new\_data = {'As:1': 0.001, 'As:2': 0.001, 'As:3': 0.001, 'As:4': 0.001, 'Qz:1': 0.001, 'Qz:2': 0.001, 'Qz:3': 0.001, 'Qz:4': 0.001, 'S-I': 0.001}  
for i in range(1, 5): # Assuming there are 4 assignments and quizzes  
 new\_data[f'As:{i}'] \*= 3 # Assignments weighted by 3  
 new\_data[f'Qz:{i}'] \*= 2 # Quizzes weighted by 2  
new\_data['S-I'] \*= 15 # S1 weighted by 15  
  
new\_df = pd.DataFrame([new\_data])  
predicted\_grade = clf.predict(new\_df)  
  
print("Predicted Grade of New Entry:", predicted\_grade)

### Nearest Neighbor Classifier:

import pandas as pd  
from sklearn.neighbors import KNeighborsClassifier  
from sklearn.model\_selection import train\_test\_split  
from sklearn.metrics import accuracy\_score  
  
# Load the data from the CSV file  
data = pd.read\_csv('DM Project/Training Data/Model Train Data Bef Mid 2.csv')  
  
# Separate the features (X) and target variable (y)  
X = data.drop('Grade', axis=1)  
y = data['Grade']  
  
# Convert the target variable to numerical values (0 for 'Fail', 1 for 'Pass')  
y = y.replace({'Fail': 0, 'Pass': 1})  
  
# Split the data into training and testing sets  
X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)  
  
# Create a KNeighborsClassifier instance  
knn = KNeighborsClassifier(n\_neighbors=5)  
  
# Train the classifier  
knn.fit(X\_train, y\_train)  
  
# Make predictions on the test set  
y\_pred = knn.predict(X\_test)  
  
# Calculate the accuracy score  
accuracy = accuracy\_score(y\_test, y\_pred)  
print(f"Accuracy: {accuracy\*100:.2f}%")

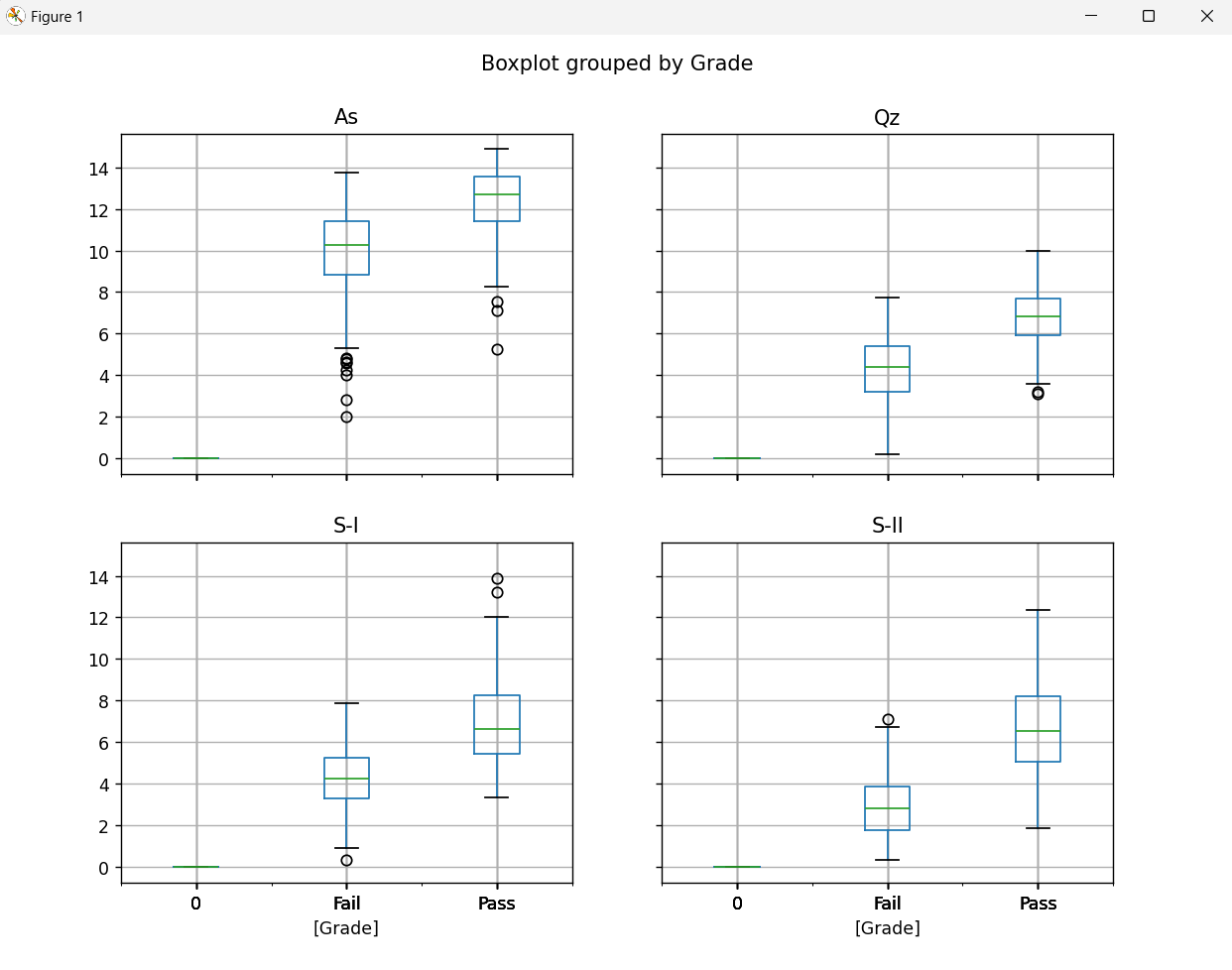
# Visualizations:

## Scatter Plot and Histograms:



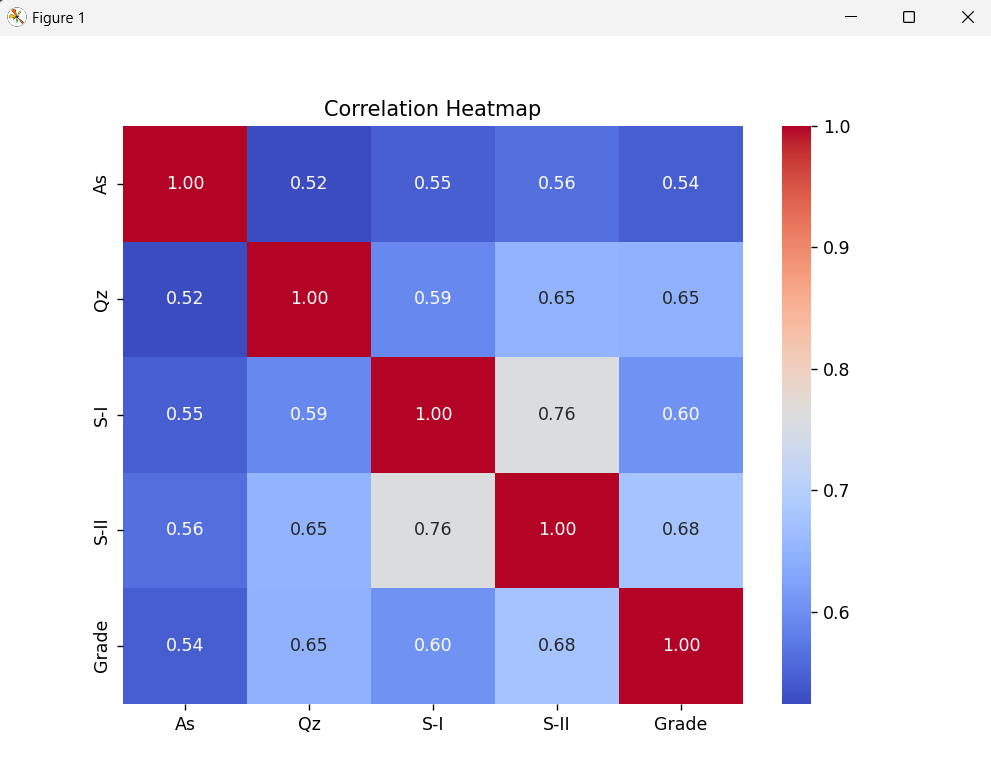
The first visualization presents a series of scatter plots and histograms depicting the relationships between various assessment components and the final grade. The scatter plots show the distribution of students' scores for assignments (As), quizzes (Qz), Mid-I (S-I), and Mid-II (S-II) exams, with points colored according to whether they passed or failed the course. The histograms along the diagonal display the density distributions of each assessment component. From these plots, we can observe potential patterns and trends in the data, such as the separation between passing and failing students based on their scores.

## Box Plot:



The second visualization presents box plots grouped by the final grade, allowing us to compare the distribution of scores for each assessment component between students who passed and those who failed. For assignments and Mid-I, the median scores of passing students are significantly higher than those of failing students, with relatively little overlap in the distributions. However, for quizzes and Mid-II, the distributions show more overlap, indicating that these components may not be as discriminative in predicting the final grade.

## Correlation Heatmap:



The third visualization is a correlation heatmap, which shows the pairwise correlations between all assessment components and the final grade. The diagonal elements represent perfect correlations of 1.0, as each variable is perfectly correlated with itself. The off-diagonal elements reveal moderate to strong positive correlations between the assessment components, ranging from 0.52 (between assignments and quizzes) to 0.76 (between Mid-I and Mid-II). The final grade exhibits the strongest correlations with Mid-II (0.68) and Mid-I (0.60), followed by quizzes (0.65) and assignments (0.54), suggesting that the midterm exams are the most influential components in determining the final grade.

# Conclusion

Based on the exploratory data analysis and visualizations presented in this report, we can draw the following conclusions:

Assessment Components as Grade Predictors: The visualizations clearly indicate that the various assessment components, including assignments, quizzes, and midterm exams, have a significant impact on determining a student's final grade in the course. The scatter plots and correlation heatmap reveal positive correlations between these components and the final grade, with the midterm exams (Mid-I and Mid-II) exhibiting the strongest correlations.

Discriminative Power of Assessments: While all assessment components contribute to predicting the final grade, the box plots suggest that assignments and Mid-I have higher discriminative power in separating passing and failing students. The distributions of scores for these components show relatively little overlap between the two groups, indicating their potential effectiveness as predictors.

Overlapping Distributions for Quizzes and Mid-II: On the other hand, the box plots for quizzes and Mid-II reveal more overlap in the distributions of scores between passing and failing students. This observation suggests that these components alone may not be as effective in accurately predicting the final grade, and additional factors or a combination of assessment components might be required for improved prediction.

Correlation Among Assessment Components: The correlation heatmap highlights moderate to strong positive correlations among the assessment components themselves. This finding implies that students who perform well in one component tend to perform well in others, indicating potential multicollinearity or redundancy in the features.

Moving forward, these insights from the exploratory data analysis will guide the selection and engineering of features for building predictive models in the next phase of the project. Appropriate techniques, such as feature selection, dimensionality reduction, or ensemble methods, may be employed to address potential multicollinearity and leverage the discriminative power of the most informative assessment components.

Additionally, it is essential to consider the limitations of the current analysis and explore ways to enhance the predictive models' performance, such as incorporating additional relevant features or employing more advanced machine learning algorithms.