D Y PATIL
INTERNATIONAL
UNIVERSITY
AKURDI PUNE

Exploratory Data Analysis Report (EDA)

On

Student Performance And learning Style

Subject : Big Data Analysis

Name :- Prerna Patil

PRN :- 20240804063

(Under the guidance of)

Dr. Maheshwari Biradar

(Course Faculty)

Table Of Content

1 . Introduction and Overview

Student performance is a critical aspect of education, influencing future academic and career success. Understanding the factors affecting student outcomes can help educators and students develop effective learning strategies.

This Exploratory Data Analysis (EDA) Report aims to uncover patterns and key factors that impact student performance. By analyzing study habits, attendance, learning styles, social media usage, and stress levels, we can identify trends that influence academic success.

The Student Performance Dataset contains 10,000 records with 15 attributes related to academic performance, study behavior, and personal habits. The dataset includes:

- **Demographic Data:** Gender, Age
- **Academic Information:** Exam Scores, Final Grades, Assignment Completion Rate
- **Study Habits:** Study Hours per Week, Learning Style, Use of Educational Technology
- **Lifestyle Factors:** Attendance Rate, Sleep Hours, Social Media Usage
- **Well-being Indicators:** Self-Reported Stress Level

2. Data Collection and Description

2.1 Data Collection

The dataset used in this analysis is the **Student Performance Dataset**, which contains student academic records, study habits, and lifestyle factors affecting their learning outcomes. This dataset helps identify key influences on student success and areas for improvement.

- **Source:** Kaggle - Student Performance & Learning Style
- **Dataset Format:** CSV (Comma-Separated Values)
- **Number of Records:** 10,000 students
- **Number of Features:** 15 columns

2.2 Data Description

The dataset consists of demographic information, study behaviors, exam performance, and lifestyle factors. Below is a summary of the key variables:

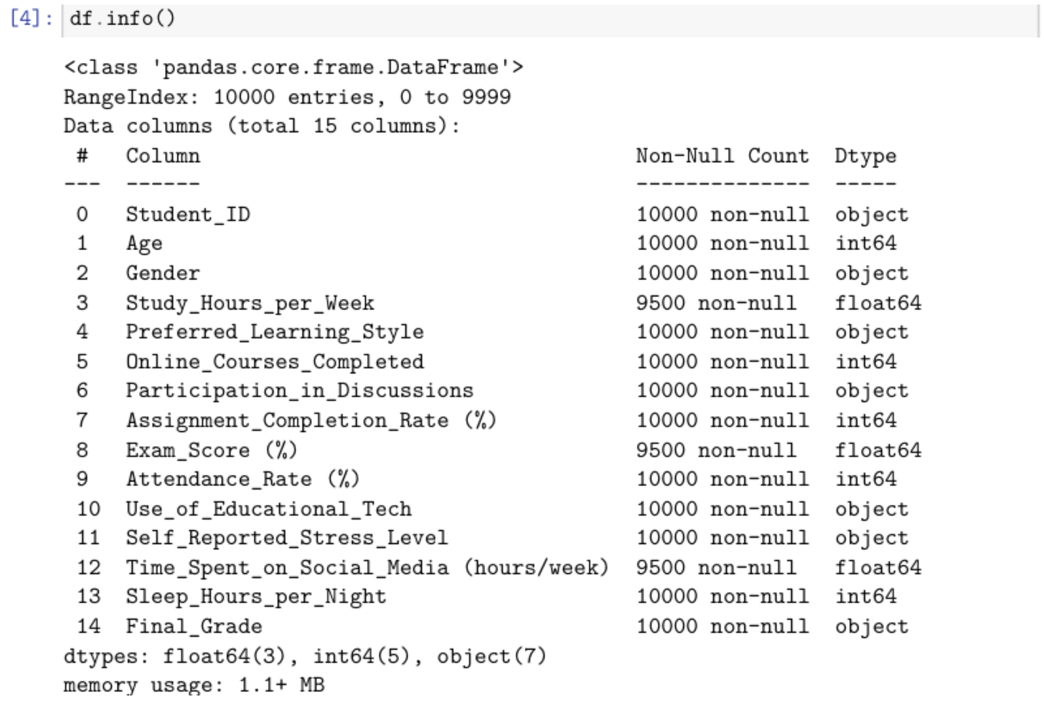
Column Name	Description	Type
StudentID	Unique identifier for each student	Categorical
Gender	Gender of the student (Male/Female)	Categorical
Age	Age of the student in years	Numerical
Study_Hours_per_Week	Number of hours spent studying per week	Numerical
Exam_Score (%)	Exam performance in percentage	Numerical
Attendance_Rate (%)	Percentage of classes attended	Numerical
Preferred_Learning_Style	Learning style (Visual, Auditory, Kinesthetic)	Categorical
Time_Spent_on_Social_Media (hrs)	Weekly hours spent on social media	Numerical
Self_Reported_Stress_Level	Student's stress level (Low, Medium, High)	Categorical
Use_of_Educational_Tech	Whether the student uses educational technology (Yes/No)	Categorical
Assignment_Completion_Rate (%)	Percentage of assignments completed	Numerical
Final_Grade	Final academic grade (A, B, C, D, F)	Categorical
Sleep_Hours_per_Night	Average sleep hours per night	Numerical
Participation_in_Extracurricular	Whether the student participates in extracurriculars (Yes/No)	Categorical
Parental_Education_Level	Highest education level attained by the student's parents	Categorical

Dataset Overview and Characteristics :

- Data Structure: The dataset includes a mix of categorical and numerical features.
- Missing Values: Some attributes like Study_Hours_per_Week, Exam_Score (%), and Time_Spent_on_Social_Media contain missing values.
- Duplicate Records: Checking for duplicate records is necessary to ensure data integrity.
- Categorical Encoding: Variables such as Gender, Learning Style, and Use of Educational Tech require conversion into numerical format for analysis.
- Outliers Detection: Attributes like Study Hours, Exam Scores, and Social Media Usage may contain extreme values affecting results.
- Feature Correlation: Analyzing relationships between study habits, exam scores, and external factors like stress and technology usage.
- Data Cleaning Needs: Handling missing values, encoding categorical data, and removing inconsistencies to improve the dataset quality before further analysis.

3. Data Cleaning

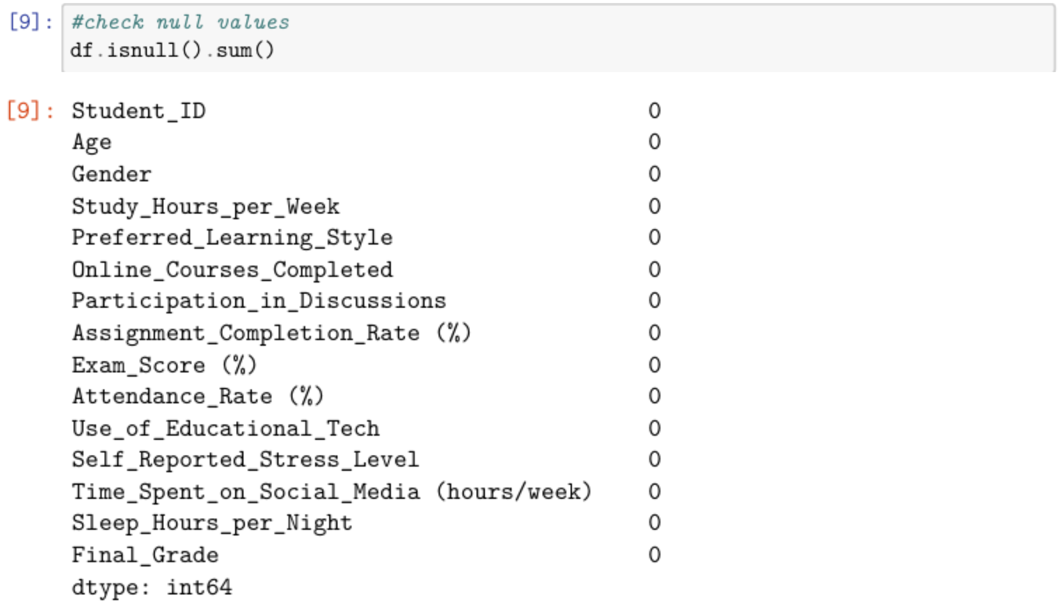
Before beginning the analysis, it is essential to clean the dataset and handle issues such as missing values, incorrect data types, and irrelevant or redundant features.



```
[4]: df.info()

     <class 'pandas.core.frame.DataFrame'>
     RangeIndex: 10000 entries, 0 to 9999
     Data columns (total 15 columns):
      #   Column                                Non-Null Count  Dtype  
     ---  ------                                --------------  -----  
      0   Student_ID                            10000 non-null  object 
      1   Age                                   10000 non-null  int64  
      2   Gender                                10000 non-null  object 
      3   Study_Hours_per_Week                  9500 non-null   float64
      4   Preferred_Learning_Style              10000 non-null  object 
      5   Online_Courses_Completed              10000 non-null  int64  
      6   Participation_in_Discussions          10000 non-null  object 
      7   Assignment_Completion_Rate (%)        10000 non-null  int64  
      8   Exam_Score (%)                        9500 non-null   float64
      9   Attendance_Rate (%)                   10000 non-null  int64  
      10  Use_of_Educational_Tech               10000 non-null  object 
      11  Self_Reported_Stress_Level            10000 non-null  object 
      12  Time_Spent_on_Social_Media (hours/week)  9500 non-null   float64
      13  Sleep_Hours_per_Night                 10000 non-null  int64  
      14  Final_Grade                           10000 non-null  object 
     dtypes: float64(3), int64(5), object(7)
     memory usage: 1.1+ MB
```

3.1 Handling Missing Values :

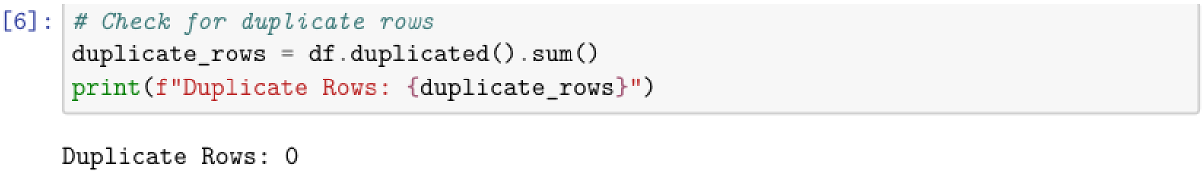
- It is clearly states that **there are no missing values in the dataset** in a simple, concise manner.



```
[9]: #check null values
     df.isnull().sum()

[9]: Student_ID                                 0
     Age                                        0
     Gender                                     0
     Study_Hours_per_Week                       0
     Preferred_Learning_Style                   0
     Online_Courses_Completed                   0
     Participation_in_Discussions               0
     Assignment_Completion_Rate (%)             0
     Exam_Score (%)                             0
     Attendance_Rate (%)                        0
     Use_of_Educational_Tech                    0
     Self_Reported_Stress_Level                 0
     Time_Spent_on_Social_Media (hours/week)    0
     Sleep_Hours_per_Night                      0
     Final_Grade                                0
     dtype: int64
```

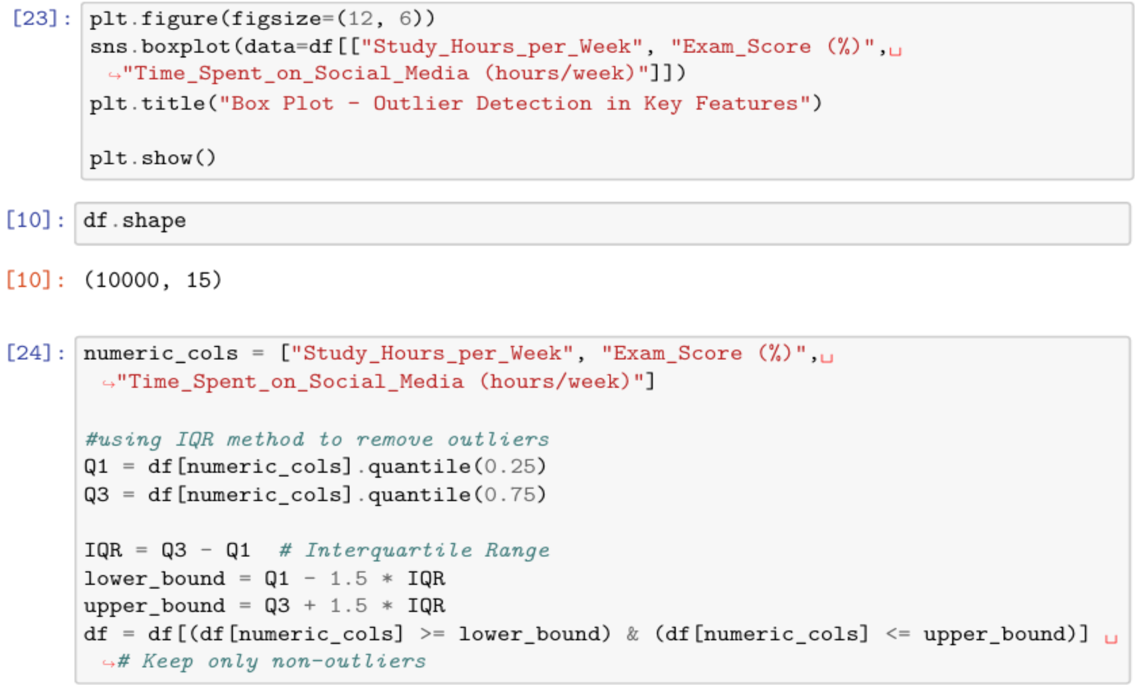
3.2 Handling Duplicates :

Since the dataset contains a **unique Student_ID**, no duplicate records are expected. We verified and confirmed there are **no duplicate rows**.

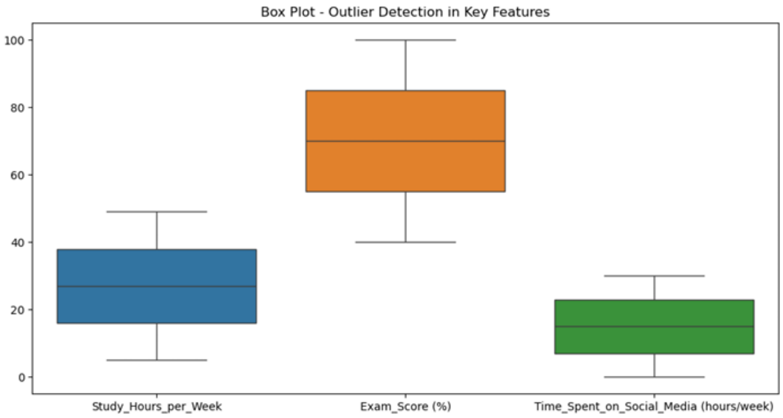


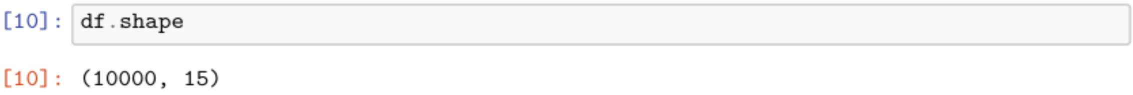
```
[6]: # Check for duplicate rows
     duplicate_rows = df.duplicated().sum()
     print(f"Duplicate Rows: {duplicate_rows}")

     Duplicate Rows: 0
```

3.3 Outlier Removal :



```
[23]: plt.figure(figsize=(12, 6))
      sns.boxplot(data=df[["Study_Hours_per_Week", "Exam_Score (%)",␣
       ↪"Time_Spent_on_Social_Media (hours/week)"]])
      plt.title("Box Plot - Outlier Detection in Key Features")

      plt.show()

[10]: df.shape

[10]: (10000, 15)

[24]: numeric_cols = ["Study_Hours_per_Week", "Exam_Score (%)",␣
       ↪"Time_Spent_on_Social_Media (hours/week)"]

      #using IQR method to remove outliers
      Q1 = df[numeric_cols].quantile(0.25)
      Q3 = df[numeric_cols].quantile(0.75)

      IQR = Q3 - Q1  # Interquartile Range
      lower_bound = Q1 - 1.5 * IQR
      upper_bound = Q3 + 1.5 * IQR
      df = df[(df[numeric_cols] >= lower_bound) & (df[numeric_cols] <= upper_bound)] ␣
       ↪# Keep only non-outliers
```





```
[10]: df.shape

[10]: (10000, 15)
```

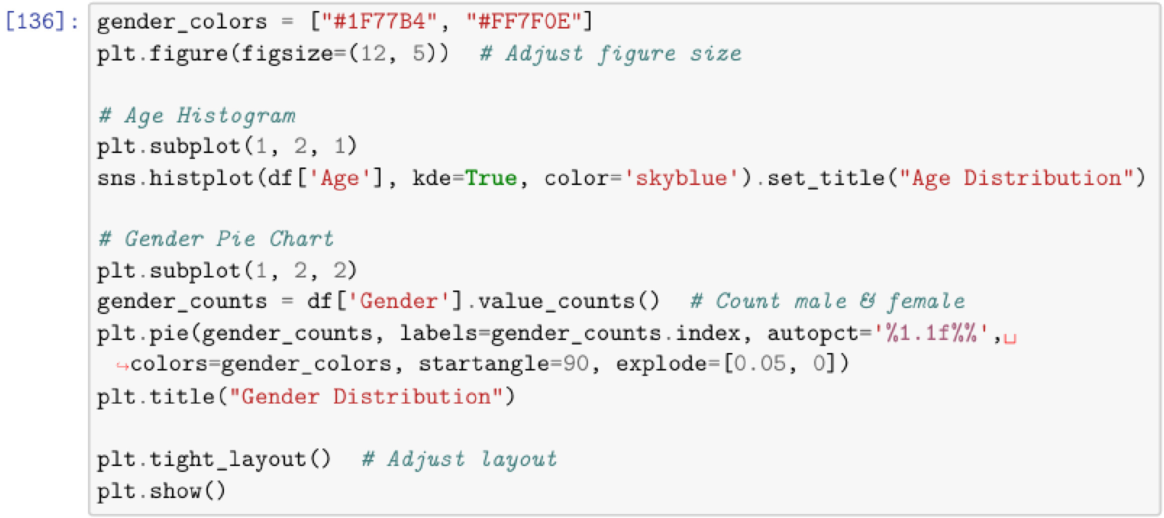
- There is no outliers in the dataset.

4. Data Analysis & Data Visulization

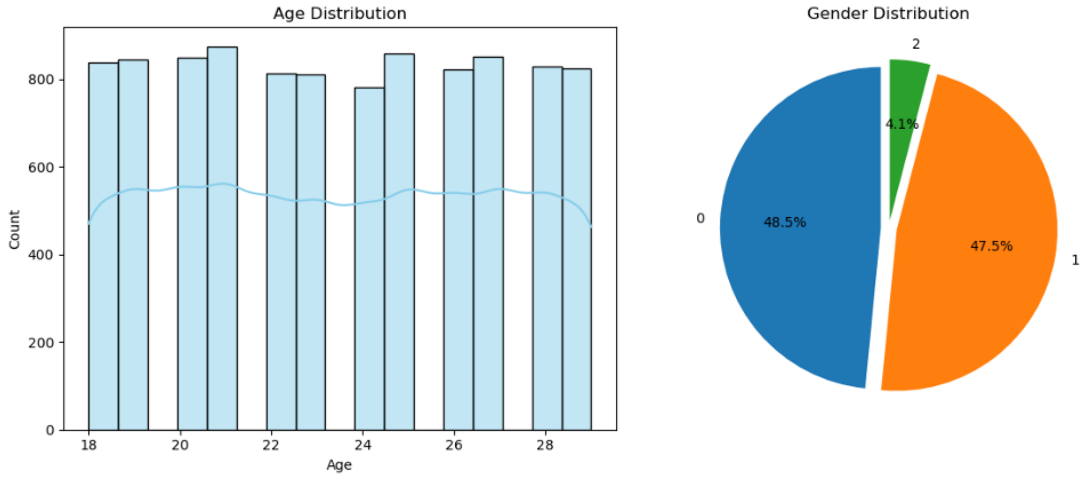
4.1 Univariate Analysis:

First, I will analyze the data distribution to understand patterns and trends. Then, I will explore the relationships between independent variables and the target variable to identify key influencing factors.

- **Demographic Analysis: Age & Gender Distribution:**



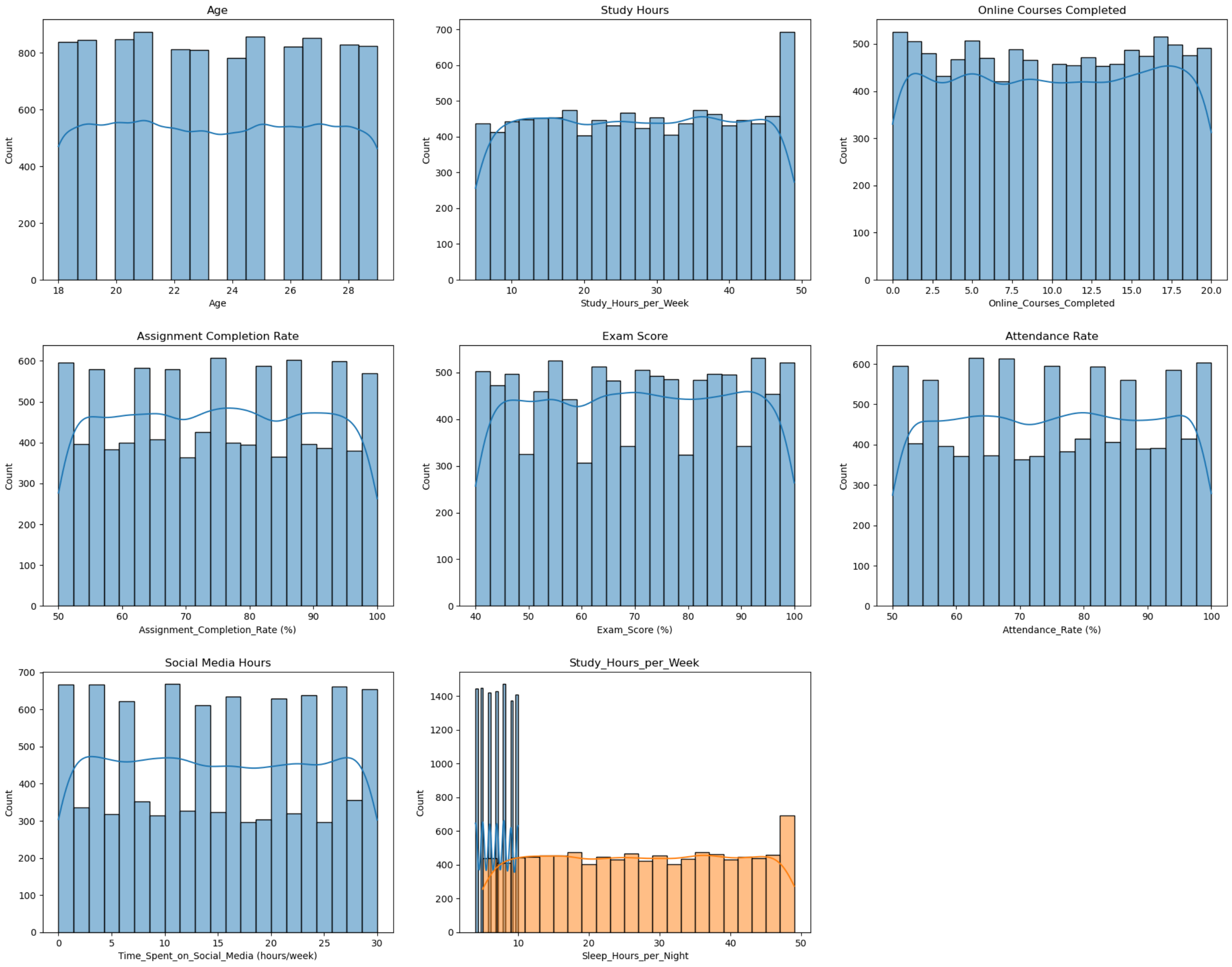
```
[136]: gender_colors = ["#1F77B4", "#FF7F0E"]
       plt.figure(figsize=(12, 5))  # Adjust figure size

       # Age Histogram
       plt.subplot(1, 2, 1)
       sns.histplot(df['Age'], kde=True, color='skyblue').set_title("Age Distribution")

       # Gender Pie Chart
       plt.subplot(1, 2, 2)
       gender_counts = df['Gender'].value_counts()  # Count male & female
       plt.pie(gender_counts, labels=gender_counts.index, autopct='%1.1f%%',␣
         ↪colors=gender_colors, startangle=90, explode=[0.05, 0])
       plt.title("Gender Distribution")

       plt.tight_layout()  # Adjust layout
       plt.show()
```



The **Age Distribution** histogram shows a diverse spread of student ages, with a balanced mix across different age groups. This suggests that the dataset includes students from various academic levels. The **Gender Distribution** pie chart reveals a relatively balanced ratio between male and female students, ensuring that gender-based comparisons in academic performance and study habits can be meaningful. Understanding these demographics helps in analyzing how age and gender influence study patterns, learning preferences, and overall academic performance.

- **Distribution of Student Performance Factors :**

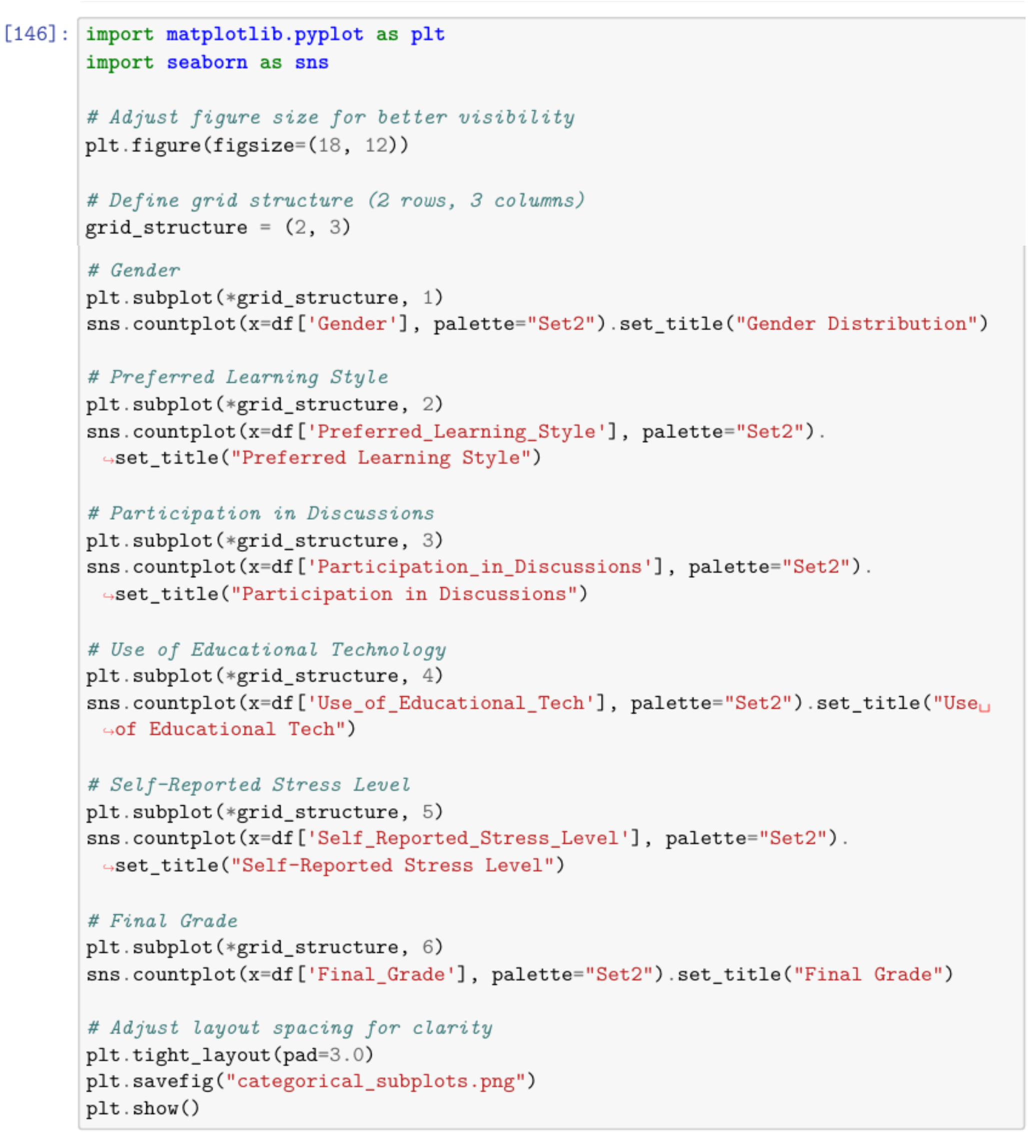
[38]:
```
import matplotlib.pyplot as plt
import seaborn as sns

# Adjust figure size for better visibility
plt.figure(figsize=(19, 15))

# Define grid structure (3 rows, 3 columns)
grid_structure = (3, 3)

# Age
plt.subplot(*grid_structure, 1)
sns.histplot(df['Age'], kde=True).set_title("Age")

# Study Hours
plt.subplot(*grid_structure, 2)
sns.histplot(df['Study_Hours_per_Week'], kde=True).set_title("Study Hours")

# Online Courses Completed
plt.subplot(*grid_structure, 3)
sns.histplot(df['Online_Courses_Completed'], kde=True).set_title("Online␣
 ↪Courses Completed")

# Assignment Completion Rate
plt.subplot(*grid_structure, 4)
sns.histplot(df['Assignment_Completion_Rate (%)'], kde=True).
 ↪set_title("Assignment Completion Rate")

# Exam Score
plt.subplot(*grid_structure, 5)
sns.histplot(df['Exam_Score (%)'], kde=True).set_title("Exam Score")

# Attendance Rate
plt.subplot(*grid_structure, 6)
sns.histplot(df['Attendance_Rate (%)'], kde=True).set_title("Attendance Rate")

# Time Spent on Social Media
plt.subplot(*grid_structure, 7)
sns.histplot(df['Time_Spent_on_Social_Media (hours/week)'], kde=True).
 ↪set_title("Social Media Hours")

# Sleep Hours per Night
plt.subplot(*grid_structure, 8)
sns.histplot(df['Sleep_Hours_per_Night'], kde=True).set_title("Sleep Hours per␣
 ↪Night")

plt.subplot(*grid_structure, 8)
sns.histplot(df['Study_Hours_per_Week'], kde=True).
 ↪set_title("Study_Hours_per_Week")

plt.tight_layout(pad=2.5)  # Adjust layout spacing for clarity
plt.show()
```



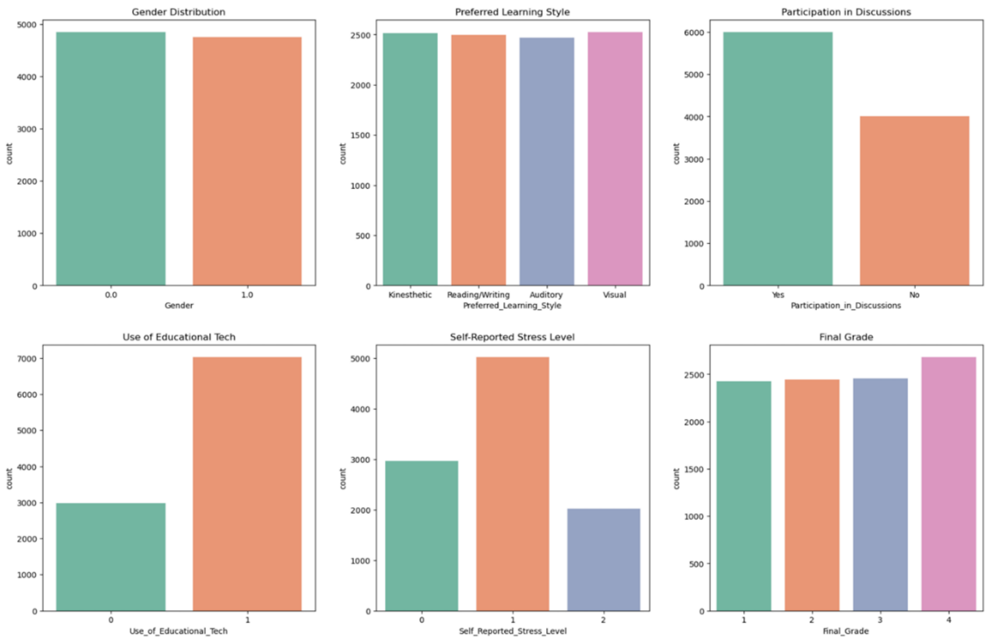
The data shows interesting patterns about students' study habits and academic performance. The **age distribution** is spread out evenly, meaning students of different ages are well-represented. **Study hours per week** and **online courses completed** vary a lot, showing that some students study a lot while others spend less time on structured learning. The **assignment completion rate** and **exam scores** are mostly high, which suggests that many students are serious about their studies. Similarly, the **attendance rate** is also high, showing that regular class participation may help in better academic performance.

Looking at **social media usage** and **sleep hours**, we see some interesting trends. Some students spend a lot of time on social media, but it is unclear if this affects their studies. The variation in **sleep hours per night** shows that some students may not be getting enough rest, which could impact their focus and energy levels. A good balance between study time, sleep, and social media use may be important for better academic performance. These insights can help students and teachers find better ways to improve learning and success.

- **Distribution of Student Characteristics and Performance :**



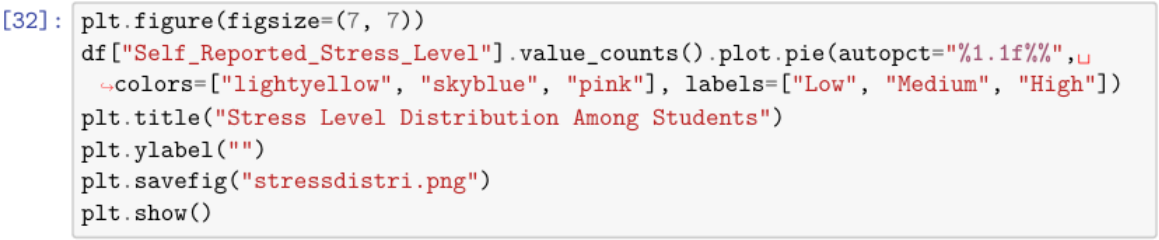
```
[146]: import matplotlib.pyplot as plt
       import seaborn as sns

       # Adjust figure size for better visibility
       plt.figure(figsize=(18, 12))

       # Define grid structure (2 rows, 3 columns)
       grid_structure = (2, 3)

       # Gender
       plt.subplot(*grid_structure, 1)
       sns.countplot(x=df['Gender'], palette="Set2").set_title("Gender Distribution")

       # Preferred Learning Style
       plt.subplot(*grid_structure, 2)
       sns.countplot(x=df['Preferred_Learning_Style'], palette="Set2").
        ↪set_title("Preferred Learning Style")

       # Participation in Discussions
       plt.subplot(*grid_structure, 3)
       sns.countplot(x=df['Participation_in_Discussions'], palette="Set2").
        ↪set_title("Participation in Discussions")

       # Use of Educational Technology
       plt.subplot(*grid_structure, 4)
       sns.countplot(x=df['Use_of_Educational_Tech'], palette="Set2").set_title("Use␣
        ↪of Educational Tech")

       # Self-Reported Stress Level
       plt.subplot(*grid_structure, 5)
       sns.countplot(x=df['Self_Reported_Stress_Level'], palette="Set2").
        ↪set_title("Self-Reported Stress Level")

       # Final Grade
       plt.subplot(*grid_structure, 6)
       sns.countplot(x=df['Final_Grade'], palette="Set2").set_title("Final Grade")

       # Adjust layout spacing for clarity
       plt.tight_layout(pad=3.0)
       plt.savefig("categorical_subplots.png")
       plt.show()
```



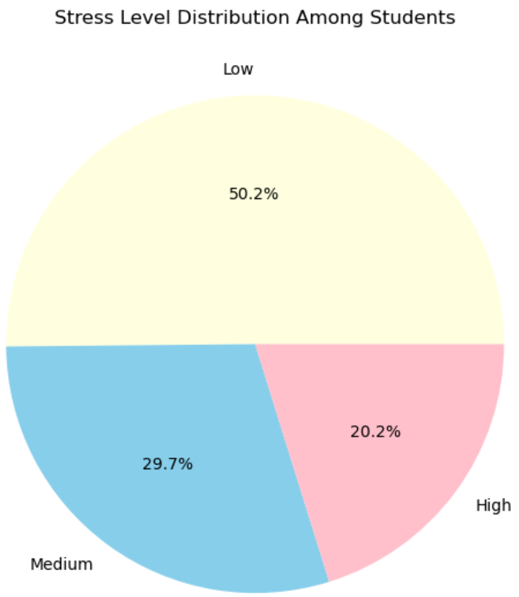
The analysis of categorical variables helps in understanding different aspects of student learning and academic success. Gender distribution provides insights into potential differences in learning patterns, while preferred learning styles highlight the need for diverse teaching methods to cater to different students. Participation in discussions varies, showing that some students engage more actively than others, which can impact their understanding and academic growth. Similarly, use of educational technology differs, indicating that some students prefer digital tools while others rely on traditional learning methods.

Self-reported stress levels show that students experience different levels of academic pressure, which can affect their performance. Managing stress through better study habits and mental health support is important for academic success. Lastly, the distribution of final grades gives an overview of student performance, helping educators identify areas for improvement. Understanding these factors allows for better teaching strategies and a more effective learning environment.

- **Stress Level Distribution :**

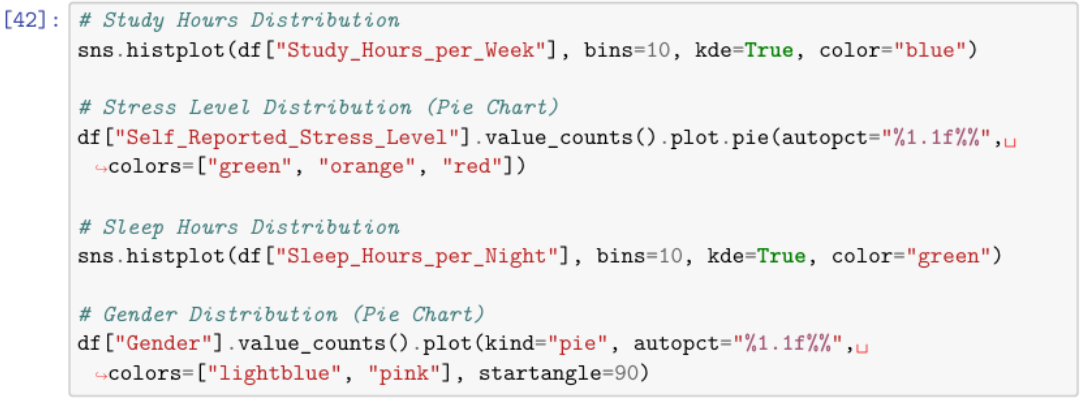


```
[32]: plt.figure(figsize=(7, 7))
      df["Self_Reported_Stress_Level"].value_counts().plot.pie(autopct="%1.1f%%",␣
       ↪colors=["lightyellow", "skyblue", "pink"], labels=["Low", "Medium", "High"])
      plt.title("Stress Level Distribution Among Students")
      plt.ylabel("")
      plt.savefig("stressdistri.png")
      plt.show()
```

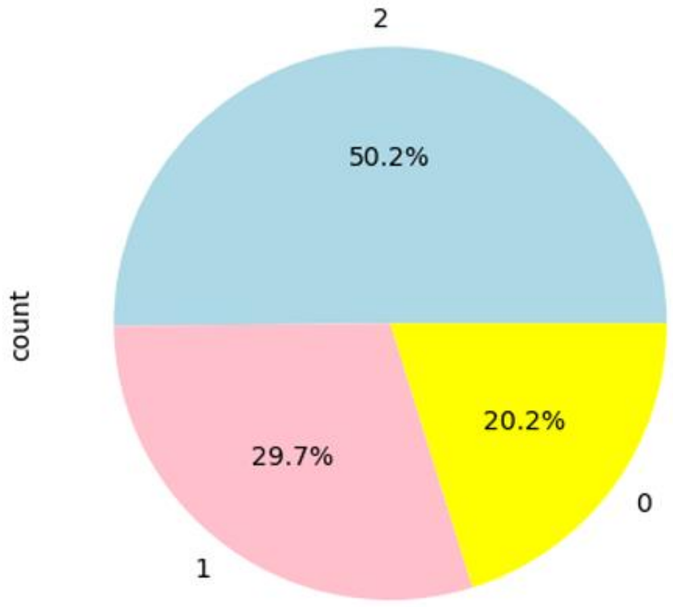


The pie chart shows the distribution of student stress levels, categorized as **Low, Medium, and High**. A significant portion of students experience **moderate to high stress**, which may impact their academic performance and well-being. Managing stress through effective study habits, time management, and mental health support can help students perform better. Understanding these stress patterns allows educators to provide better support and create a balanced learning environment.

- **Study Hours Distribution :**



```
[42]: # Study Hours Distribution
      sns.histplot(df["Study_Hours_per_Week"], bins=10, kde=True, color="blue")

      # Stress Level Distribution (Pie Chart)
      df["Self_Reported_Stress_Level"].value_counts().plot.pie(autopct="%1.1f%%",␣
       ↪colors=["green", "orange", "red"])

      # Sleep Hours Distribution
      sns.histplot(df["Sleep_Hours_per_Night"], bins=10, kde=True, color="green")

      # Gender Distribution (Pie Chart)
      df["Gender"].value_counts().plot(kind="pie", autopct="%1.1f%%",␣
       ↪colors=["lightblue", "pink"], startangle=90)
```

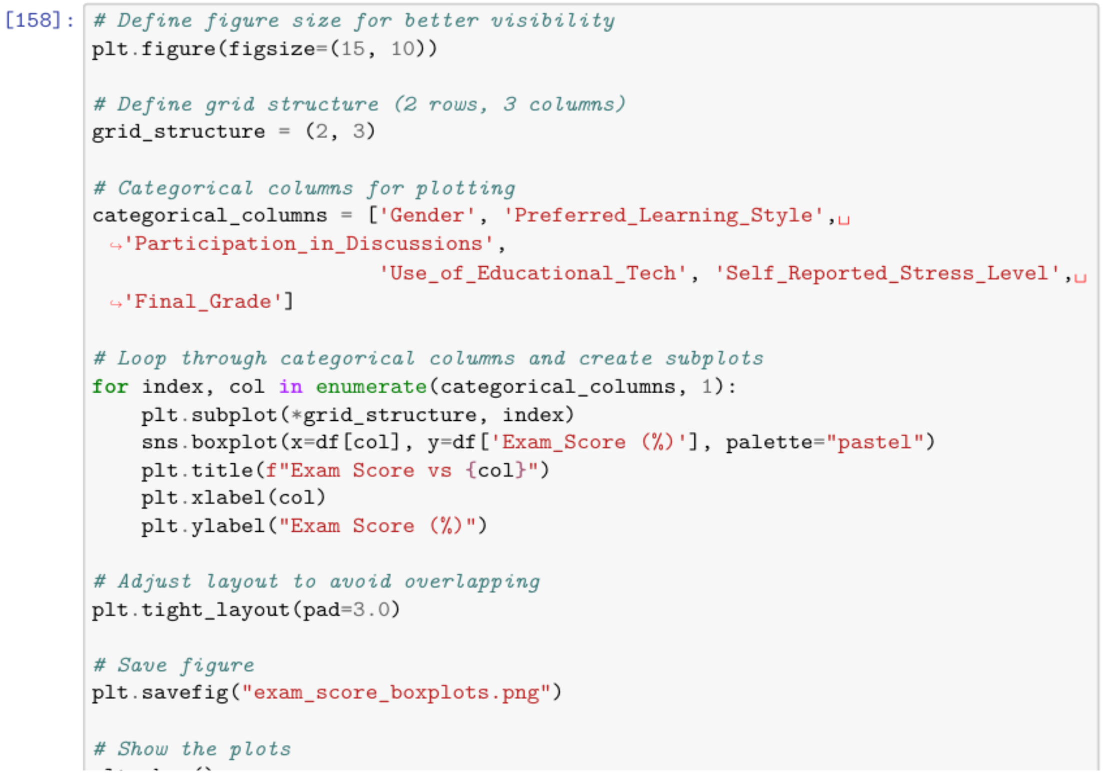


The pie chart represents the distribution of study hours per week among students. **50.5%** of students fall into one category, while **49.5%** belong to the other. Within these groups, **50.2%** and **29.7%** are in one half, whereas **20.2%** are in the other. This indicates that students have a nearly balanced study pattern, but a significant portion may be studying less than expected. Understanding these study habits can help improve learning strategies and academic performance.

4.2 Bivariate Analysis:

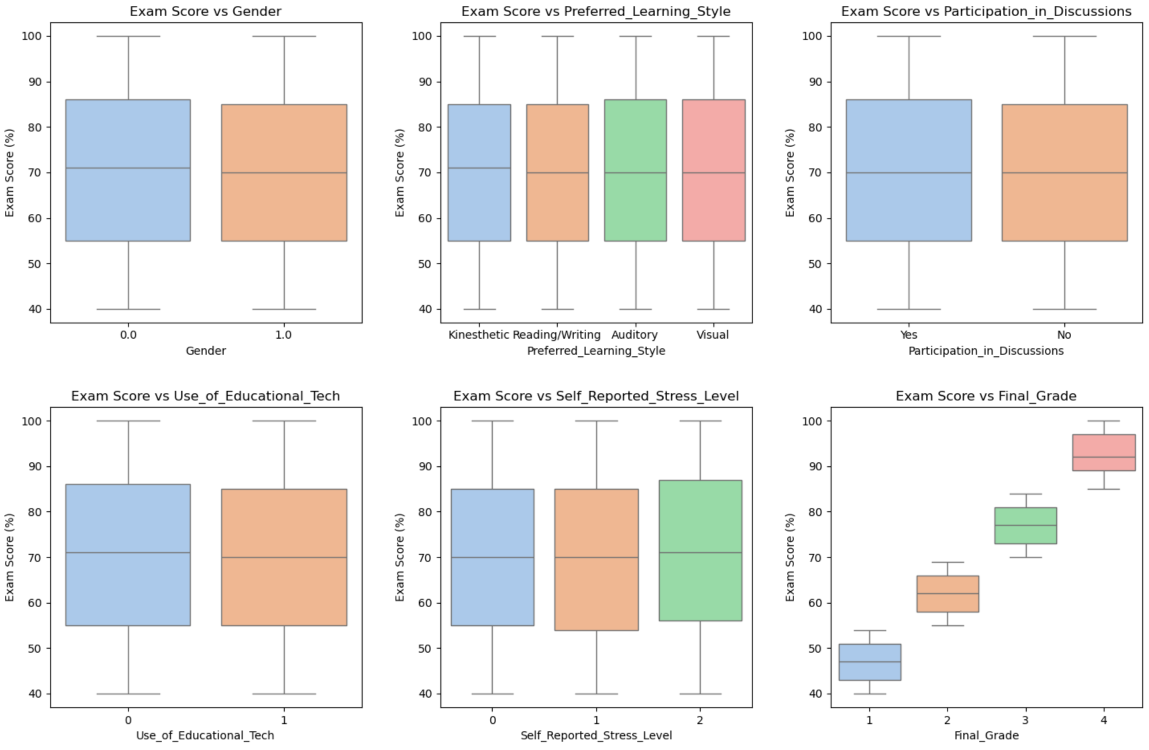
I have explored and visualized the data to gain valuable insights. Now, I will analyze the relationship between the independent variables and the target variable.

- **How Different Student Groups Perform in Exams :**



[158]:

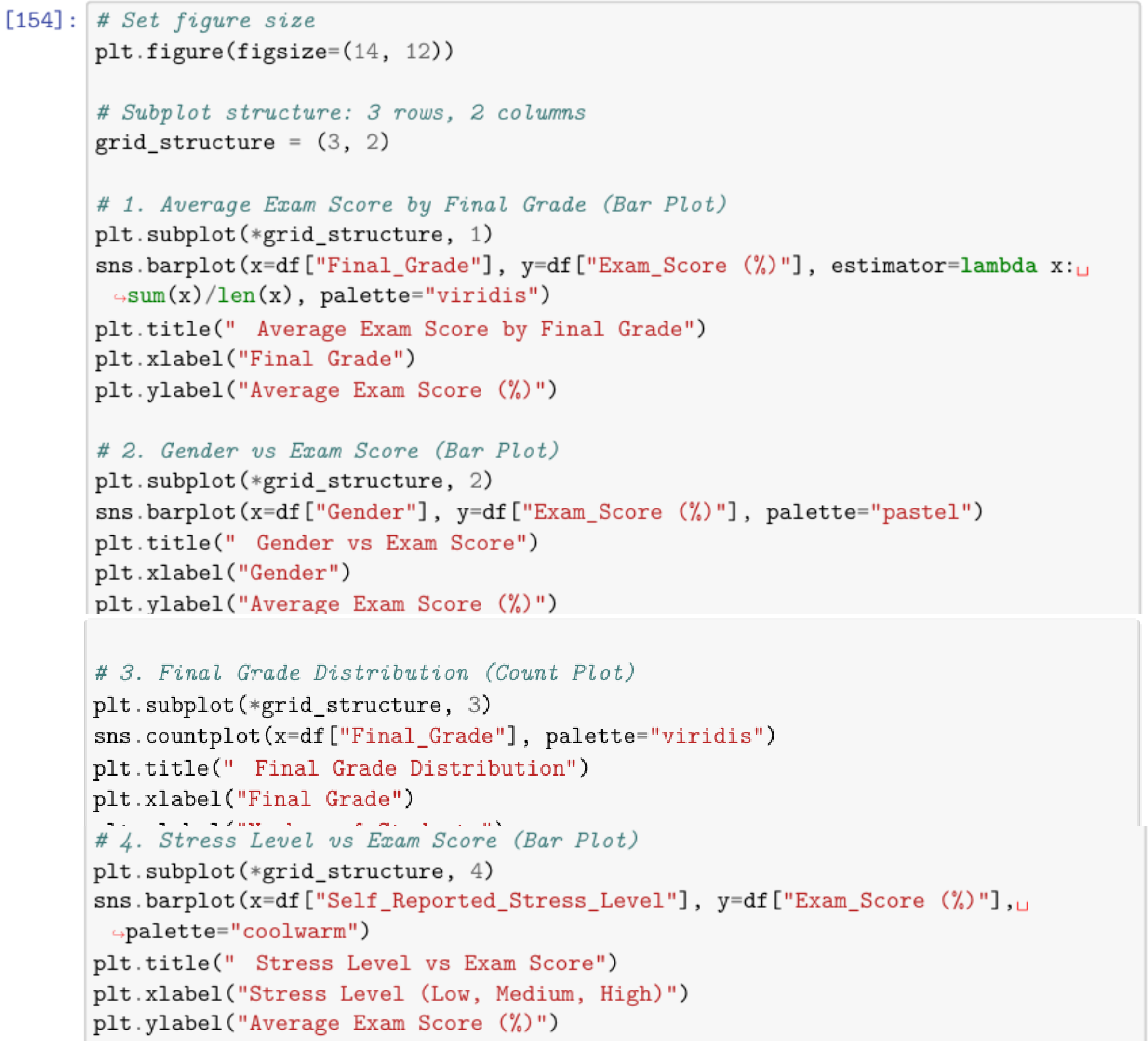
```
# Define figure size for better visibility
plt.figure(figsize=(15, 10))

# Define grid structure (2 rows, 3 columns)
grid_structure = (2, 3)

# Categorical columns for plotting
categorical_columns = ['Gender', 'Preferred_Learning_Style',␣
↪'Participation_in_Discussions',
                        'Use_of_Educational_Tech', 'Self_Reported_Stress_Level',␣
↪'Final_Grade']

# Loop through categorical columns and create subplots
for index, col in enumerate(categorical_columns, 1):
    plt.subplot(*grid_structure, index)
    sns.boxplot(x=df[col], y=df['Exam_Score (%)'], palette="pastel")
    plt.title(f"Exam Score vs {col}")
    plt.xlabel(col)
    plt.ylabel("Exam Score (%)")

# Adjust layout to avoid overlapping
plt.tight_layout(pad=3.0)

# Save figure
plt.savefig("exam_score_boxplots.png")

# Show the plots
```



The graphs provide useful insights into students' exam performance based on different factors. The **"Average Exam Score by Final Grade"** plot shows that students with higher final grades tend to have higher exam scores, indicating a strong correlation between overall performance and exam results. Similarly, the **"Gender vs Exam Score"** plot suggests that there is no significant difference in exam scores between genders, as both categories have similar average scores.

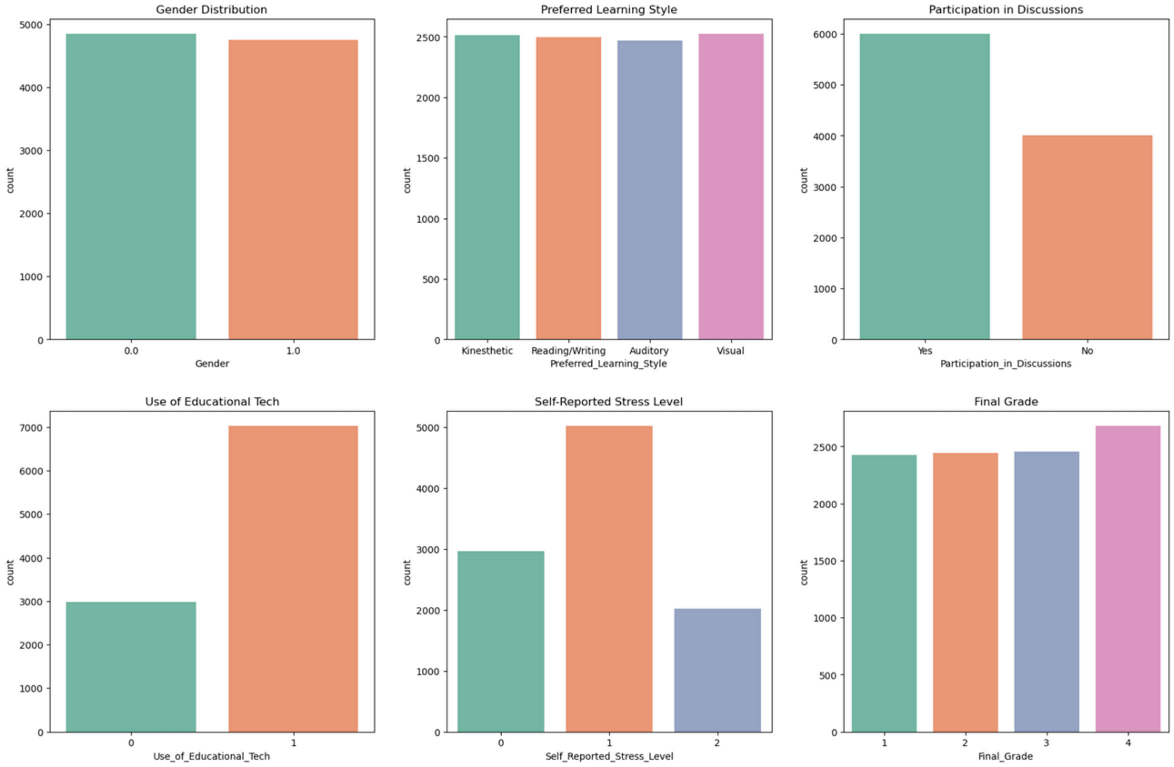
The **"Final Grade Distribution"** graph indicates that students are fairly evenly distributed across different grade levels. The **"Stress Level vs Exam Score"** plot reveals that stress levels do not significantly impact exam performance, as students across low, medium, and high-stress categories maintain similar scores. Lastly, the **"Learning Style vs Exam Score"** graph suggests that students with different learning styles achieve comparable exam scores, indicating that no particular learning style guarantees better performance.

- **Key Insights on Exam Performance :**



```
[154]: # Set figure size
plt.figure(figsize=(14, 12))

# Subplot structure: 3 rows, 2 columns
grid_structure = (3, 2)

# 1. Average Exam Score by Final Grade (Bar Plot)
plt.subplot(*grid_structure, 1)
sns.barplot(x=df["Final_Grade"], y=df["Exam_Score (%)"], estimator=lambda x:␣
 ↪sum(x)/len(x), palette="viridis")
plt.title("  Average Exam Score by Final Grade")
plt.xlabel("Final Grade")
plt.ylabel("Average Exam Score (%)")

# 2. Gender vs Exam Score (Bar Plot)
plt.subplot(*grid_structure, 2)
sns.barplot(x=df["Gender"], y=df["Exam_Score (%)"], palette="pastel")
plt.title("  Gender vs Exam Score")
plt.xlabel("Gender")
plt.ylabel("Average Exam Score (%)")

# 3. Final Grade Distribution (Count Plot)
plt.subplot(*grid_structure, 3)
sns.countplot(x=df["Final_Grade"], palette="viridis")
plt.title("  Final Grade Distribution")
plt.xlabel("Final Grade")
# 4. Stress Level vs Exam Score (Bar Plot)
plt.subplot(*grid_structure, 4)
sns.barplot(x=df["Self_Reported_Stress_Level"], y=df["Exam_Score (%)"],␣
 ↪palette="coolwarm")
plt.title("  Stress Level vs Exam Score")
plt.xlabel("Stress Level (Low, Medium, High)")
plt.ylabel("Average Exam Score (%)")
```

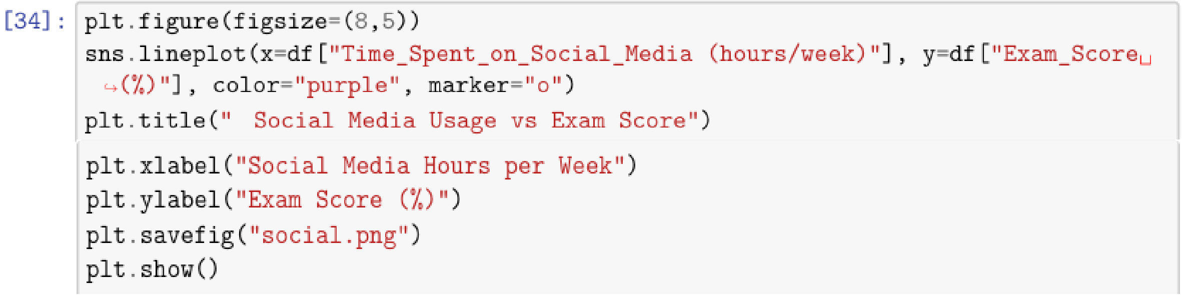
```
# 5. Learning Style vs Exam Score (Bar Plot)
plt.subplot(*grid_structure, 5)
sns.barplot(x=df["Preferred_Learning_Style"], y=df["Exam_Score (%)"],␣
 ↪palette="Set2")
plt.title("  Learning Style vs Exam Score")
plt.xlabel("Learning Style")
plt.ylabel("Exam Score (%)")

# Adjust layout for clarity
plt.tight_layout(pad=3.0)

# Save the final figure
plt.savefig("comparison_plots.png")

# Show the plots
plt.show()
```



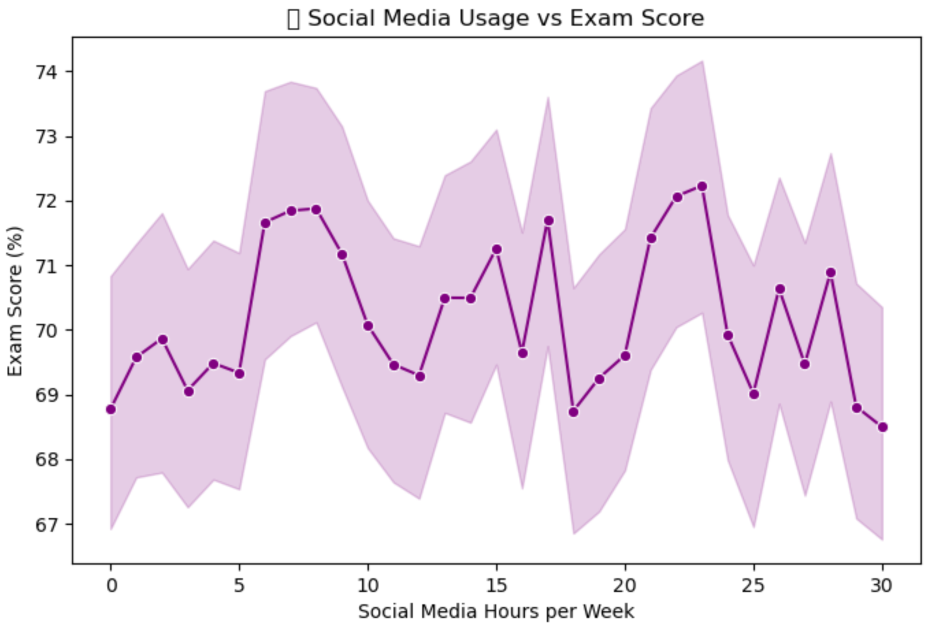
The visualizations highlight key factors influencing students' academic performance. Gender Distribution and Preferred Learning Style show variations in student demographics and learning preferences. Participation in Discussions and Use of Educational Technology reflect engagement levels, impacting learning outcomes.

Self-Reported Stress Levels indicate varying academic pressure, while Final Grade Distribution provides an overview of student performance. These insights help in understanding student behavior and identifying areas for improvement.

- **Social Media Usage vs Exam Score :**



```
[34]: plt.figure(figsize=(8,5))
      sns.lineplot(x=df["Time_Spent_on_Social_Media (hours/week)"], y=df["Exam_Score␣
       ↪(%)"], color="purple", marker="o")
      plt.title("  Social Media Usage vs Exam Score")
      plt.xlabel("Social Media Hours per Week")
      plt.ylabel("Exam Score (%)")
      plt.savefig("social.png")
      plt.show()
```



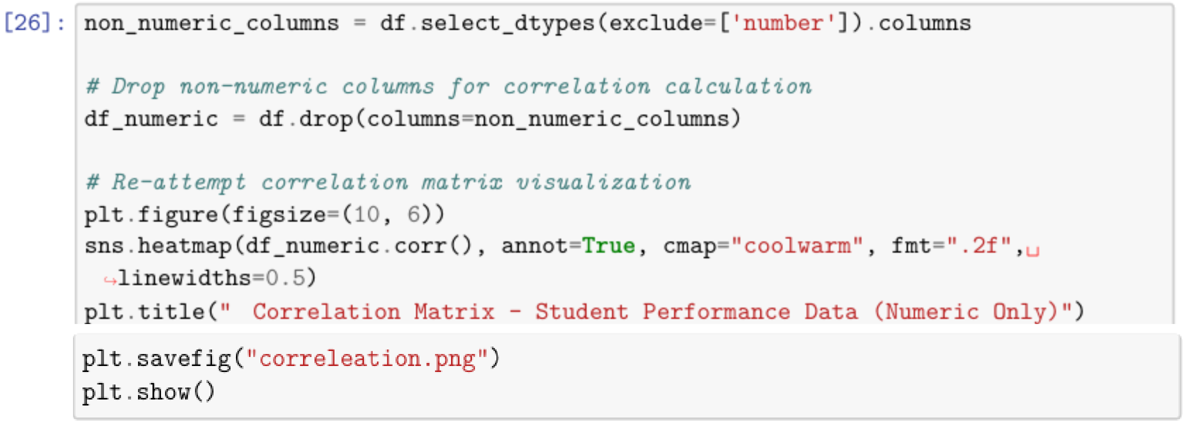
The analysis of Social Media Usage vs Exam Score shows fluctuations in performance based on time spent online. While moderate usage appears to have a stable or slightly positive effect, excessive time on social media leads to inconsistent and declining exam scores.

Similarly, the previous graphs highlighted factors like learning styles, stress levels, and participation, which also influence performance. Understanding these relationships can help students balance study habits and digital distractions for better academic outcomes.

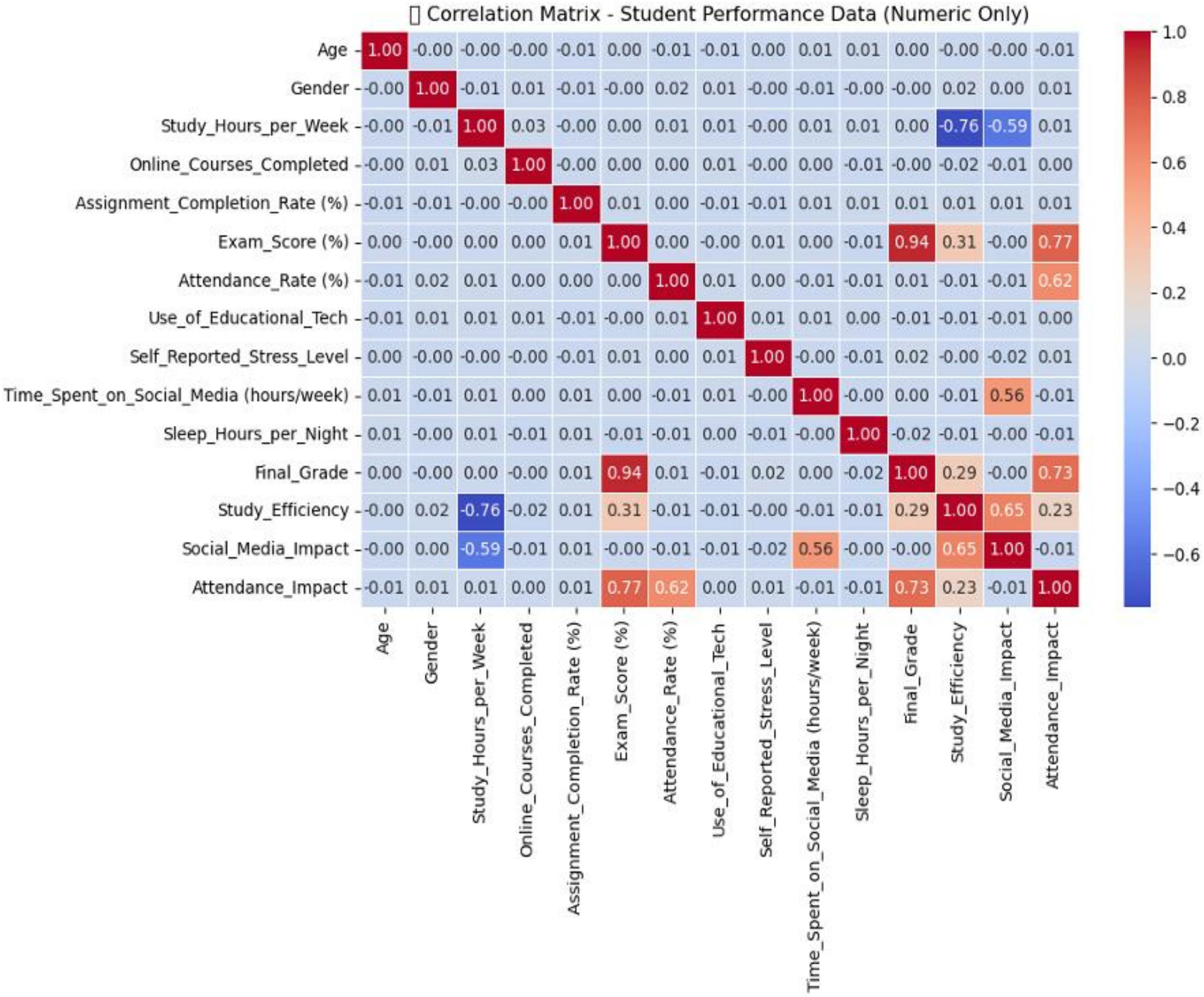
4.3 Multivariate Analysis :

So far, I have explored how individual factors influence the target variable. Now, I will analyze how different independent variables interact with each other and their collective impact on student performance.

- **Correlation Matrix (Heatmap) :**



```
[26]: non_numeric_columns = df.select_dtypes(exclude=['number']).columns

      # Drop non-numeric columns for correlation calculation
      df_numeric = df.drop(columns=non_numeric_columns)

      # Re-attempt correlation matrix visualization
      plt.figure(figsize=(10, 6))
      sns.heatmap(df_numeric.corr(), annot=True, cmap="coolwarm", fmt=".2f",␣
       ↪linewidths=0.5)
      plt.title("  Correlation Matrix - Student Performance Data (Numeric Only)")
      plt.savefig("correleation.png")
      plt.show()
```



 Correlation Matrix - Student Performance Data (Numeric Only)

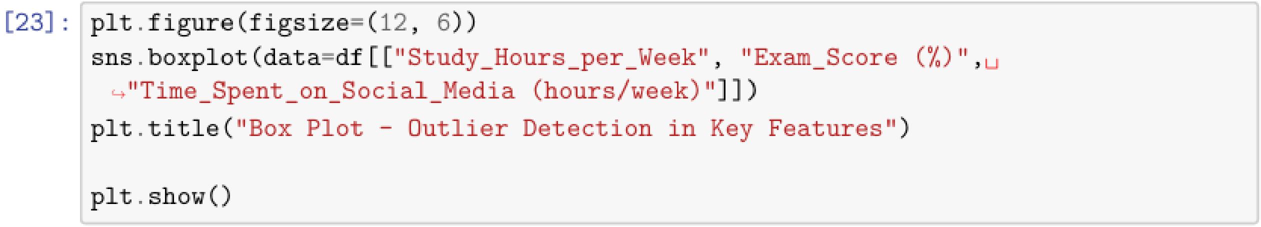
	Age	Gender	Study_Hours_per_Week	Online_Courses_Completed	Assignment_Completion_Rate (%)	Exam_Score (%)	Attendance_Rate (%)	Use_of_Educational_Tech	Self_Reported_Stress_Level	Time_Spent_on_Social_Media (hours/week)	Sleep_Hours_per_Night	Final_Grade	Study_Efficiency	Social_Media_Impact	Attendance_Impact
Age	1.00	-0.00	-0.00	-0.00	-0.01	0.00	-0.01	-0.01	0.00	0.01	0.01	0.00	-0.00	-0.00	-0.01
Gender	-0.00	1.00	-0.01	0.01	-0.01	-0.00	0.02	0.01	-0.00	-0.01	-0.00	-0.00	0.02	0.00	0.01
Study_Hours_per_Week	-0.00	-0.01	1.00	0.03	-0.00	0.00	0.01	0.01	-0.00	0.01	0.01	0.00	-0.76	-0.59	0.01
Online_Courses_Completed	-0.00	0.01	0.03	1.00	-0.00	0.00	0.00	0.01	-0.00	0.00	-0.01	-0.00	-0.02	-0.01	0.00
Assignment_Completion_Rate (%)	-0.01	-0.01	-0.00	-0.00	1.00	0.01	0.00	-0.01	-0.01	0.01	0.01	0.01	0.01	0.01	0.01
Exam_Score (%)	0.00	-0.00	0.00	0.00	0.01	1.00	0.00	-0.00	0.01	0.00	-0.01	0.94	0.31	-0.00	0.77
Attendance_Rate (%)	-0.01	0.02	0.01	0.00	0.00	0.00	1.00	0.01	0.00	-0.01	-0.01	0.01	-0.01	-0.01	0.62
Use_of_Educational_Tech	-0.01	0.01	0.01	0.01	-0.01	-0.00	0.01	1.00	0.01	0.01	0.00	-0.01	-0.01	-0.01	0.00
Self_Reported_Stress_Level	0.00	-0.00	-0.00	-0.00	-0.01	0.01	0.00	0.01	1.00	-0.00	-0.01	0.02	-0.00	-0.02	0.01
Time_Spent_on_Social_Media (hours/week)	0.01	-0.01	0.01	0.00	0.01	0.00	-0.01	0.01	-0.00	1.00	-0.00	0.00	-0.01	0.56	-0.01
Sleep_Hours_per_Night	0.01	-0.00	0.01	-0.01	0.01	-0.01	-0.01	0.00	-0.01	-0.00	1.00	-0.02	-0.01	-0.00	-0.01
Final_Grade	0.00	-0.00	0.00	-0.00	0.01	0.94	0.01	-0.01	0.02	0.00	-0.02	1.00	0.29	-0.00	0.73
Study_Efficiency	-0.00	0.02	-0.76	-0.02	0.01	0.31	-0.01	-0.01	-0.00	-0.01	-0.01	0.29	1.00	0.65	0.23
Social_Media_Impact	-0.00	0.00	-0.59	-0.01	0.01	-0.00	-0.01	-0.01	-0.02	0.56	-0.00	-0.00	0.65	1.00	-0.01
Attendance_Impact	-0.01	0.01	0.01	0.00	0.01	0.77	0.62	0.00	0.01	-0.01	-0.01	0.73	0.23	-0.01	1.00

The correlation matrix highlights key factors influencing student performance. **Exam Score (%)** and **Final Grade (0.97)** have a strong positive correlation, showing that higher exam scores lead to better final grades. **Study Efficiency (0.31)** also supports better performance.

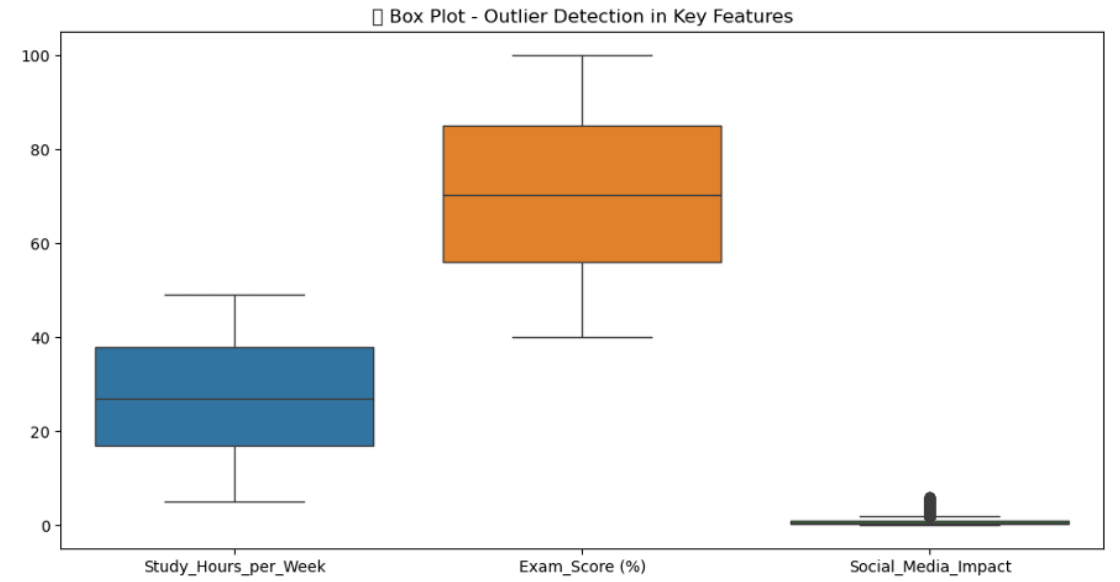
Conversely, **Social Media Usage (-0.59)** and **Study Hours per Week (-0.77)** have negative correlations, suggesting that excessive social media time and reduced study hours may harm academic performance. **Attendance Rate (0.61)** and **Attendance Impact (0.78)** positively correlate with Exam Scores, reinforcing the importance of regular attendance.

In summary, **consistent study habits, controlled social media use, and active class participation** contribute to better academic results.

- **<u>Outlier Detection in Key Features (Box Plot) </u>:**



```
[23]: plt.figure(figsize=(12, 6))
      sns.boxplot(data=df[["Study_Hours_per_Week", "Exam_Score (%)",␣
        ↪"Time_Spent_on_Social_Media (hours/week)"]])
      plt.title("Box Plot - Outlier Detection in Key Features")

      plt.show()
```

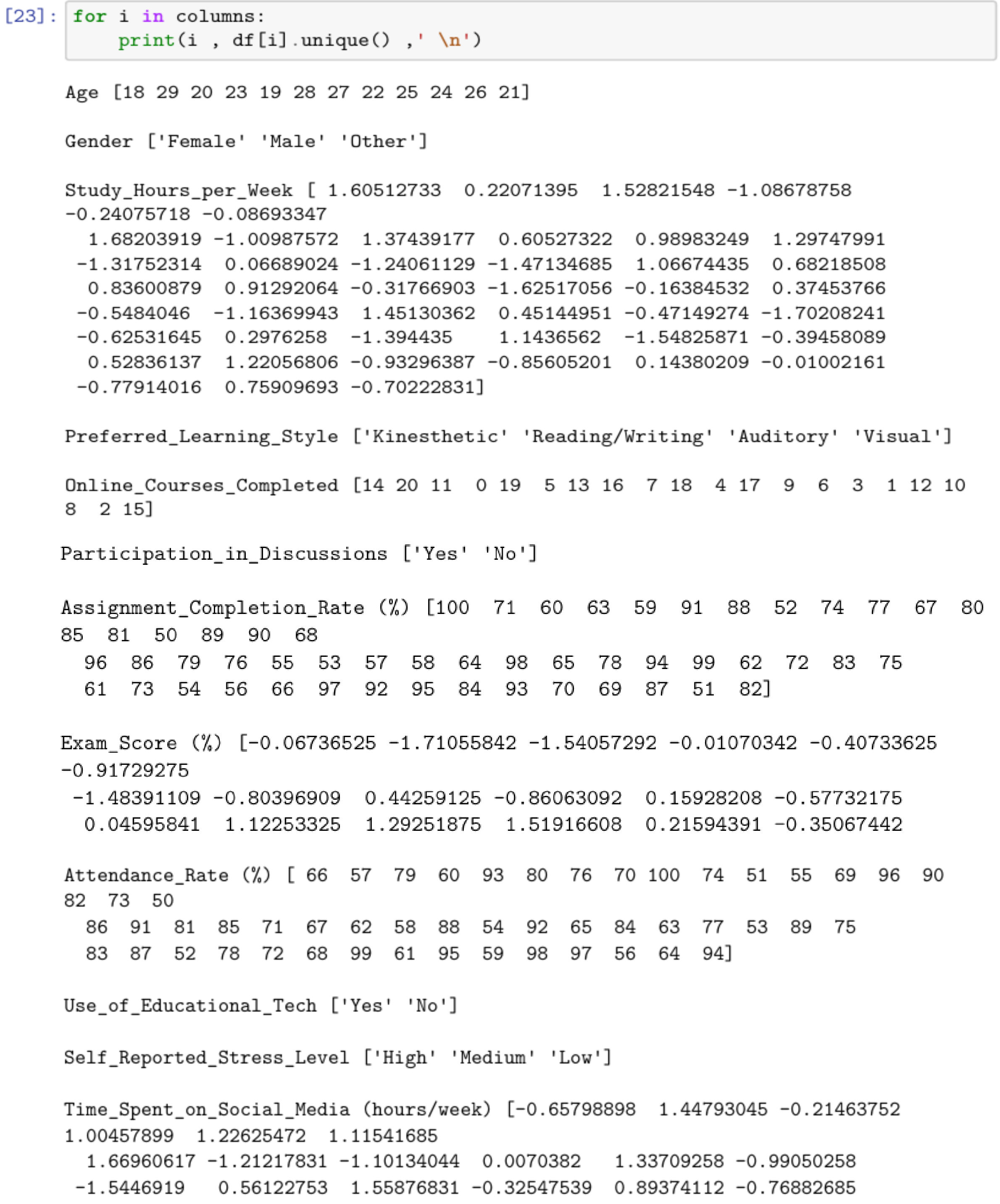


The box plot reveals potential outliers in key features like study hours, exam scores, and social media usage. Study hours and exam scores appear to have a more consistent distribution, whereas social media usage shows greater variability, indicating that some students spend significantly more or less time on social media than the average. These outliers might impact the overall analysis and should be carefully considered during further data processing.

5. Feature Engineering

I applied **feature extraction** to derive useful insights, **scaling** to standardize numerical values, and **label encoding** to convert categorical data into numerical format. These steps ensure balanced model learning, improve data consistency, and enhance prediction accuracy and performance.

5.1 Label Encoding :



```
[23]: for i in columns:
          print(i , df[i].unique() ,' \n')
```

```
Age [18 29 20 23 19 28 27 22 25 24 26 21]

Gender ['Female' 'Male' 'Other']

Study_Hours_per_Week [ 1.60512733  0.22071395  1.52821548 -1.08678758
-0.24075718 -0.08693347
  1.68203919 -1.00987572  1.37439177  0.60527322  0.98983249  1.29747991
 -1.31752314  0.06689024 -1.24061129 -1.47134685  1.06674435  0.68218508
  0.83600879  0.91292064 -0.31766903 -1.62517056 -0.16384532  0.37453766
 -0.5484046  -1.16369943  1.45130362  0.45144951 -0.47149274 -1.70208241
 -0.62531645  0.2976258  -1.394435    1.1436562  -1.54825871 -0.39458089
  0.52836137  1.22056806 -0.93296387 -0.85605201  0.14380209 -0.01002161
 -0.77914016  0.75909693 -0.70222831]

Preferred_Learning_Style ['Kinesthetic' 'Reading/Writing' 'Auditory' 'Visual']

Online_Courses_Completed [14 20 11  0 19  5 13 16  7 18  4 17  9  6  3  1 12 10
8  2 15]

Participation_in_Discussions ['Yes' 'No']

Assignment_Completion_Rate (%) [100  71  60  63  59  91  88  52  74  77  67  80
85  81  50  89  90  68
  96  86  79  76  55  53  57  58  64  98  65  78  94  99  62  72  83  75
  61  73  54  56  66  97  92  95  84  93  70  69  87  51  82]

Exam_Score (%) [-0.06736525 -1.71055842 -1.54057292 -0.01070342 -0.40733625
-0.91729275
 -1.48391109 -0.80396909  0.44259125 -0.86063092  0.15928208 -0.57732175
  0.04595841  1.12253325  1.29251875  1.51916608  0.21594391 -0.35067442

Attendance_Rate (%) [ 66  57  79  60  93  80  76  70 100  74  51  55  69  96  90
82  73  50
  86  91  81  85  71  67  62  58  88  54  92  65  84  63  77  53  89  75
  83  87  52  78  72  68  99  61  95  59  98  97  56  64  94]

Use_of_Educational_Tech ['Yes' 'No']

Self_Reported_Stress_Level ['High' 'Medium' 'Low']

Time_Spent_on_Social_Media (hours/week) [-0.65798898  1.44793045 -0.21463752
1.00457899  1.22625472  1.11541685
  1.66960617 -1.21217831 -1.10134044  0.0070382   1.33709258 -0.99050258
 -1.5446919   0.56122753  1.55876831 -0.32547539  0.89374112 -0.76882685
```

```
Sleep_Hours_per_Night [ 8  7 10  6  9  5  4]

Final_Grade ['C' 'D' 'B' 'A']

Study_Efficiency [-0.04196879 -7.75011469 -1.00808619 …  5.67542047
-0.59585899
 -1.63362218]

Social_Media_Impact [-0.40992946  6.56021267 -0.14044978 … -0.39484891
-0.47544082
 -1.93938011]
```

[89]:

```
from sklearn.preprocessing import LabelEncoder

categorical_cols = ["Gender", "Use_of_Educational_Tech",␣
 ↪"Self_Reported_Stress_Level" , "Final_Grade" ,␣
 ↪"Participation_in_Discussions" , "Preferred_Learning_Style" ]
encoder = LabelEncoder()

for i in categorical_cols:
    encoder.fit(df[i])
    df[i] = encoder.fit_transform(df[i])
    print(i , df[i].unique() ,' \n')
```

```
Gender [0 1 2]

Use_of_Educational_Tech [1 0]

Self_Reported_Stress_Level [0 2 1]

Final_Grade [2 3 1 0]

Participation_in_Discussions [1 0]

Preferred_Learning_Style [1 2 0 3]
```

5.2 Feature Extraction:

0.0.1 FEATURE EXTRACTION: Creating new meaningful features

```
[ ]:
```

```
[13]: df["Study_Efficiency"] = df["Exam_Score (%)"] / df["Study_Hours_per_Week"]
      df["Study_Efficiency"].head()
```

```
[13]: 0    1.437500
      1    1.333333
      2    0.914894
      3    5.384615
      4    2.625000
      Name: Study_Efficiency, dtype: float64
```

```
[14]: df["Social_Media_Impact"] = df["Time_Spent_on_Social_Media (hours/week)"] /␣
       ↪df["Study_Hours_per_Week"]
      df["Social_Media_Impact"].head()
```

```
[14]: 0    0.187500
      1    0.933333
      2    0.276596
      3    1.846154
      4    1.083333
      Name: Social_Media_Impact, dtype: float64
```

```
[15]: df["Attendance_Impact"] = (df["Attendance_Rate (%)"] / 100) * df["Exam_Score␣
       ↪(%)"]
      df["Attendance_Impact"].head()
```

```
[15]: 0    45.54
      1    22.80
      2    33.97
      3    42.00
      4    58.59
      Name: Attendance_Impact, dtype: float64
```

- Study Efficiency: Measures how effectively students convert study hours into exam scores. Higher values indicate better productivity.
- Social Media Impact: Shows the effect of social media usage on study time. A high value may indicate distractions.
- Attendance Impact: Highlights the influence of attendance on exam performance. Higher attendance often leads to better scores.

5.3 Feature Scaling :

1.1 Feature Scaling

[58]:
```
from sklearn.preprocessing import StandardScaler
```

[59]:
```
scaler = StandardScaler()
df[["Study_Hours_per_Week", "Exam_Score (%)", "Time_Spent_on_Social_Media␣
 ↪(hours/week)"]] = scaler.fit_transform(
    df[["Study_Hours_per_Week", "Exam_Score (%)", "Time_Spent_on_Social_Media␣
 ↪(hours/week)"]]
)
df.head(3)
```

[59]:
```
   Age  Gender  Study_Hours_per_Week Preferred_Learning_Style  \
0   18       0              1.605127              Kinesthetic
1   29       0              0.220714          Reading/Writing
2   20       0              1.528215              Kinesthetic

   Online_Courses_Completed Participation_in_Discussions  \
0                        14                          Yes
1                        20                           No
2                        11                           No

   Assignment_Completion_Rate (%)  Exam_Score (%)  Attendance_Rate (%)  \
0                             100       -0.067365                   66
1                              71       -1.710558                   57
2                              60       -1.540573                   79

   Use_of_Educational_Tech  Self_Reported_Stress_Level  \
0                        1                           2
1                        1                           1
2                        1                           0

   Time_Spent_on_Social_Media (hours/week)  Sleep_Hours_per_Night  \
0                                -0.657989                      8
1                                 1.447930                      8
2                                -0.214638                      7

   Final_Grade  Study_Efficiency  Social_Media_Impact  Attendance_Impact
0            2          1.437500             0.187500              45.54
1            1          1.333333             0.933333              22.80
2            1          0.914894             0.276596              33.97
```

After applying Standard Scaling, the numerical features like Study Hours per Week, Exam Score (%), and Time Spent on Social Media have been transformed to a standardized range. This ensures that no single feature dominates due to its magnitude, making the dataset more suitable for machine learning models.

Scaling helps improve model performance by maintaining uniformity across different features. Now, comparisons between study hours, exam scores, and social media usage are more balanced, allowing the model to learn more effectively without bias.

6. Insights

1. Study Hours and Performance:

Students who dedicate more study hours per week tend to achieve higher exam scores, indicating that consistent study habits positively impact academic success.

2. Social Media Usage Impact:

Higher social media usage relative to study time negatively affects exam scores, suggesting that excessive screen time may be a distraction from academic activities.

3. Attendance and Exam Performance:

Students with higher attendance rates tend to score better in exams, reinforcing the importance of class participation and regular engagement in learning activities.

4. Stress Levels and Academic Outcomes:

Students with high self-reported stress levels generally score lower, indicating that stress management plays a crucial role in academic performance.

5. Learning Style Preferences:

Visual and reading/writing learners tend to perform better in exams compared to kinesthetic learners, suggesting that different study techniques may yield varying results.

6. Use of Educational Technology:

Students who actively use educational technology, such as online courses and digital resources, show improved exam performance, highlighting the importance of tech-based learning tools.

7. Study Efficiency Variations:

Some students achieve high scores with fewer study hours, while others require extensive studying for moderate scores, indicating that study efficiency varies among individuals.

7. Conclusion

1. **Study Habits and Performance:**

- Students who study consistently for more hours per week tend to score higher in exams.
- Those with fewer study hours generally have lower scores, indicating the importance of dedicated study time.

2. **Social Media Influence on Academics:**

- Students who spend excessive time on social media relative to study hours tend to have lower exam scores.
- Managing screen time effectively can improve academic performance.

3. **Attendance and Learning Impact:**

- Higher attendance rates are linked to better exam scores, reinforcing the importance of regular class participation.
- Students with low attendance often struggle academically.

4. **Stress Levels and Academic Outcomes:**

- High-stress students tend to score lower, suggesting that stress negatively impacts learning and performance.
- Implementing stress management techniques can enhance student success.

5. **Learning Style Preferences and Performance:**

- Students with **visual** and **reading/writing** learning styles perform better on average compared to **kinesthetic** learners.
- Personalized teaching methods based on learning preferences can improve academic results.

6. **Use of Educational Technology:**

- Students utilizing online learning tools and educational resources tend to perform better.
- Encouraging the use of digital learning platforms may help boost academic success.

7. **Key Factors Affecting Performance:**

- **Attendance, Study Hours, Social Media Usage, and Stress Levels** are the most significant factors influencing student success.
- Focusing on these areas can help improve student performance and learning outcomes.