

*Name* : Harmanpreet Singh

*Register No.* : 22MCS0042

*Course* : Data Analytics - Lab

*Course Code* : MCSE615L

*Faculty* : Dr. Prakash M

*Assignment* : Exploratory Data Analysis and Insights Generation from [*Student Performance*](vhttps://archive.ics.uci.edu/ml/datasets/Student+Performance) Dataset

*Repository*

<https://github.com/harmanpreet-s/data-analytics-student-performance>

Introduction

The Student Performance Analysis and Predictive Modeling project aims to analyze and model student performance based on various attributes. The project utilizes two datasets, namely the Math and Portuguese datasets, which contain information about students' demographics, family background, study habits, and grades.

Objective

The objective of the assignment is to analyze student performance based on the given datasets and build predictive models to forecast student performance using machine learning algorithms. The assignment also involves feature selection to identify the most relevant features that contribute to student performance. Additionally, clustering analysis is performed to group students based on similar characteristics. The final objective is to evaluate and validate the performance of the models or analysis techniques using appropriate evaluation metrics.

Dataset Overview

There are two datasets: one for the Math course (student-mat.csv) and another for the Portuguese language course (student-por.csv). The dataset contains various attributes about the students, such as their personal information, family background, study habits, social activities, and academic performance.

The dataset also includes three columns (G1, G2, G3) that represent the students' grades for the corresponding course. G1 represents the first period grade, G2 represents the second period grade, and G3 represents the final grade.

Phase 1: Data Exploration

The project begins with data exploration to understand the structure and content of the datasets. The code provided loads the Math and Portuguese datasets using the pandas library and displays the first few rows of each dataset. Additionally, it prints the number of records in each dataset and checks for missing values.

*Importing initial required libraries and loading the datasets*

*# Importing the initial libraries*

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

*# Load the datasets*

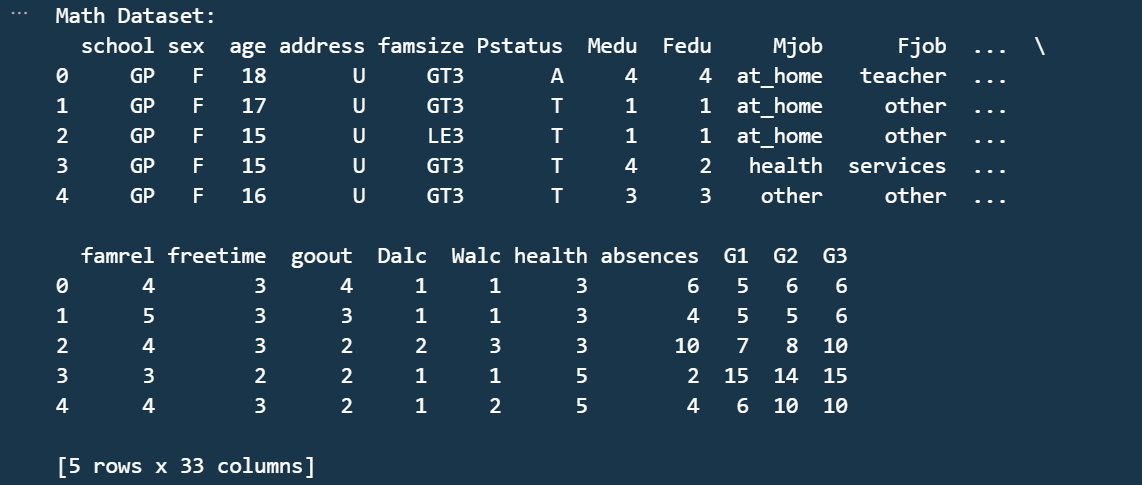
df\_math = pd.read\_csv('./dataset/student-mat.csv', sep=';')

df\_portuguese = pd.read\_csv('./dataset/student-por.csv', sep=';')

*Exploring the values in the Maths dataset*

print("Math Dataset:")

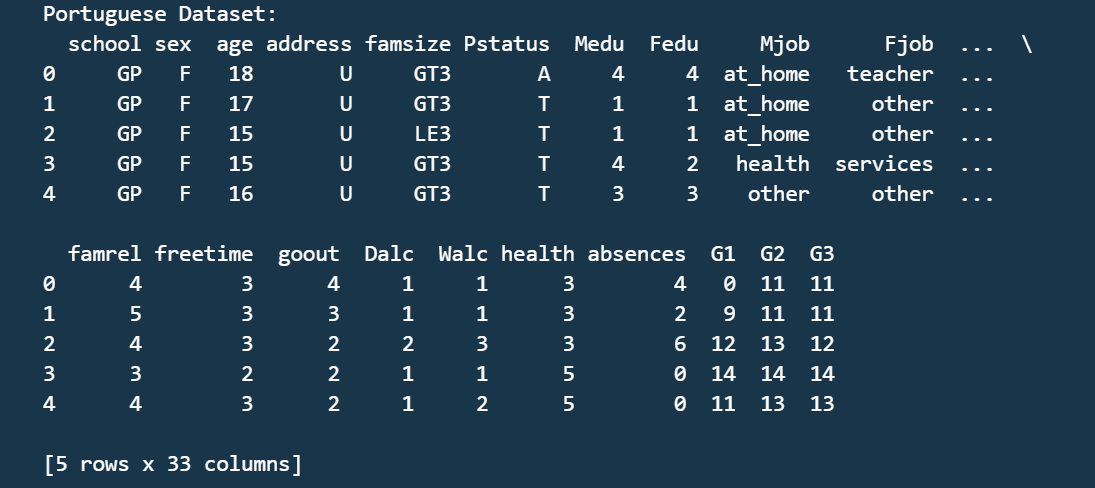
print(df\_math.head())  *# Display the first few rows of the math dataset*



*Exploring the values in the Portuguese dataset*

print("Portuguese Dataset:")

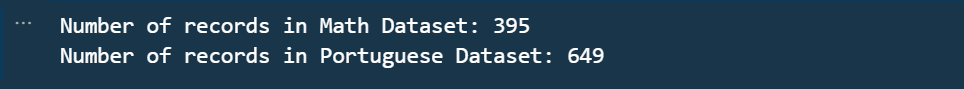
print(df\_portuguese.head())  *# Display the first few rows of the Portuguese dataset*



*Checking the number of records*

print("Number of records in Math Dataset:", len(df\_math))

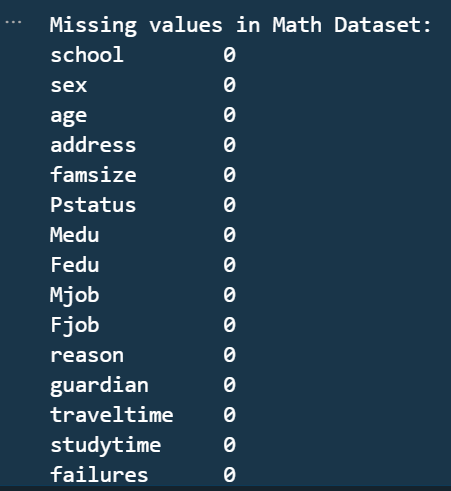
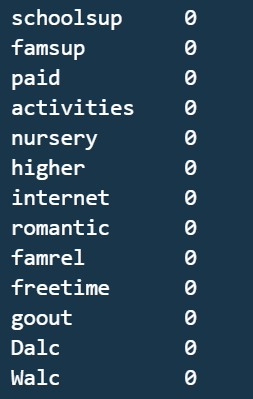
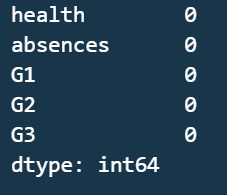
print("Number of records in Portuguese Dataset:", len(df\_portuguese))



*# Check for missing values in the math dataset*

print("Missing values in Math Dataset:")

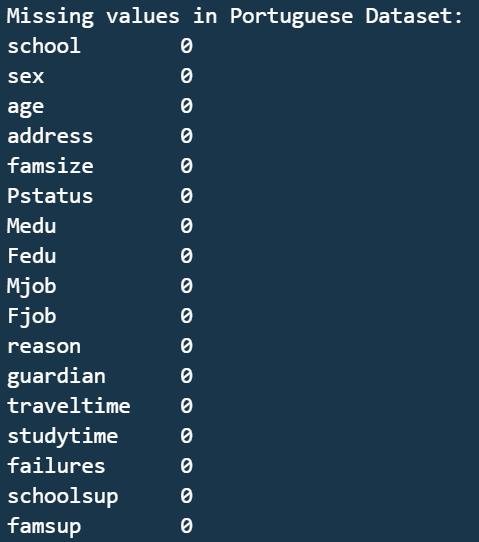
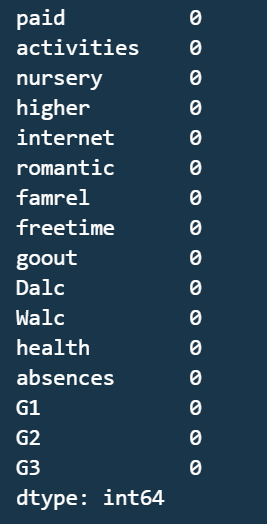
print(df\_math.isnull().sum())

*# Check for missing values in the Portuguese dataset*

print("Missing values in Portuguese Dataset:")

print(df\_portuguese.isnull().sum())

There are no missing values found in the dataset.

*Checking for Outliers*

*# Check for outliers in numerical variables*

numerical\_vars = ['age', 'absences', 'G1', 'G2', 'G3']

for var in numerical\_vars:

*# Box plot*

    df\_math.boxplot(column=var)

    plt.title('22MCS0042 | Box Plot for ' + var + ' in Math Dataset')

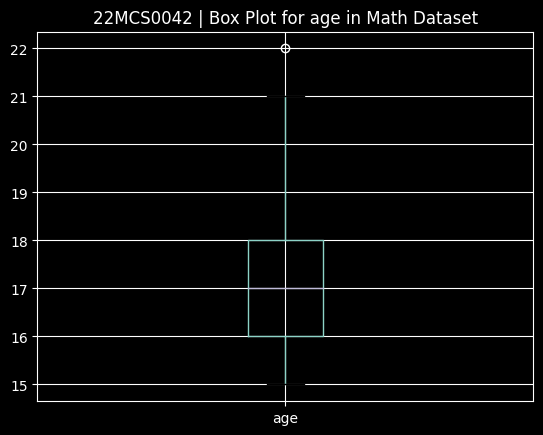
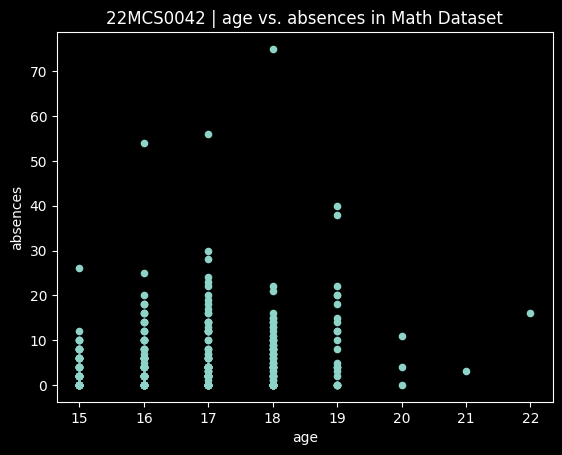
    plt.show()

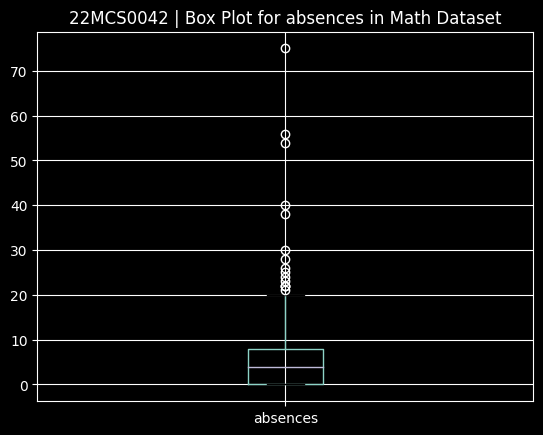
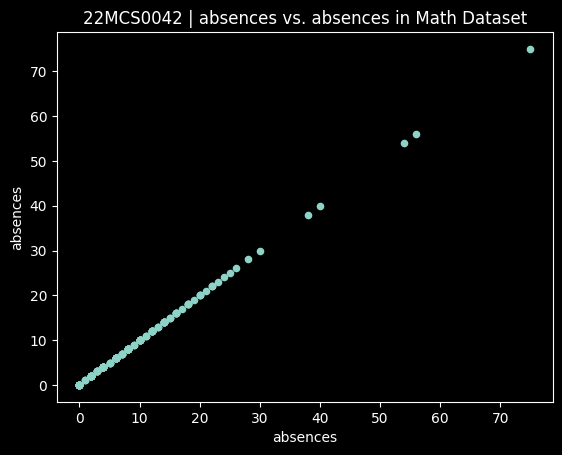
*# Scatter plot (against another numerical variable)*

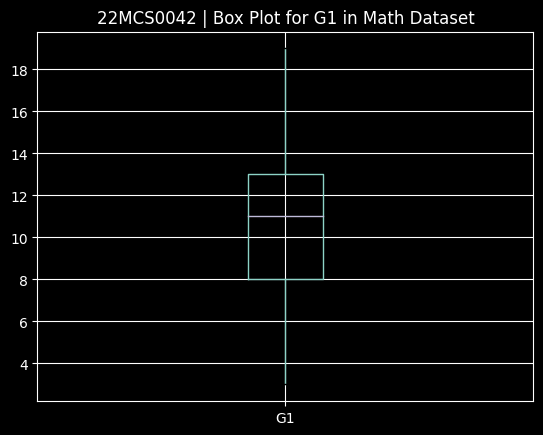
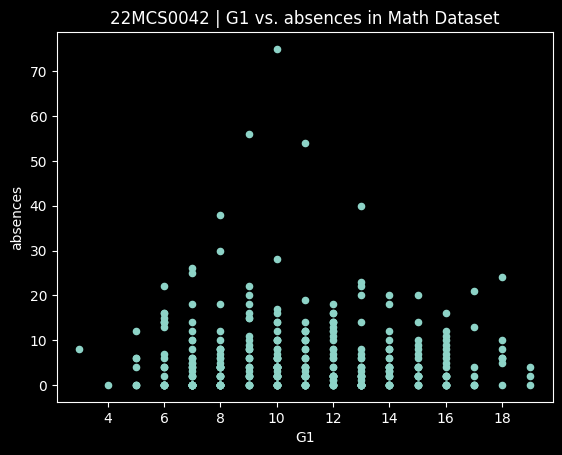
    df\_math.plot.scatter(x=var, y='absences')

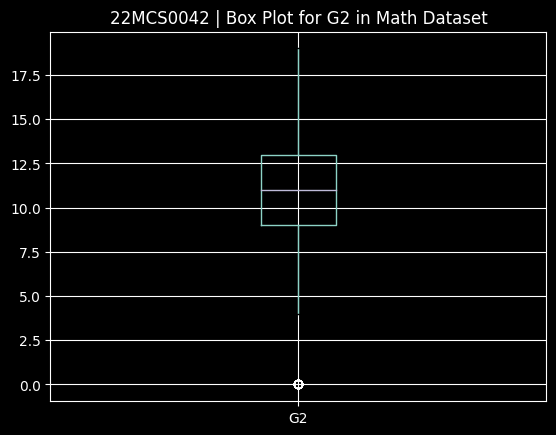
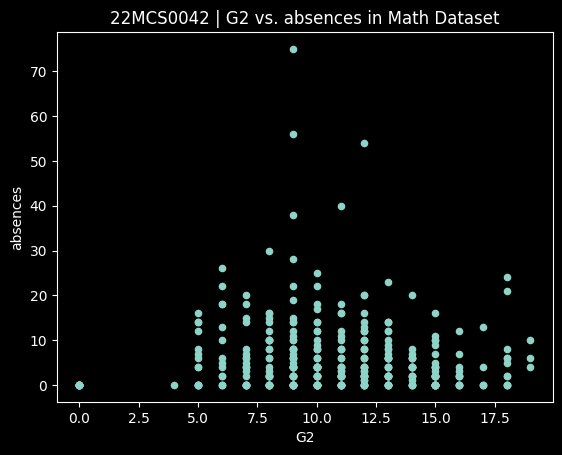
    plt.title('22MCS0042 | ' + var + ' vs. absences in Math Dataset')

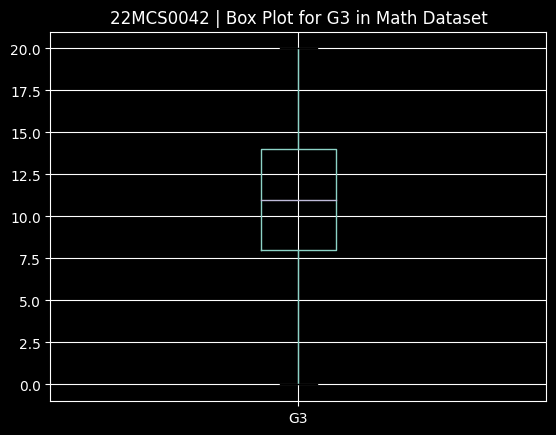
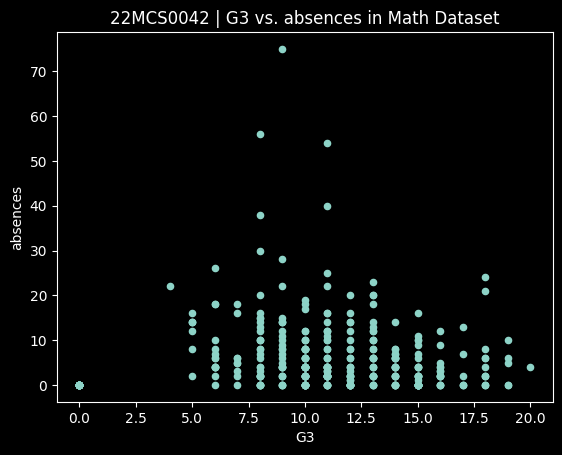
    plt.show()

*# Check cardinality in categorical variables*

categorical\_vars = ['school', 'sex', 'address', 'famsize', 'Pstatus', 'Mjob', 'Fjob', 'reason', 'guardian',

                    'schoolsup', 'famsup', 'paid', 'activities', 'nursery', 'higher', 'internet', 'romantic']

for var in categorical\_vars:

*# Unique values*

    unique\_values = df\_math[var].nunique()

    print("Unique values in", var, "in Math Dataset:", unique\_values)

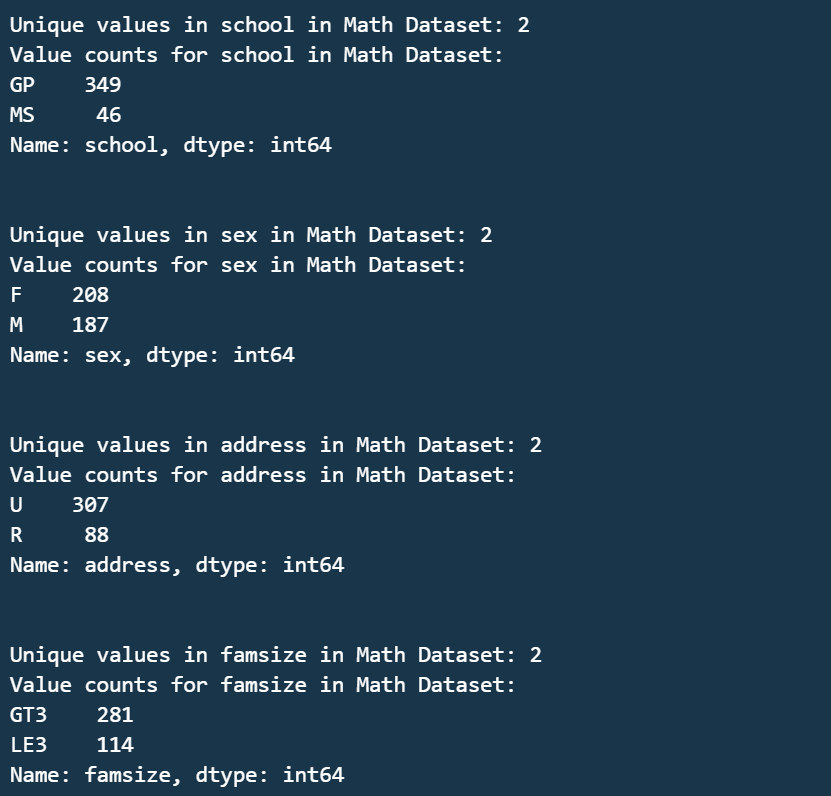
*# Value counts*

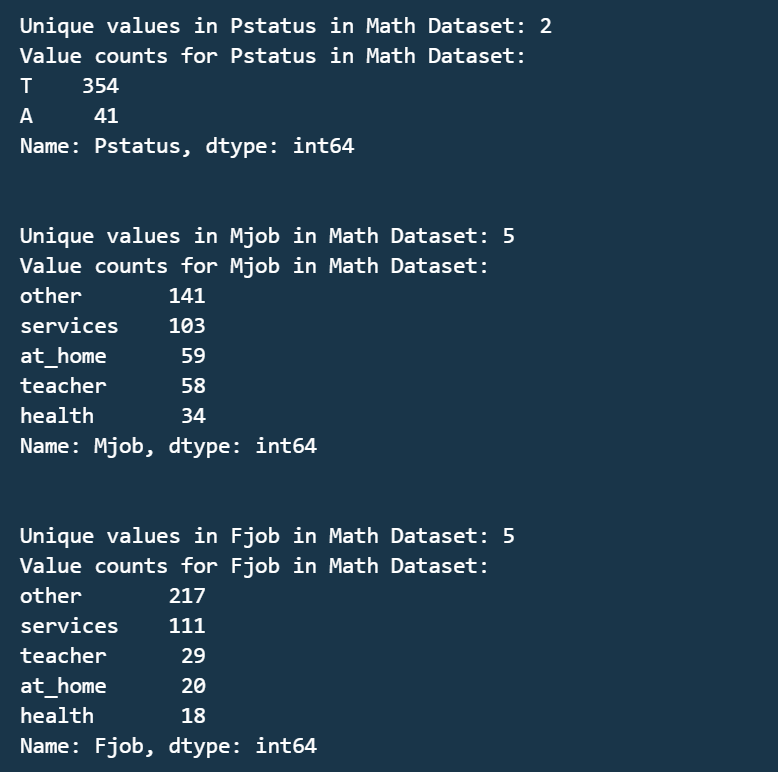
    value\_counts = df\_math[var].value\_counts()

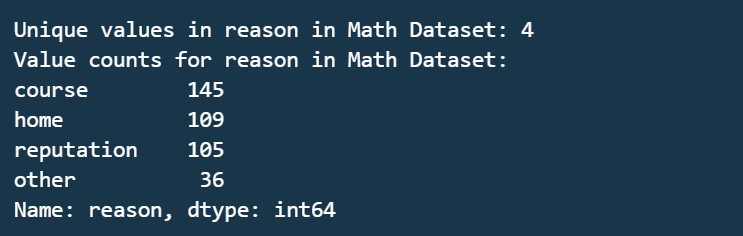
    print("Value counts for", var, "in Math Dataset:")

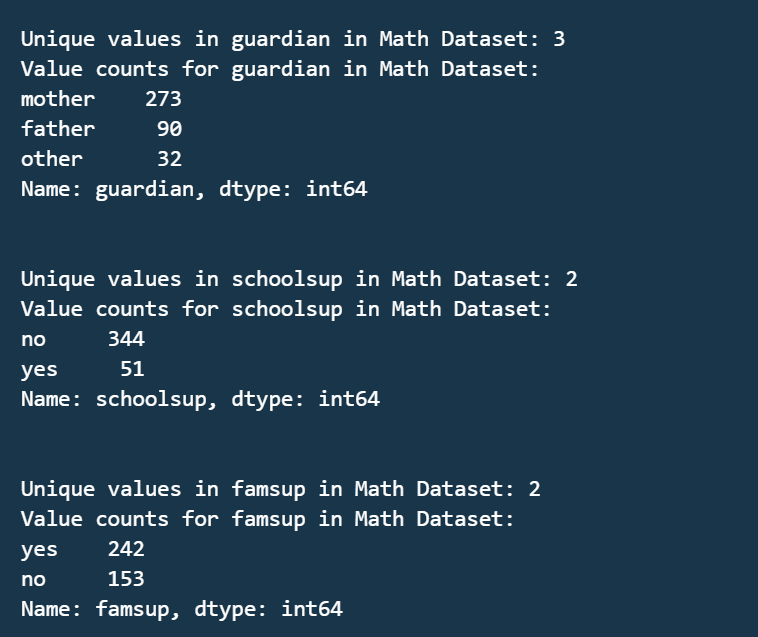
    print(value\_counts)

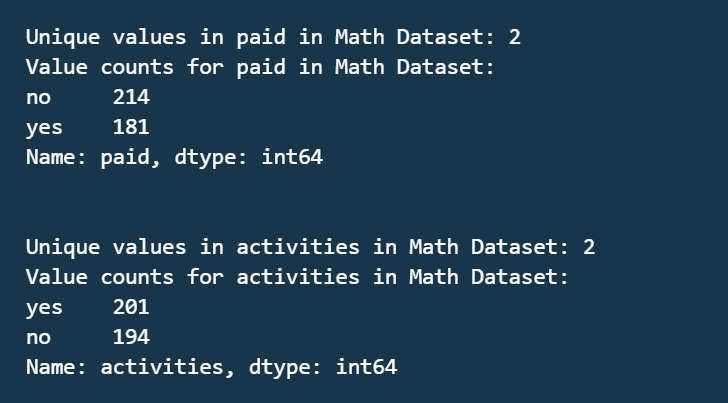
    print("\n")

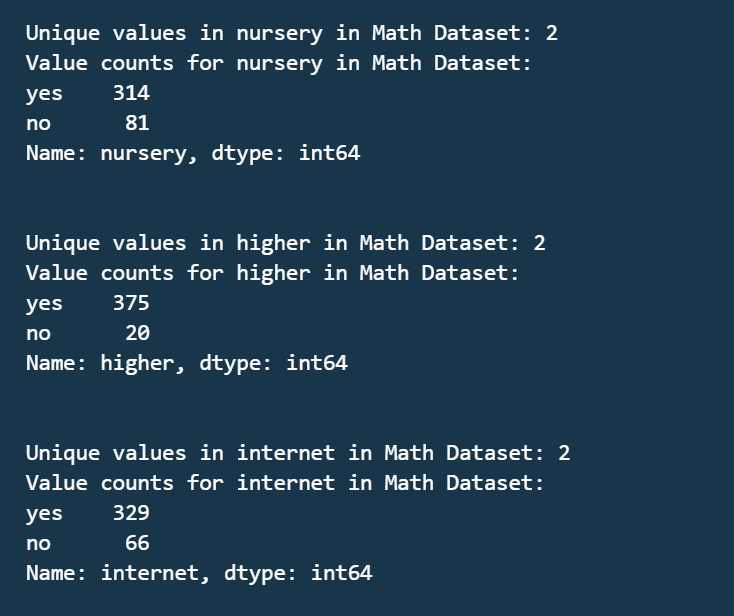


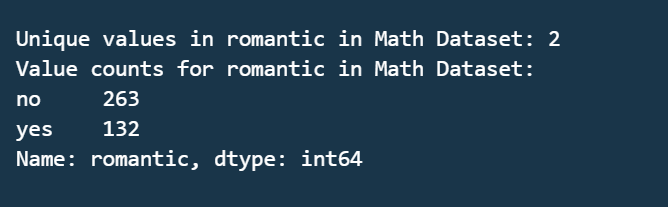












Phase 2: Data Integration

To perform comprehensive analysis, the Math and Portuguese datasets are merged based on common attributes using the merge function from pandas. The resulting merged dataset, named merged\_df, combines the information from both datasets into a single dataset.

*# Merge the two datasets based on common attributes*

merged\_df = pd.merge(df\_math, df\_portuguese, on=['school', 'sex', 'age', 'address', 'famsize', 'Pstatus',

                                                 'Medu', 'Fedu', 'Mjob', 'Fjob', 'reason', 'guardian',

                                                 'traveltime', 'studytime', 'failures', 'schoolsup',

                                                 'famsup', 'paid', 'activities', 'nursery', 'higher',

                                                 'internet', 'romantic', 'famrel', 'freetime', 'goout',

                                                 'Dalc', 'Walc', 'health', 'absences'], suffixes=('\_math', '\_portuguese'))

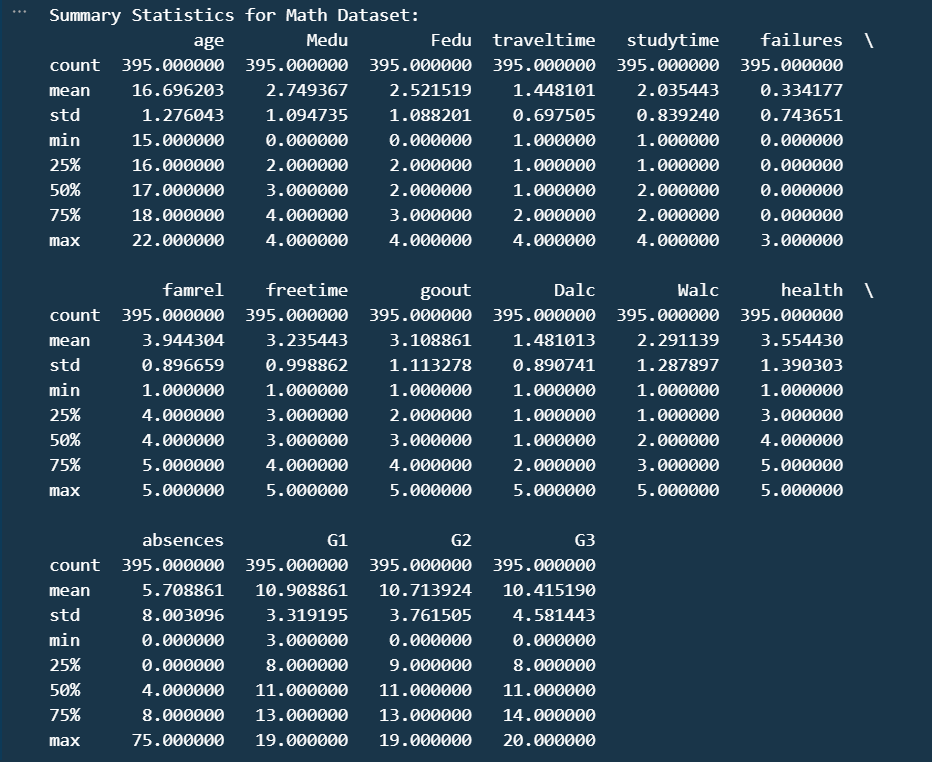
Phase 3: Descriptive Statistics

Descriptive statistics provide a summary of the datasets, including measures of central tendency and variability. The code provided calculates and displays summary statistics for both the Math and Portuguese datasets using the describe() function.

*# Display summary statistics for the math dataset*

print("Summary Statistics for Math Dataset:")

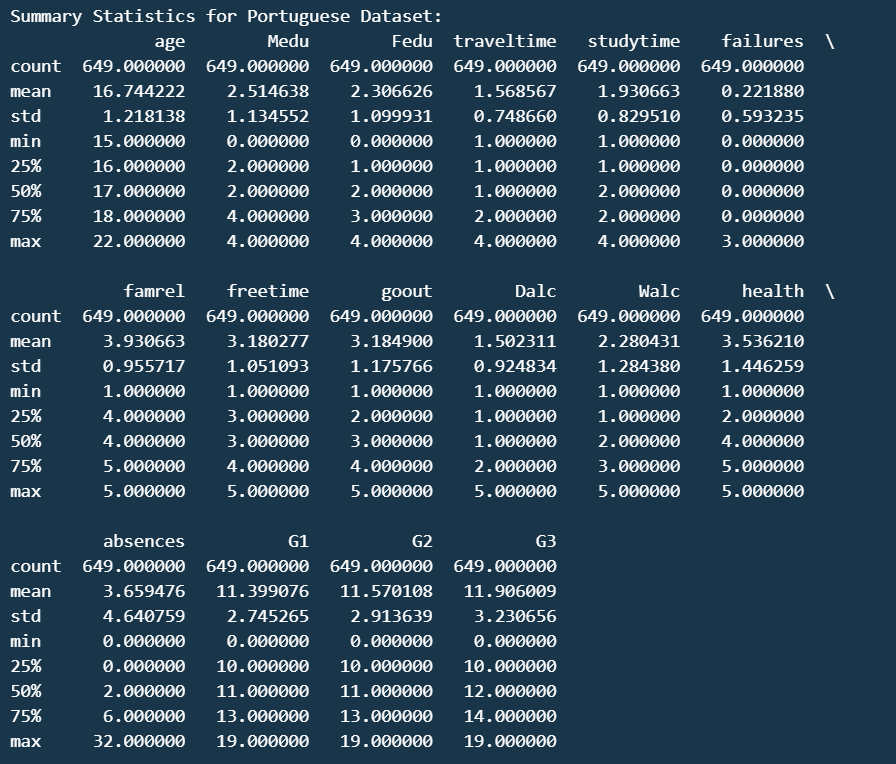
print(df\_math.describe())



*# Display summary statistics for the Portuguese dataset*

print("Summary Statistics for Portuguese Dataset:")

print(df\_portuguese.describe())



Phase 4: Data Visualization

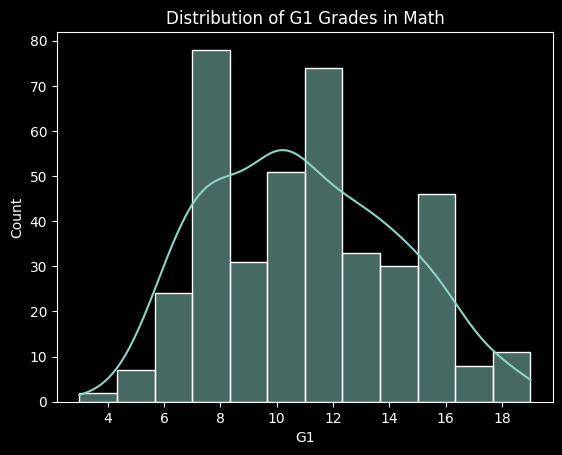
Data visualization is essential for gaining insights and understanding the distribution and relationships within the datasets. The code provided includes examples of data visualization using seaborn and matplotlib libraries. It generates histograms to visualize the distribution of G1 grades in Math and a box plot to explore the relationship between sex and the final grade (G3) in Portuguese.

*# Plot to visualize the distribution of the 'G1' grades in the Math dataset*

sns.histplot(df\_math['G1'], kde=True)

plt.title("22MCS0042 | Distribution of G1 Grades in Math")

plt.show()



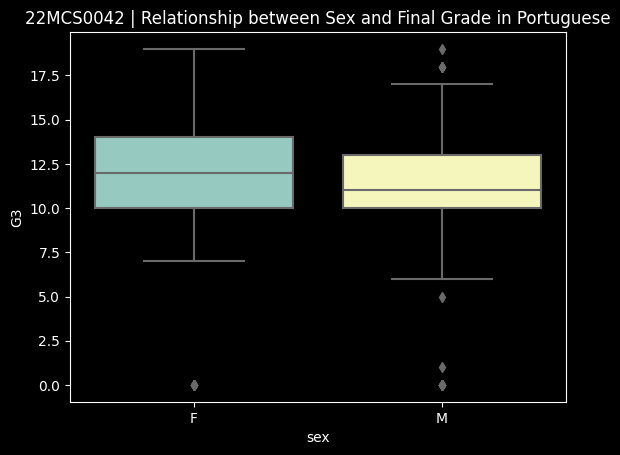
The histogram displays the frequency or count of different grade ranges on the x-axis and the corresponding number of students on the y-axis. The shape of the histogram provides insights into the distribution pattern of the 'G1' grades.

*# Plot relationship between Sex and Final Grade in Portuguese*

sns.boxplot(x='sex', y='G3', data=df\_portuguese)

plt.title("22MCS0042 | Relationship between Sex and Final Grade in Portuguese")

plt.show()



We generated a boxplot to explore the relationship between sex and final grades in the Portuguese dataset, and a histogram to analyze the distribution of G1 grades for a specific student in the Math dataset. These visualizations aid in understanding the grade distributions and patterns within the respective datasets.

*# Correlation Analysis*

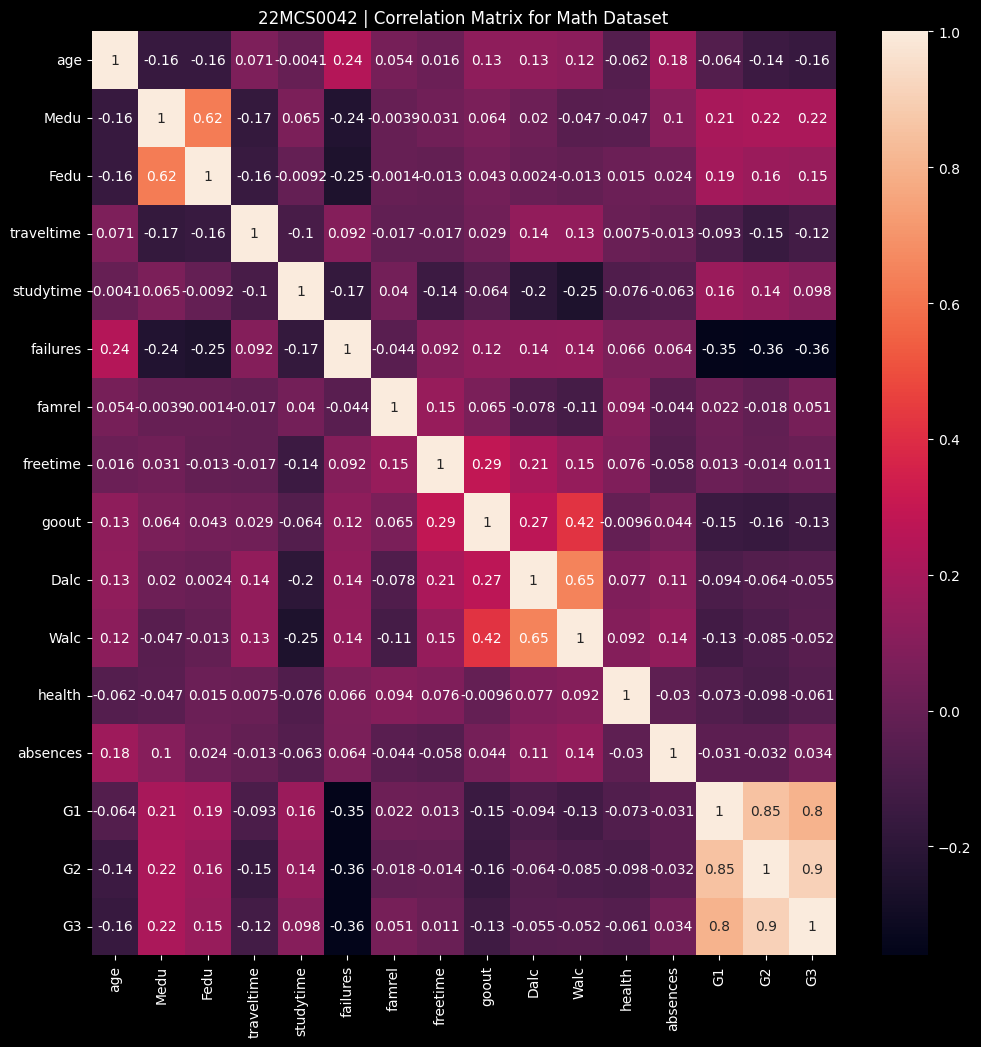
fig, ax = plt.subplots(figsize=(12, 12))

correlation\_matrix = df\_math.corr()

sns.heatmap(correlation\_matrix, annot=True)

plt.title("22MCS0042 | Correlation Matrix for Math Dataset")

plt.show()



A correlation analysis on the Math dataset and generates a heatmap visualization to depict the correlation matrix. The heatmap provides valuable insights into the relationships between different variables, allowing for the identification of potentially significant associations within the dataset.

*Preparing for further work*

*Importing the required libraries to work with the Feature Selection, Modeling, Analysis and Evaluation*

*# Import required libraries*

from sklearn.model\_selection import train\_test\_split

from sklearn.linear\_model import LinearRegression

from sklearn.feature\_selection import SelectKBest, f\_regression

from sklearn.cluster import KMeans

from sklearn.metrics import silhouette\_score

from sklearn.metrics import mean\_squared\_error, r2\_score

*Loading the maths dataset again to primarily work on it.*

*# Load the Maths dataset*

df = pd.read\_csv('./dataset/student-mat.csv', sep=';')

*Selecting the relevant feature and target variable for the further work*

*# Select relevant features and target variable*

features = ['age', 'Medu', 'Fedu', 'studytime', 'failures', 'absences']

target = 'G3'

X = df[features]

y = df[target]

*Splitting the dataset into Training and Testing sets, taking 80% training data with random\_state = 42*

*# 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)

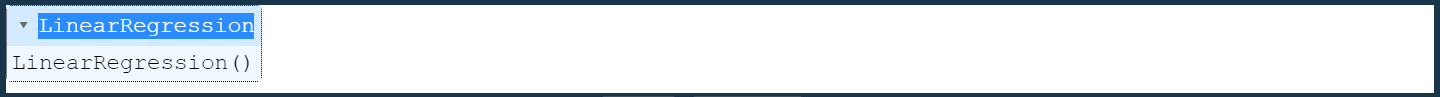
Phase 5: Predictive Modeling

Predictive modeling aims to forecast student performance using machine learning algorithms. The code provided demonstrates the initial steps for building predictive models. We are using LinearRegression for our modelling.

*# Predictive Modeling - Linear Regression*

model = LinearRegression()

model.fit(X\_train, y\_train)



Phase 6: Feature Selection

Feature selection involves identifying the most relevant features that contribute to student performance. Techniques like feature importance and correlation analysis can be applied to determine the significant features.

*# Feature Selection - SelectKBest*

selector = SelectKBest(score\_func=f\_regression, k=3)

X\_train\_selected = selector.fit\_transform(X\_train, y\_train)

selected\_features = [features[i] for i in selector.get\_support(indices=True)]

print(selected\_features)



Phase 7: Clustering Analysis

Clustering analysis groups students based on similar characteristics, allowing for the identification of distinct student profiles or segments.

*# Clustering Analysis - KMeans*

kmeans = KMeans(n\_clusters=3, random\_state=42)

clusters = kmeans.fit\_predict(X\_train)

Phase 8: Evaluation and Validation

Evaluation and validation assess the performance and reliability of the models or analysis techniques used in the project.

*# Evaluate the predictive model*

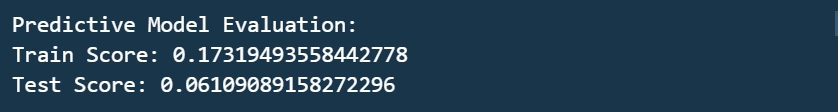
train\_score = model.score(X\_train, y\_train)

test\_score = model.score(X\_test, y\_test)

print("Predictive Model Evaluation:")

print("Train Score:", train\_score)

print("Test Score:", test\_score)



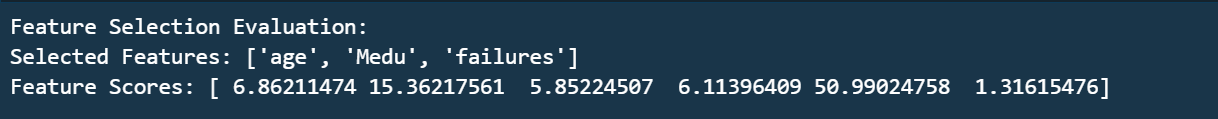
*# Evaluate the feature selection*

feature\_scores = selector.scores\_

print("Feature Selection Evaluation:")

print("Selected Features:", selected\_features)

print("Feature Scores:", feature\_scores)

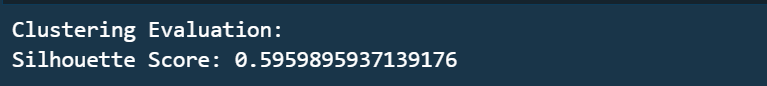


*# Evaluate the clustering*

silhouette\_avg = silhouette\_score(X\_train, clusters)

print("Clustering Evaluation:")

print("Silhouette Score:", silhouette\_avg)



*# Generate predictions*

y\_pred = model.predict(X\_test)

*# Calculate accuracy score*

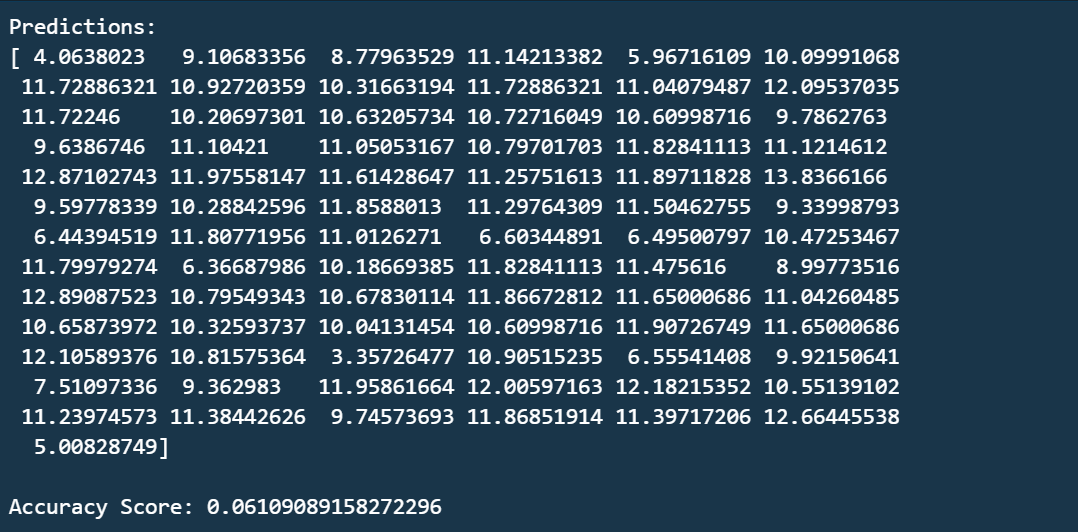
accuracy = model.score(X\_test, y\_test)

*# Print predictions and accuracy score*

print("Predictions:")

print(y\_pred)

print("\nAccuracy Score:", accuracy)



*# Calculate mean squared error (MSE)*

mse = mean\_squared\_error(y\_test, y\_pred)

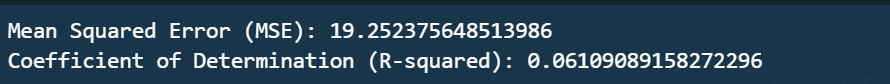
*# Calculate coefficient of determination (R-squared)*

r2 = r2\_score(y\_test, y\_pred)

*# Print evaluation metrics*

print("Mean Squared Error (MSE):", mse)

print("Coefficient of Determination (R-squared):", r2)



*Conclusion*

The Student Performance Analysis and Predictive Modeling project provides insights into student performance based on the Math and Portuguese datasets. It includes data exploration, data cleaning, data integration, descriptive statistics, data visualization, and initial steps for predictive modeling. Further analysis, such as feature selection, clustering analysis, and evaluation/validation, can be performed to gain deeper insights into student performance factors and develop accurate predictive models.