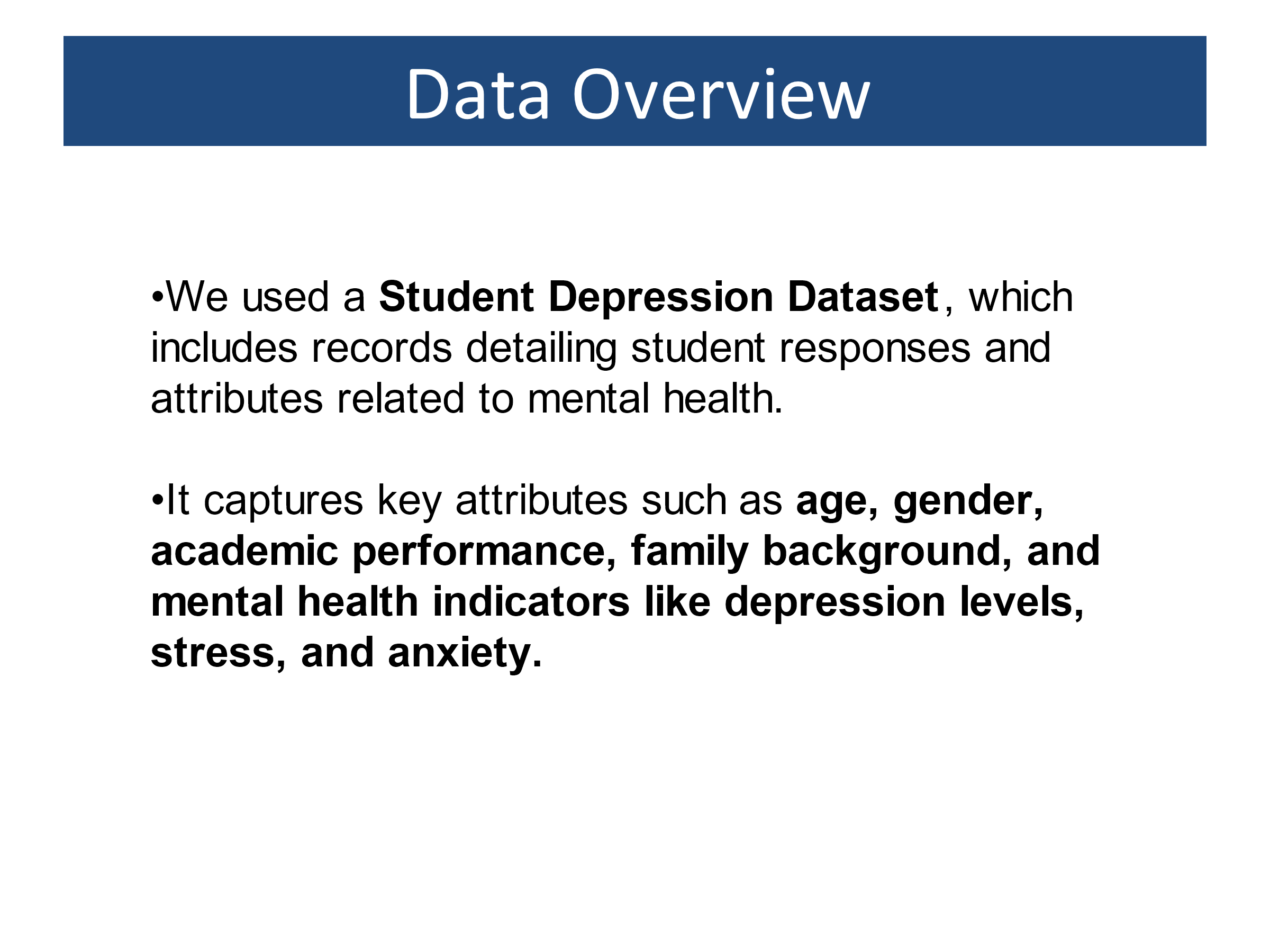
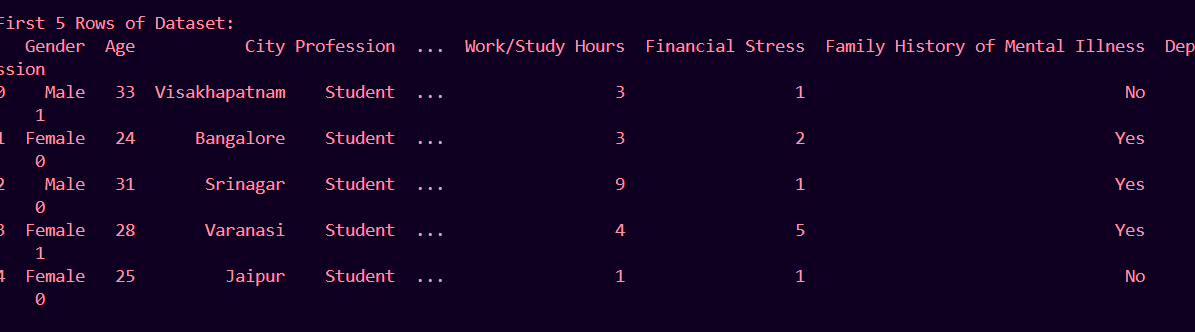
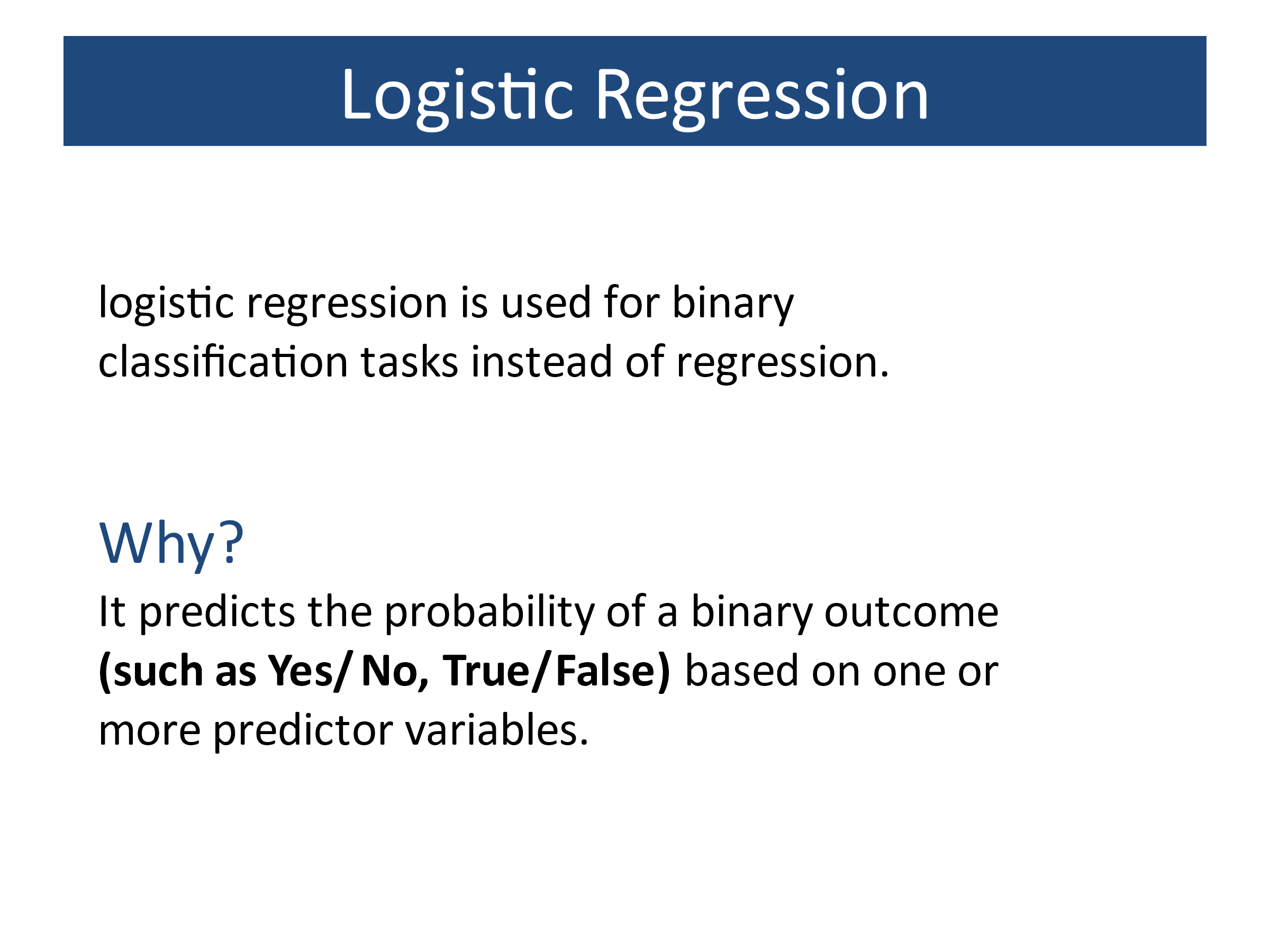
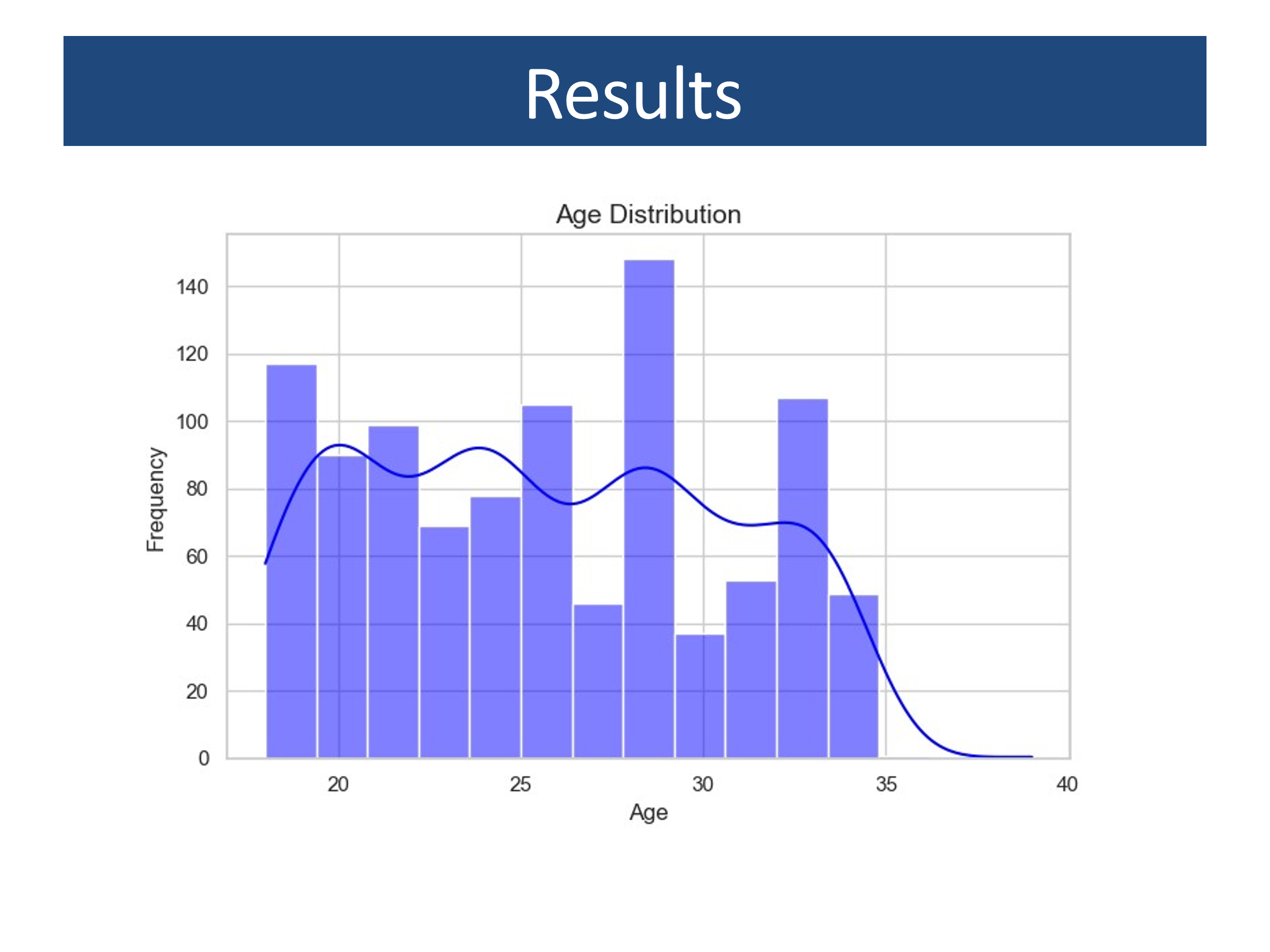
**Project Report**

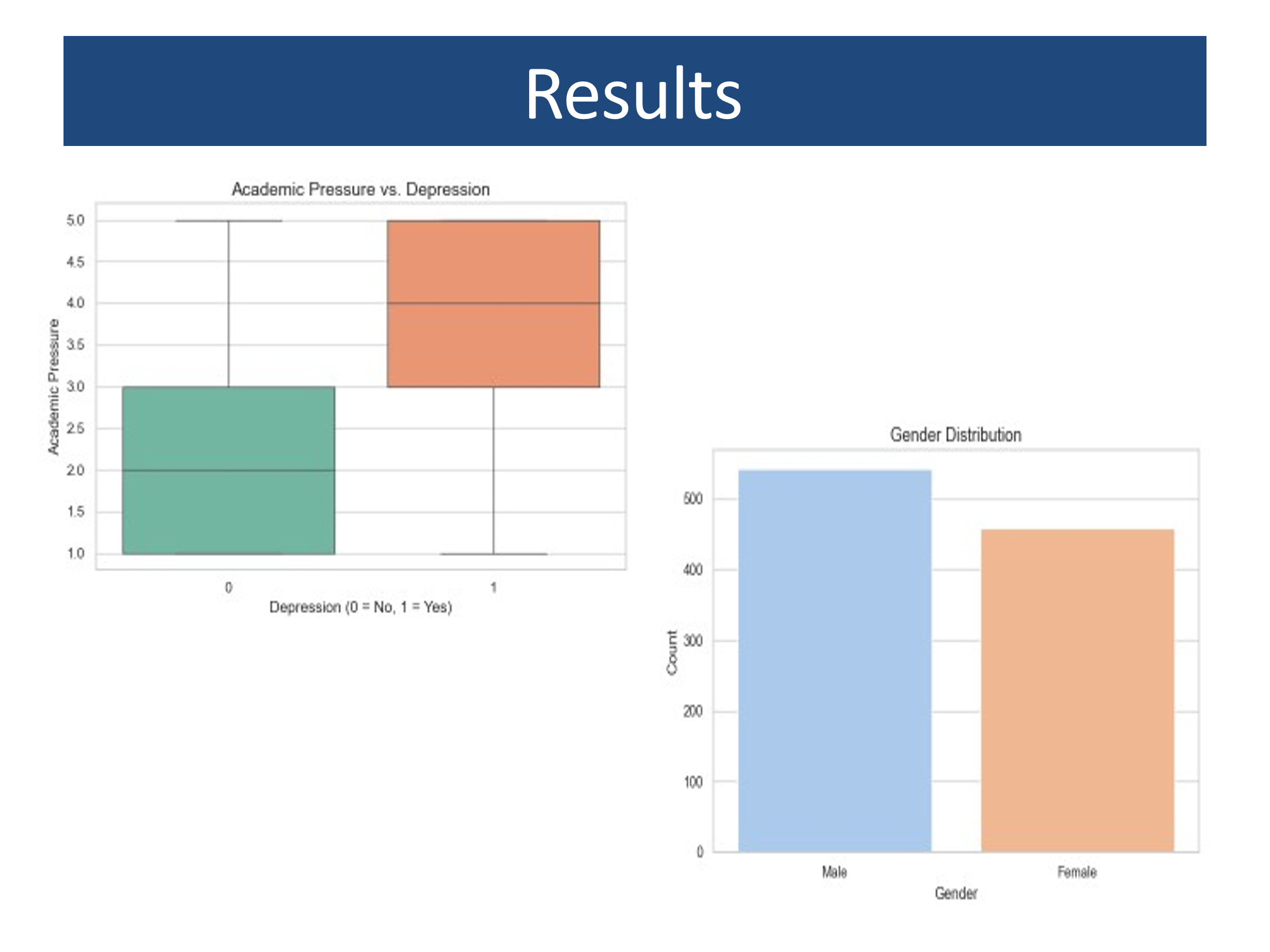
**(Student Depression Analysis)**

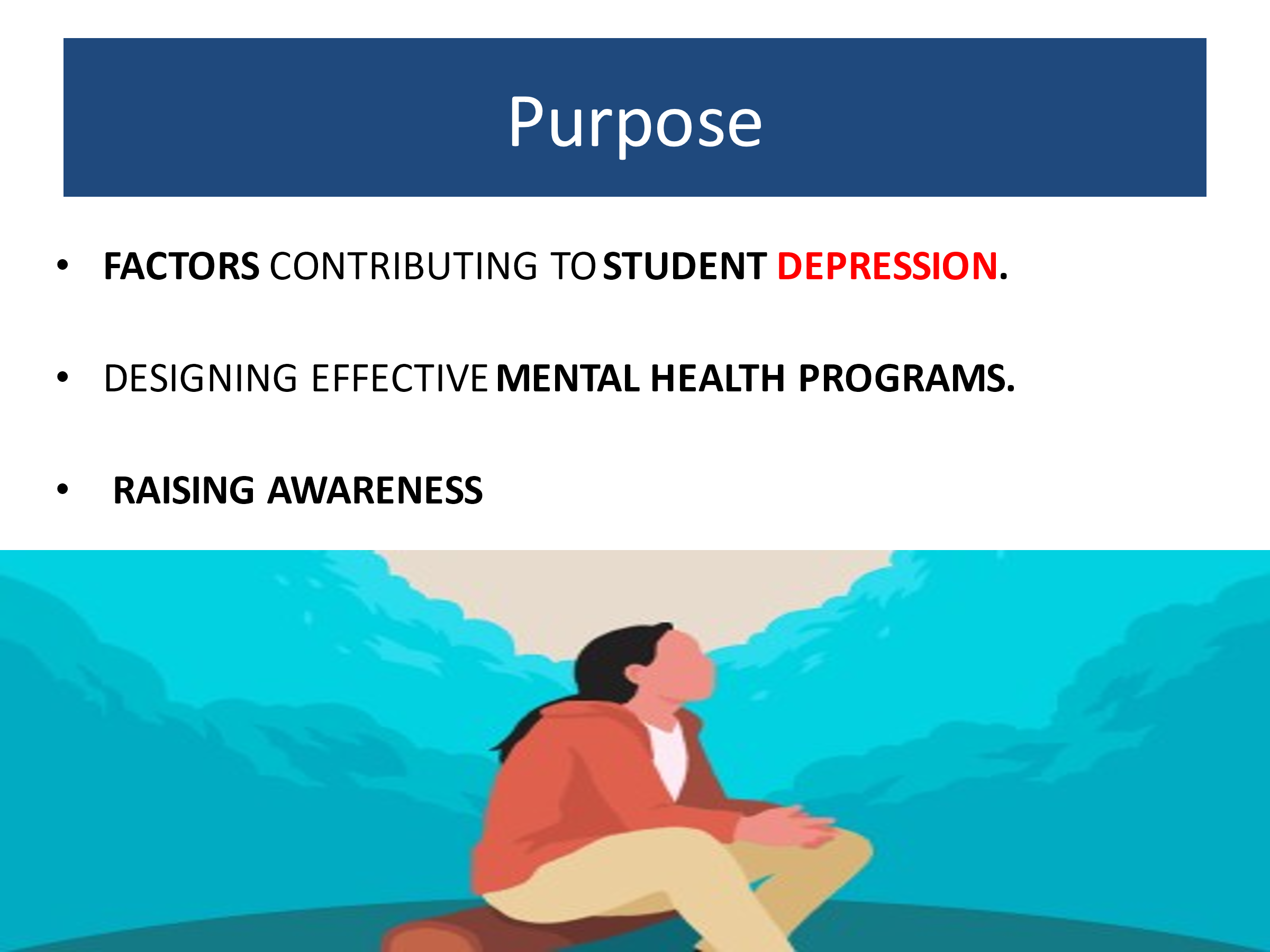












**Code:**

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

import warnings

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

from sklearn.linear\_model import LogisticRegression

from sklearn.metrics import classification\_report, confusion\_matrix

*# Suppress FutureWarnings*

warnings.simplefilter(action='ignore', category=FutureWarning)

*# Load the dataset*

file\_path = r"C:\Users\Khan\OneDrive - Iqra University Islamabad Chak Shahzad\software engineering\5th Semester\Data Science\presentation\Student Depression Dataset.xlsx"

dataset = pd.read\_excel(file\_path)

*# Display dataset information*

print("Dataset Info:")

print(dataset.info())

print("\nFirst 5 Rows of Dataset:")

print(dataset.head())

*# Descriptive statistics for numerical columns*

print("\nDescriptive Statistics:")

print(dataset.describe())

*# Unique values in categorical columns*

categorical\_columns = dataset.select\_dtypes(include='object').columns

unique\_values = {col: dataset[col].nunique() for col in categorical\_columns}

print("\nUnique Values in Categorical Columns:")

print(unique\_values)

*# Set style for plots*

sns.set(style="whitegrid")

*# Plot 1: Distribution of Age*

plt.figure(figsize=(8, 5))

sns.histplot(dataset['Age'], kde=True, bins=15, color="blue")

plt.title("Age Distribution", fontsize=14)

plt.xlabel("Age")

plt.ylabel("Frequency")

plt.show()

*# Plot 2: Depression vs. Academic Pressure*

plt.figure(figsize=(8, 5))

sns.boxplot(x="Depression", y="Academic Pressure", data=dataset, palette="Set2", hue=None)

plt.title("Academic Pressure vs. Depression", fontsize=14)

plt.xlabel("Depression (0 = No, 1 = Yes)")

plt.ylabel("Academic Pressure")

plt.show()

*# Plot 3: Gender Distribution*

plt.figure(figsize=(8, 5))

sns.countplot(x="Gender", data=dataset, palette="pastel", hue=None)

plt.title("Gender Distribution", fontsize=14)

plt.xlabel("Gender")

plt.ylabel("Count")

plt.show()

*# Predictive Analysis: Logistic Regression*

*# Selecting features and target for predictive analysis*

features = dataset[['Academic Pressure', 'Financial Stress', 'Study Satisfaction']]

target = dataset['Depression']

*# Splitting data into training and testing sets*

X\_train, X\_test, y\_train, y\_test = train\_test\_split(features, target, test\_size=0.3, random\_state=42)

*# Logistic Regression model*

logistic\_model = LogisticRegression(max\_iter=1000)

logistic\_model.fit(X\_train, y\_train)

*# Predictions and evaluation*

y\_pred = logistic\_model.predict(X\_test)

conf\_matrix = confusion\_matrix(y\_test, y\_pred)

classification\_rep = classification\_report(y\_test, y\_pred)

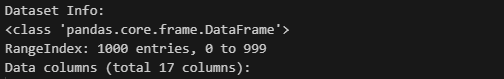
print("\nConfusion Matrix:")

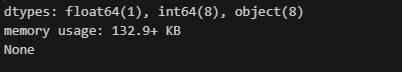
print(conf\_matrix)

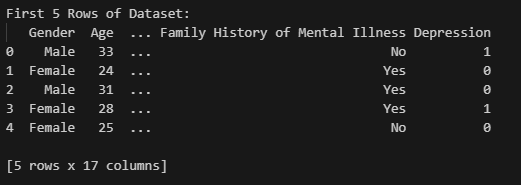
print("\nClassification Report:")

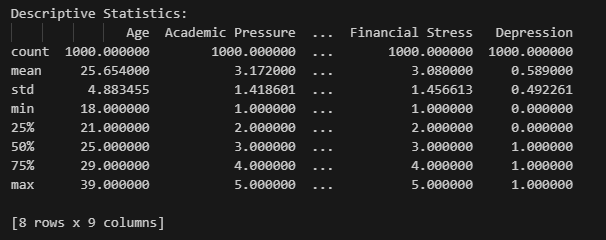
print(classification\_rep)

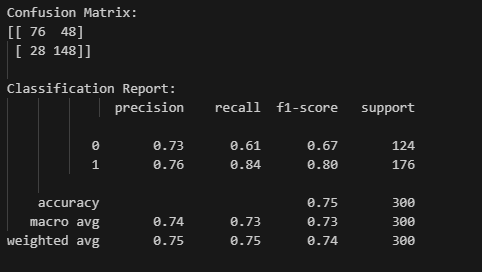
**Code Output:**

****









**Code Explanation :**

**1. Importing Libraries:** The code begins with importing necessary libraries. Each of these libraries plays a specific role:

* pandas is utilized for data manipulation and analysis.
* matplotlib.pyplot and seaborn are used for data visualization.
* warnings helps in managing warnings generated by Python.
* sklearn.model\_selection, sklearn.linear\_model, and sklearn.metrics are essential for constructing machine learning models and evaluating their performance.

**2. Suppressing Warnings:** The warnings.simplefilter is set to ignore FutureWarning to make the output cleaner by suppressing these types of warnings.

**3. Loading Dataset:** The dataset is loaded from an Excel file into a pandas DataFrame using the pd.read\_excel function. The path where the file is saved is provided in file\_path.

**4. Displaying Dataset Information:**

* dataset.info() provides a concise summary of the dataset including column names, count of non-null values, and data types.
* dataset.head() shows the first five rows for a quick glance at the data.
* dataset.describe() generates descriptive statistics of numerical columns such as mean, median, standard deviation, etc.

**5. Unique Values in Categorical Columns:** This block identifies categorical columns in the dataframe and counts the unique values in each of these columns.

**6. Data Visualization:** Three plots are created to visualize different aspects of the dataset:

* **Distribution of Age:** A histogram with a KDE (Kernel Density Estimate) is plotted using seaborn to visualize the age distribution.
* **Depression vs. Academic Pressure:** A box plot shows the distribution of academic pressure among students with and without depression.
* **Gender Distribution:** A count plot visualizes the distribution of gender in the dataset.

**7. Predictive Analysis Using Logistic Regression:**

* **Selecting Features and Target:** Features include academic pressure, financial stress, and study satisfaction, while the target is depression.
* **Splitting the Data:** Using train\_test\_split to divide the dataset into training and testing sets (70% training and 30% testing).
* **Model Training:** A logistic regression model is trained using the training dataset.
* **Predictions and Evaluation:** The model predicts depression in the test dataset, and its performance is evaluated using a confusion matrix and classification report. The confusion matrix shows the count of true positive, true negative, false positive, and false negative predictions, while the classification report provides precision, recall, f1-score, and support for each class.