Sri Lanka Institute of Information Technology

IT2011 Artificial Intelligence and Machine Learning

24/09/2025

Year 2 Semester 1

2025-Y2-S1-MLB-B9G1-07

Student Depression Analys - Data Preprocessing and Analysis

IT24102351- R.M.Islah
IT24102326-Dadallage.S
IT24102380-Fernando K.S.N.A
IT24102325-Muditha Bimsara K.P
IT24102361- Sharanjan. S
IT24102357- Jayasinghe J.A.K.N

1. Introduction

Project Overview

Depression among students is a growing concern worldwide, often resulting from academic stress, social pressure, financial difficulties, and personal challenges. Early detection of depressive tendencies can play a vital role in providing timely intervention and support. Traditional methods, such as counseling sessions and surveys, are often limited by their subjective nature, time consumption, and lack of scalability.

To address these limitations, this project proposes the development of a **Student Depression Detection AI Model**. The system leverages Artificial Intelligence and Machine Learning techniques to analyze student-related data—such as responses to questionnaires, academic performance, attendance, behavioral patterns, and even textual or social media content—to identify indicators of depression. By detecting risk levels early, the model aims to provide a reliable decision-support tool for universities, counselors, and healthcare professionals to better monitor student mental health and take preventive action.

This model not only aids in identifying students at risk but also contributes toward reducing stigma by providing an automated, confidential, and unbiased assessment system. It is envisioned as part of a broader mental health support framework that enhances student well-being and academic success.

Objectives

1. **Data Collection and Integration**
    a. Gather relevant student data (survey responses, academic records, behavioral indicators, etc.) from different sources and combine them into a structured dataset.
2. **Data Cleaning**
    a. Handle missing values, duplicate entries, and irrelevant data to ensure a consistent and reliable dataset for modeling.
3. **Data Transformation**

    a. Convert raw data into usable formats (e.g., converting categorical data into numerical, scaling values, or encoding text responses).
4. **Feature Engineering**
    a. Extract and create meaningful features (such as average grades, attendance rate, sentiment from text answers) that may indicate signs of depression.
5. **Exploratory Data Analysis (EDA)**
    a. Use statistical summaries and visualization techniques to understand data distributions, correlations, and patterns in relation to depression levels.
6. **Outlier Detection and Handling**
    a. Identify extreme or abnormal values that could distort the model's accuracy and decide whether to remove or adjust them.
7. **Data Normalization/Standardization**
    a. Scale numerical features to ensure that all variables contribute equally to the model training.

2. Dataset Description

- Dataset Source: https://www.kaggle.com/datasets/adilshamim8/student-depression-dataset
- Number of Rows: 140,699 (Balanced 41.7% - 58.3%split)
- Number of Features: 18
- Purpose: Predict whether a student is depressed or not.

3. Prerequisites

To successfully complete this project, the following prerequisites are required:

Programming Language:
• Python

Development Environment:
• Google Colab

Required Libraries:
• pandas-for data manipulation and analysis
• numpy-for numerical operations and array handling
• matplotlib-for data visualization
• seaborn-for advanced statistical visualizations
• scikit-learn-for data preprocessing and machine learning utilities

4. Roles of the group members

Name	Responsible role
Jayasinghe J.A.K.N	Data Cleaning
Dadallage.S	Data Representation
Muditha Bimsara K.P	Feature Selection & Reduction
Fernando K.S.N.A	Feature Scaling
Sharanjan. S	Data Discretization
R.M.Islah	Feature Creation Data Imputation & Estimation

1. Data Cleaning

- Handle Missing Values
- Outliers

- Drop Irrelevant Columns

2. Data Representation

- Encoding (converting categorical → numeric)

3. Feature Selection & Reduction

- PCA (Dimensionality Reduction)
- Correlation & Multicollinearity Check (removing redundant features)

4. Feature Scaling

- Normalization / Scaling (Min-Max, Standardization)

5. Data Discretization

- Binning / Discretization (continuous → categorical bins)

6. Feature Creation

- Feature Engineering (creating new useful variables from raw data)

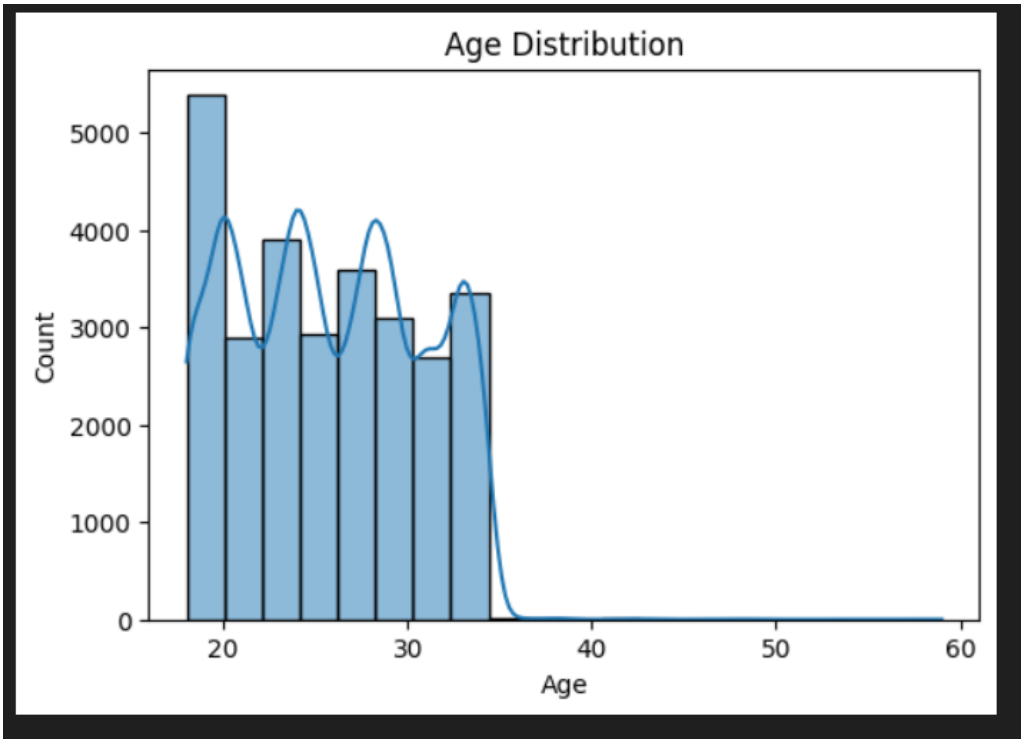
7. Data Imputation & Estimation

- Regression (used to estimate/fill missing values)

1. Handle Missing Values

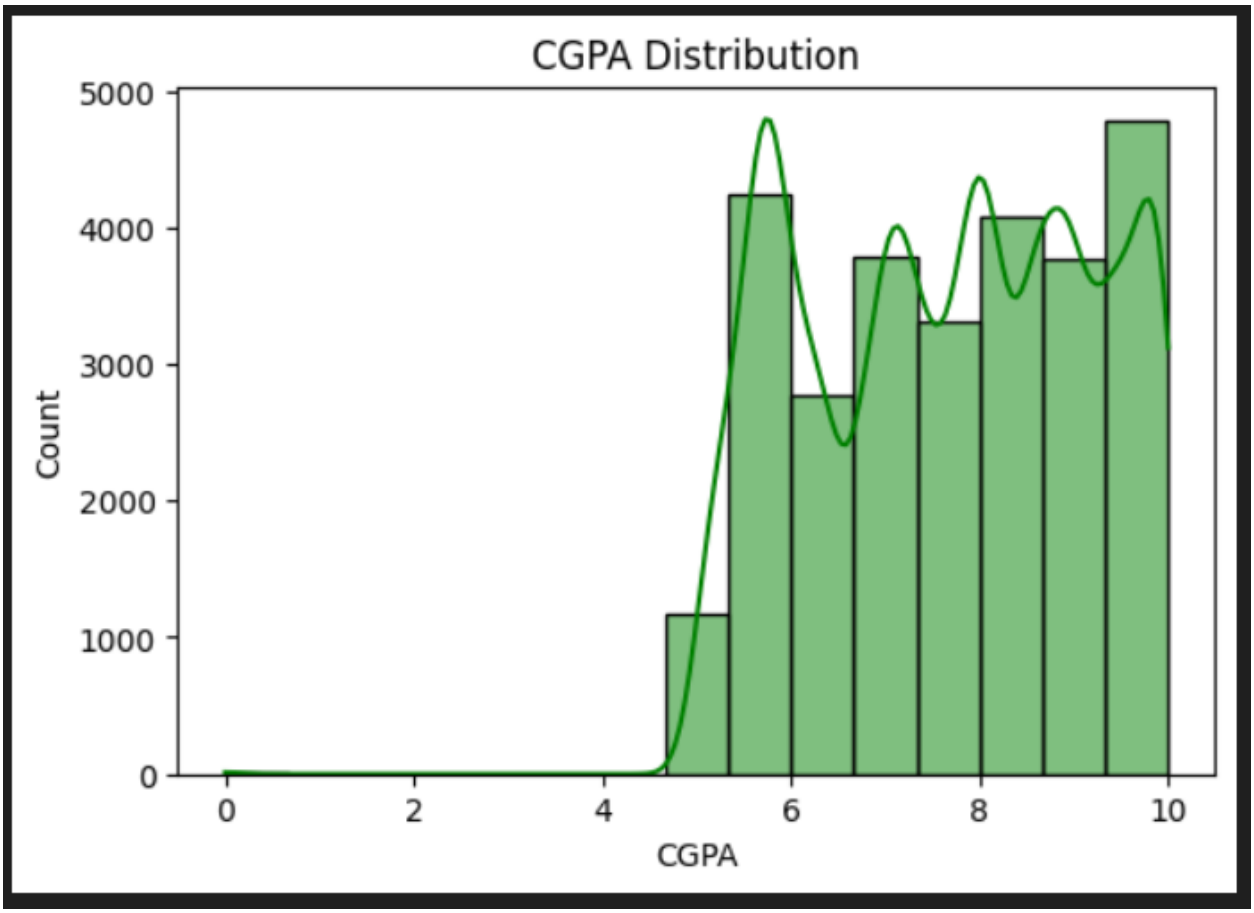
- Missing values can reduce model accuracy and bias results.
- **Techniques:**
  - Numeric columns → fill with mean, median, or mode.
  - Categorical columns → fill with most frequent value (mode) or a placeholder like `"missing"`.
  - Drop rows/columns with too many missing values.

- Use advanced methods like KNN imputation for complex cases.



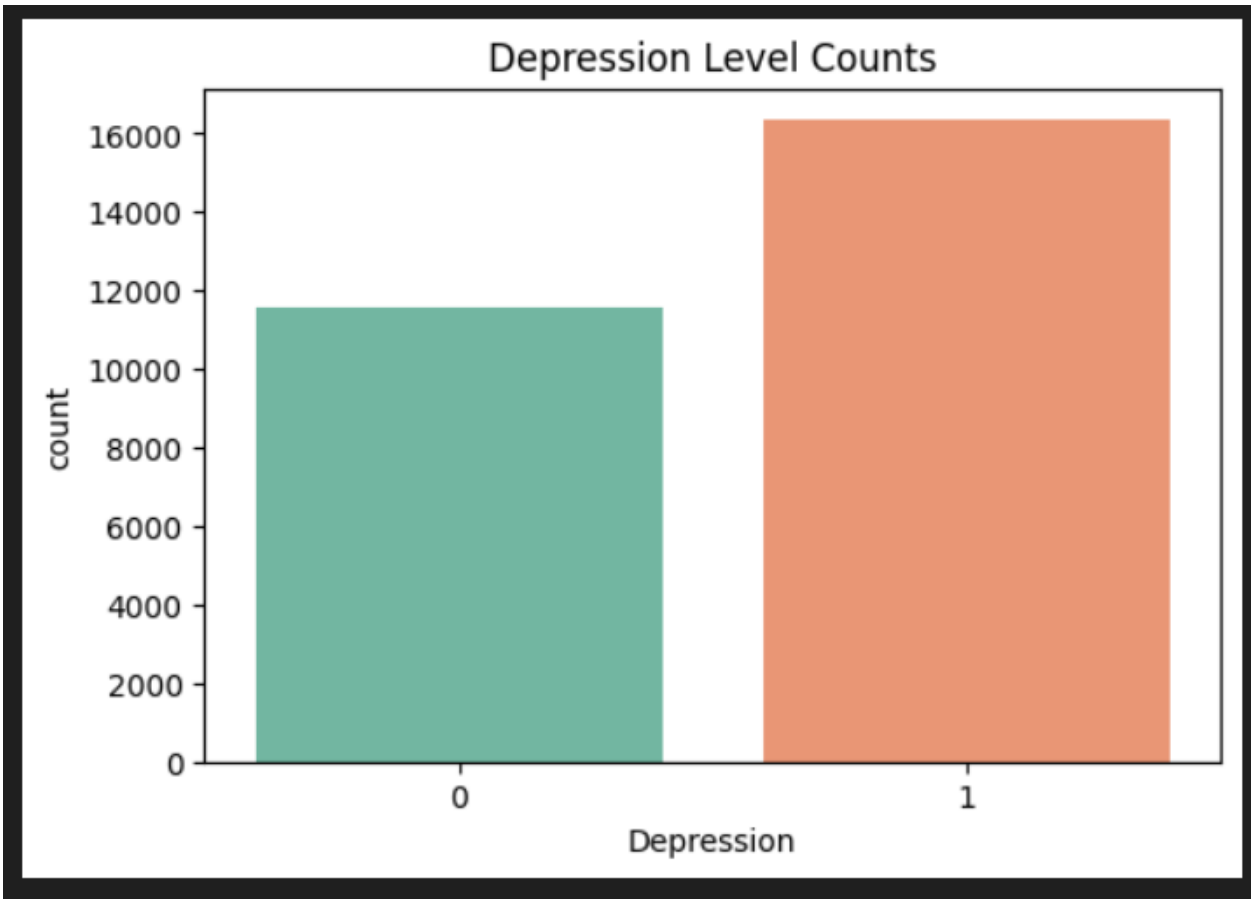
2. Drop Irrelevant Columns

- Some columns don't add value to predictions (e.g., IDs, timestamps if not useful).
- Remove columns with:
  - Unique values for every row (no pattern).
  - Too many missing values.
  - High redundancy with other features.



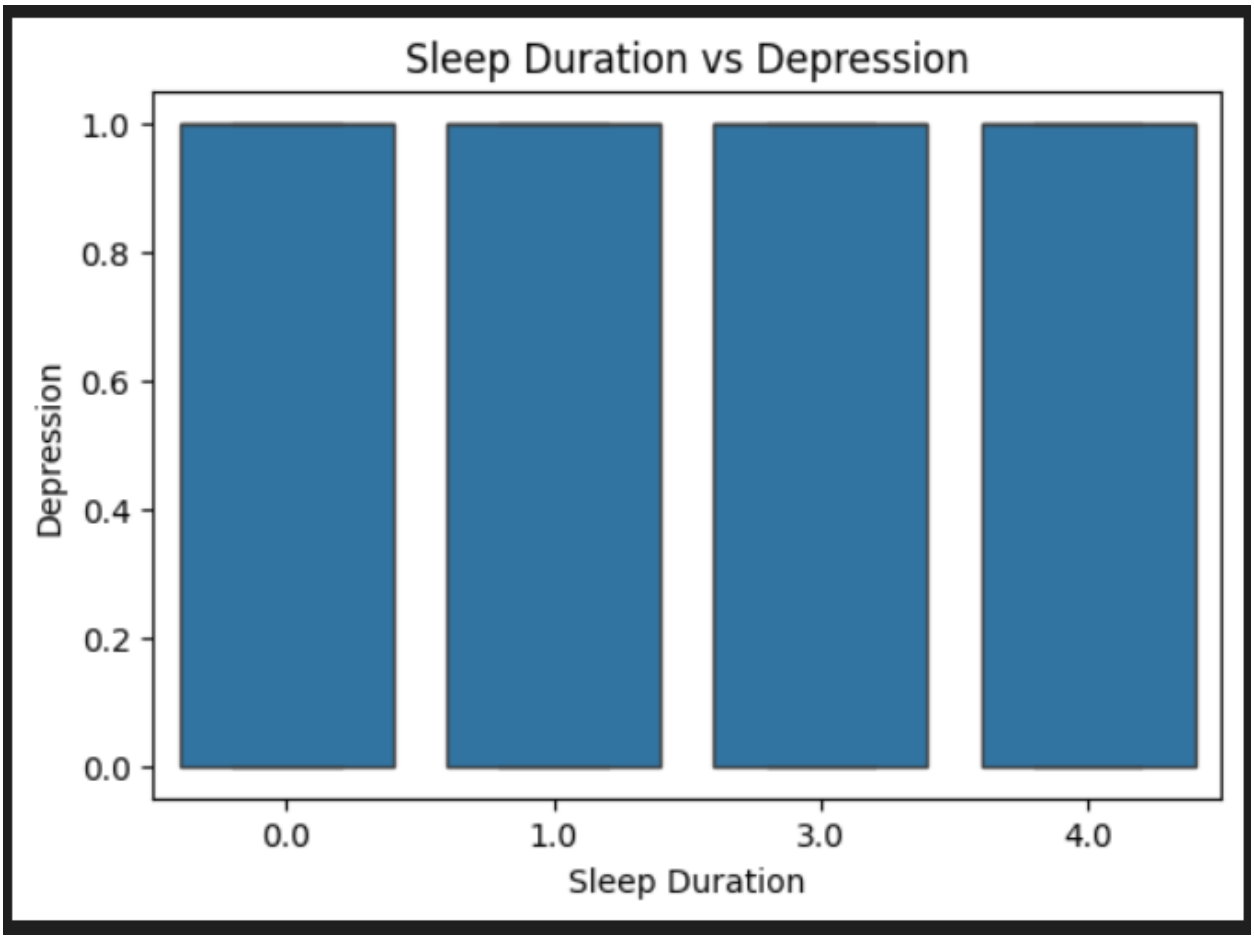
3. Encoding (Convert Categorical → Numeric)

- ML algorithms work better with numbers.
- **Types:**
  - **Label Encoding:** Assigns integers (e.g., Male → 0, Female → 1).
  - **One-Hot Encoding:** Creates binary columns for each category.
  - **Ordinal Encoding:** For ranked data (e.g., Low < Medium < High).



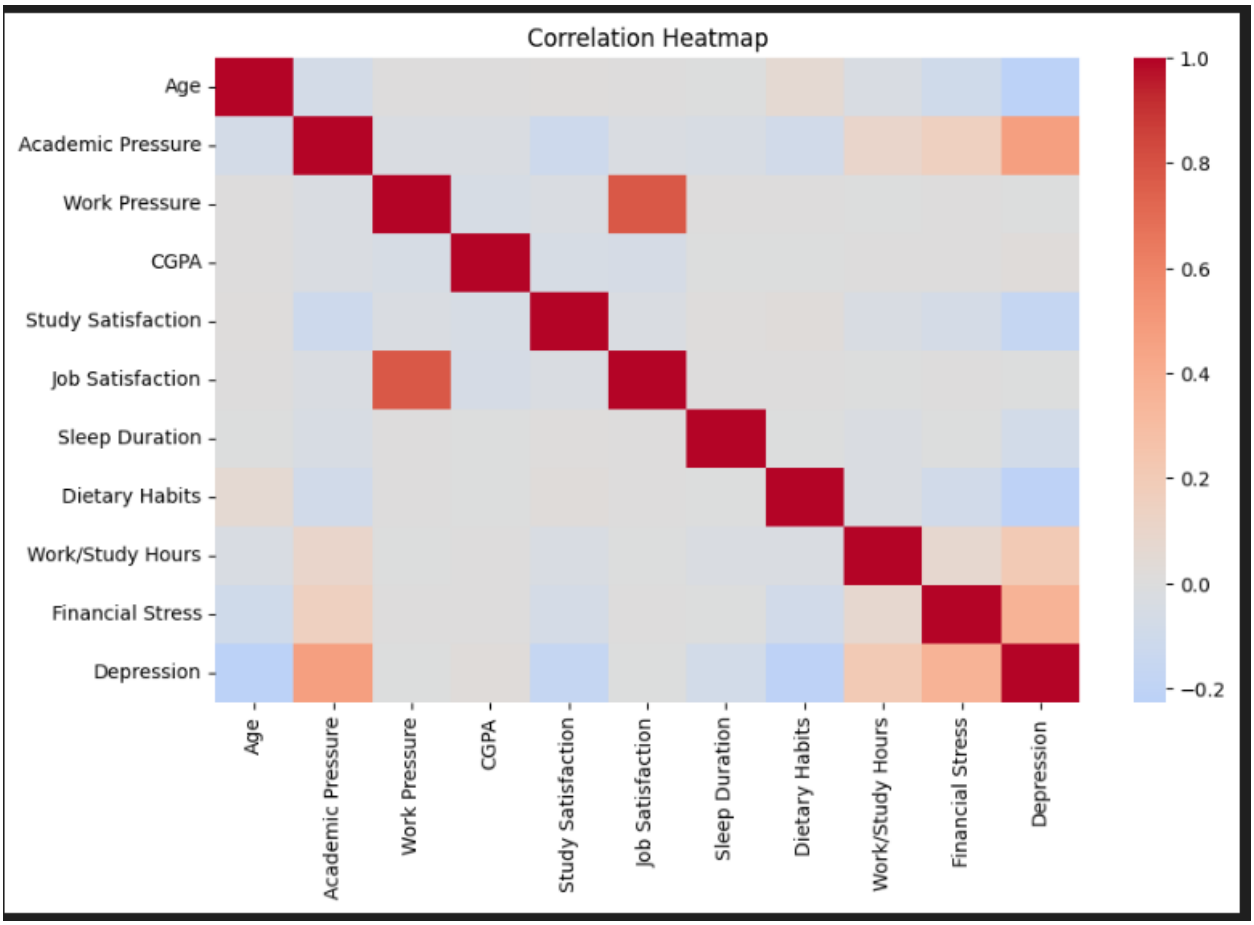
4. Outliers

- Outliers are extreme values that can distort analysis.
- **Detection:** Boxplots, Z-score, IQR method.
- **Handling:**
    - Remove outliers if they are errors.
    - Cap values at threshold (winsorization).
    - Transform (log, square root) to reduce effect.



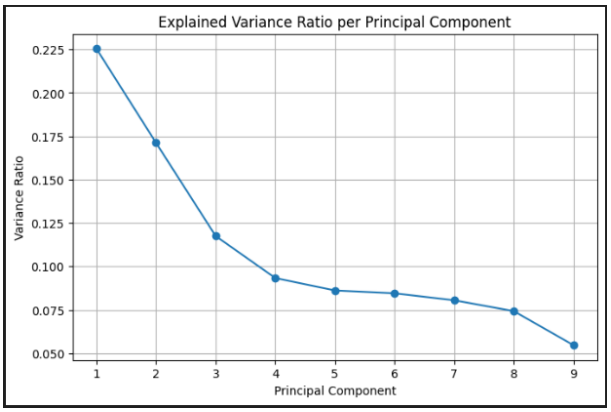
5. Normalization / Scaling

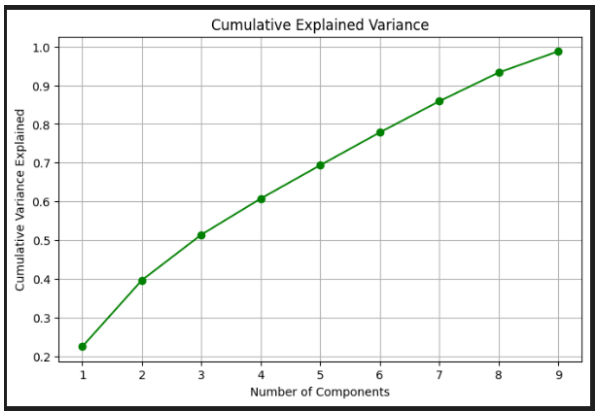
- Ensures all features are on a similar scale.
- **Techniques:**
  - **Min-Max Scaling (Normalization):** Scales values to [0,1].
  - **Standardization (Z-score):** Transforms to mean = 0, std = 1.
  - Needed for algorithms like KNN, SVM, Gradient Descent models.

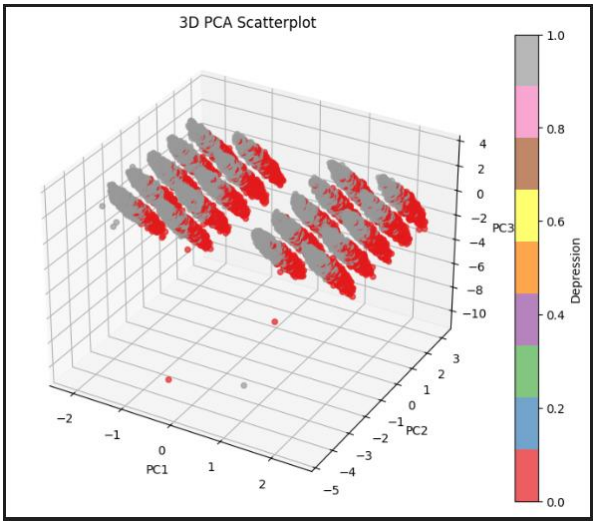


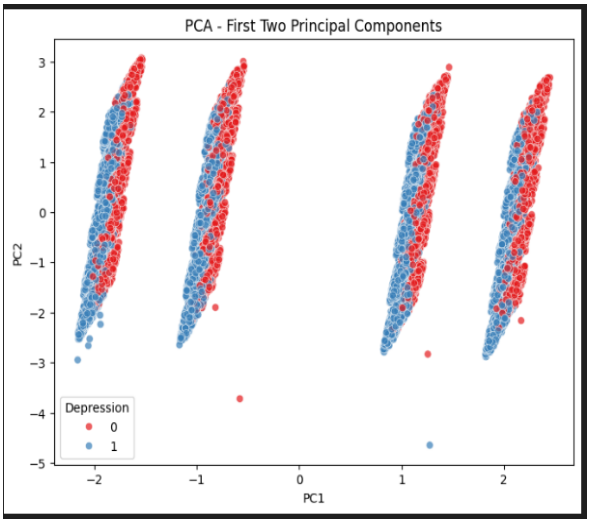
6. PCA (Dimensionality Reduction)

- Principal Component Analysis reduces number of features while keeping most information.
- Converts correlated features into fewer uncorrelated “principal components.”
- Helps:
    - Remove noise.
    - Reduce overfitting.
    - Speed up training.



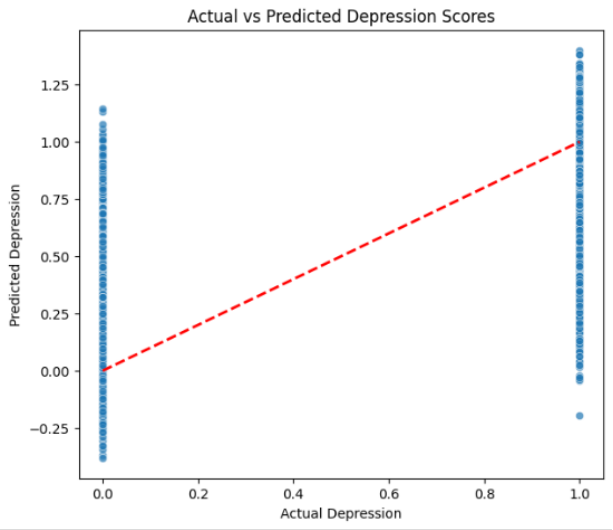


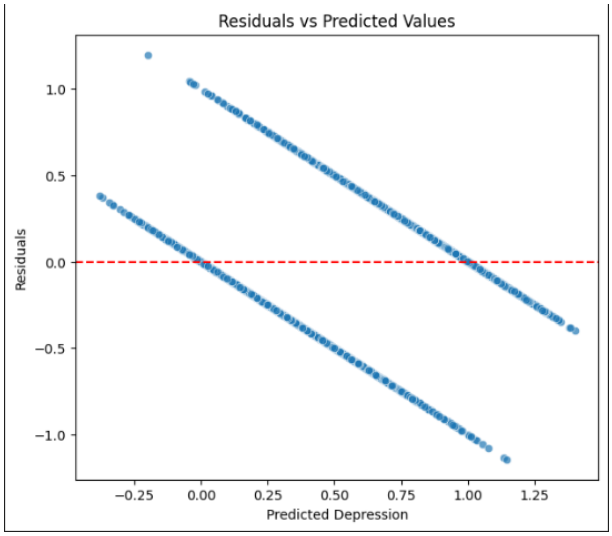


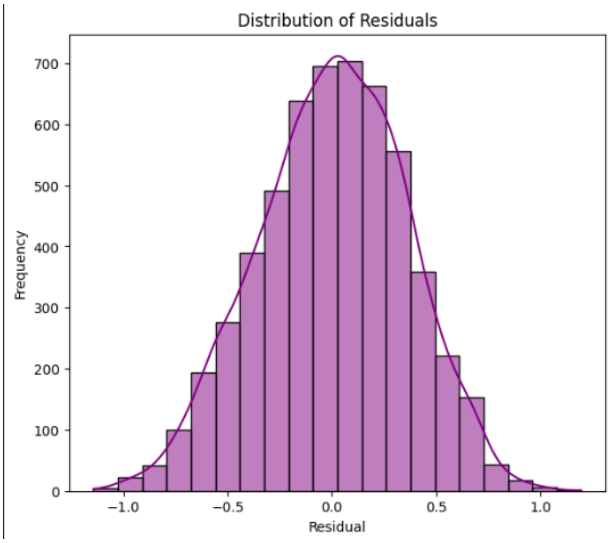


7. Regression (for feature handling)

- Can be used to **predict and impute missing values** instead of dropping them.
- Example: Predict missing age values using regression on other variables.
- Also used for feature relationships in preprocessing.

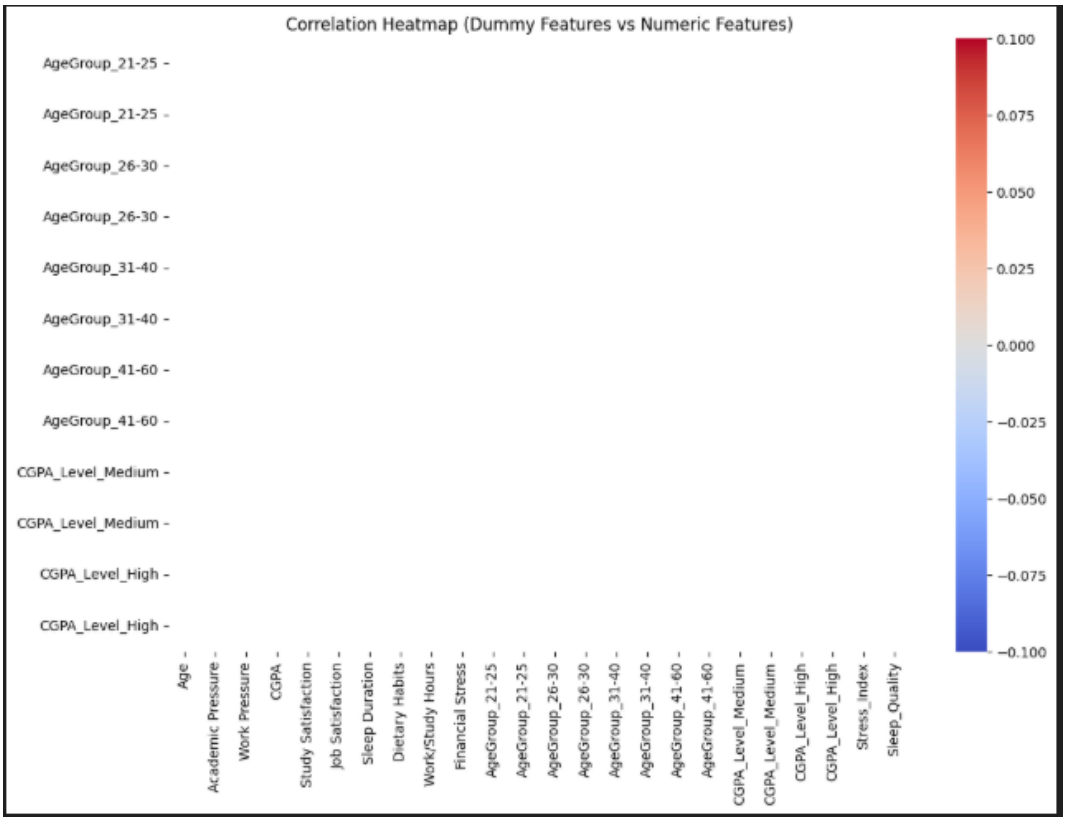






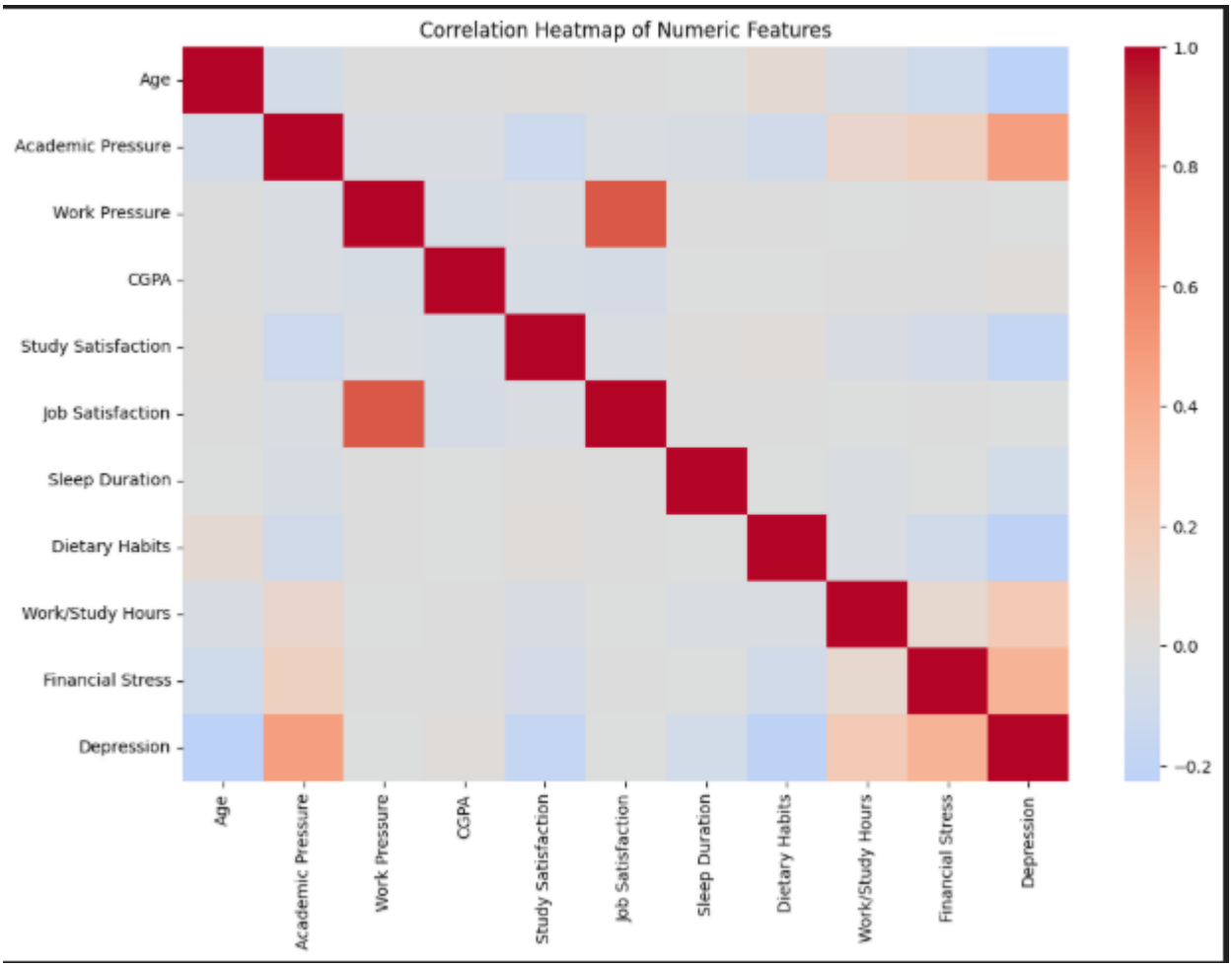
8. Binning / Discretization

- Convert continuous values into **discrete bins**.
- Example: Age → “Child (0–12), Teen (13–19), Adult (20–59), Senior (60+)”.
- Reduces noise, makes patterns more interpretable.



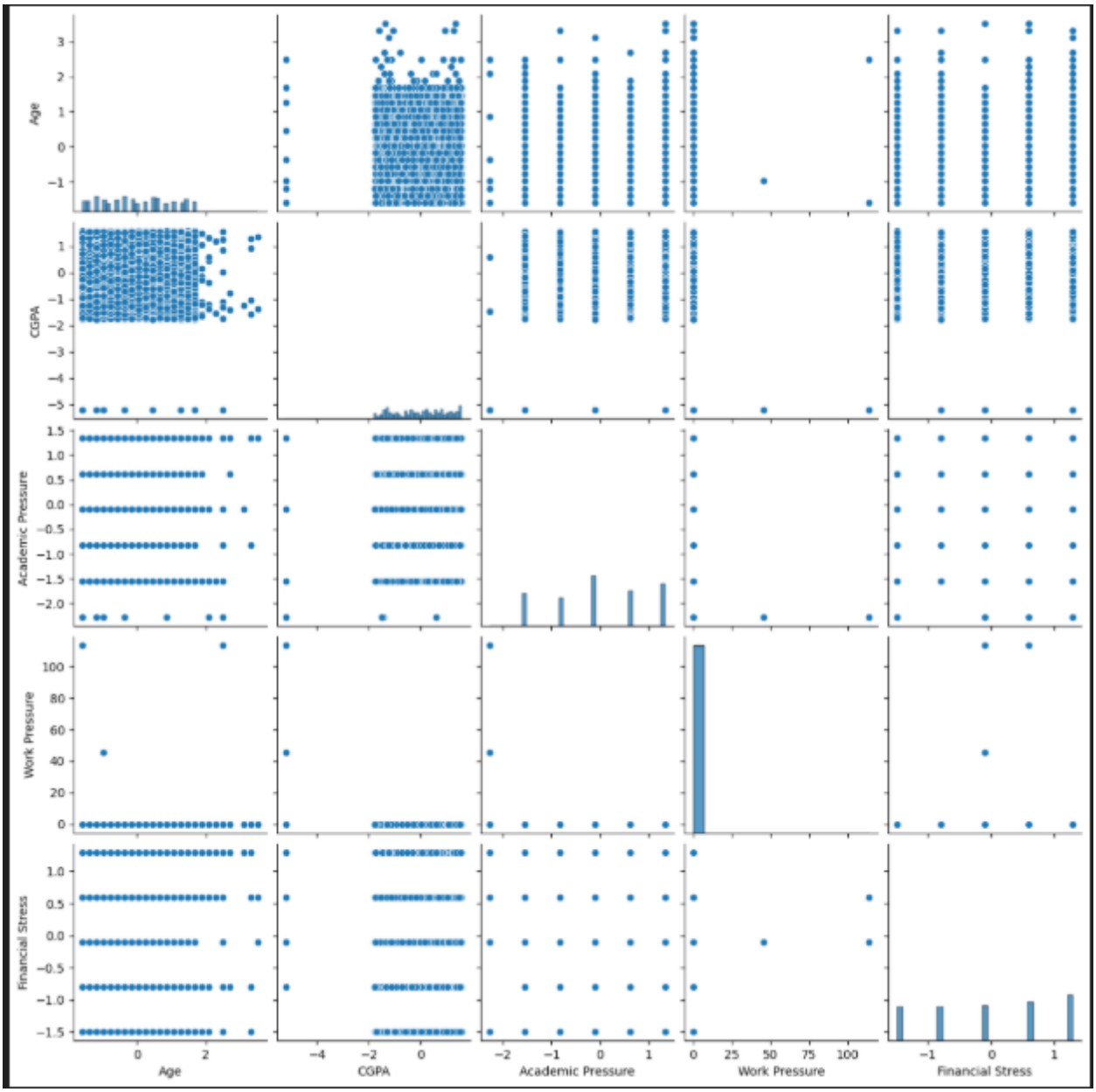
9. Correlation & Multicollinearity Check

- **Correlation analysis** checks relationships between features.
- If two features are highly correlated (multicollinearity), one may be dropped to avoid redundancy.
- Tools: Correlation heatmap, Variance Inflation Factor (VIF).



10. Feature Engineering

- Create new features or transform existing ones to improve model performance.
- Examples:
  - Extracting **day, month, year** from a date.
  - Creating interaction terms (e.g., Price × Quantity = Revenue).
  - Text features like **word count, sentiment score**.
- Adds domain knowledge into preprocessing.



5. Logical Flow and Collaboration

Logical Flow

During the data preprocessing and analysis stage, the logical flow begins with collecting and consolidating raw student data, followed by cleaning to handle missing values, duplicates, and inconsistencies. The process continues with transforming and encoding data into machine-readable formats, performing feature engineering to highlight meaningful attributes, and applying exploratory data analysis (EDA) to uncover trends and correlations related to depression.

Collaboration

Collaboration plays a key role in this phase, as data scientists, domain experts (such as psychologists or counselors), and academic staff work together to validate feature selection, interpret patterns, and ensure the dataset reflects real-world student behavior accurately. This teamwork ensures the processed data is both technically sound and contextually relevant for building a reliable depression detection model.

6.Conclusion

In conclusion, the data preprocessing and analysis phase establishes a strong foundation for the Student Depression Detection AI Model by transforming raw, unstructured information into a clean, reliable, and meaningful dataset. Through systematic cleaning, transformation, feature engineering, and exploratory analysis, the data is prepared to accurately represent patterns linked to depression. This stage not only enhances the quality of the dataset but also ensures that subsequent machine learning models are trained on precise and insightful inputs, ultimately improving the reliability and effectiveness of the depression detection system.