PROBABILITY AND STATISTICS SCE

Analyzing Depression Levels in Students

Roll No	Name	PRN No.	Department
62	Purrva Bothra	22311722	CE(SE)
67	Khushi Shah	22420051	CE(SE)
71	Pranita Kute	22420110	CE(SE)

Subject Teacher
Shweta Tiwaskar

1. Introduction

This mini-project explores statistical insights and probability distributions using a student depression dataset. The objective is to understand patterns and relationships between academic performance and mental health indicators through preprocessing, visualization, and probabilistic analysis.

The aim of this mini project is to:

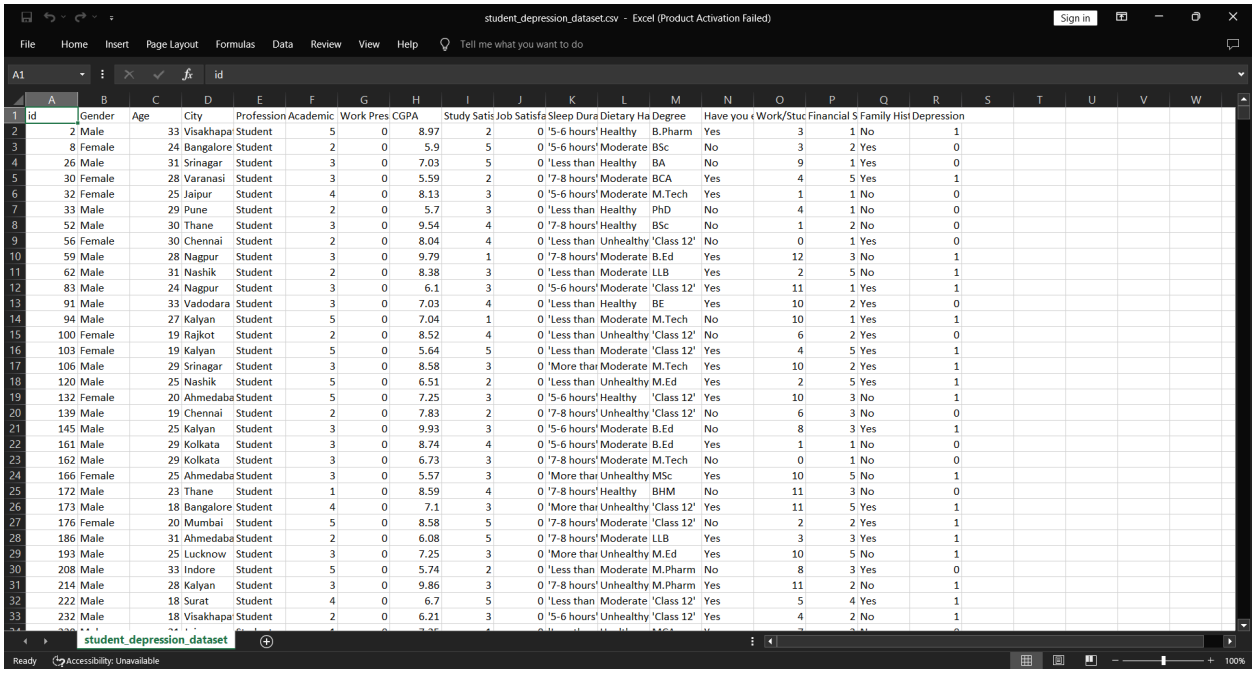
I. Preprocess the dataset for better analysis.
II. Handle missing values appropriately.
III. Apply fundamental concepts from probability and statistics.
IV. Visualize relationships and patterns in the dataset.

2. Dataset Description

Source: student_depression_dataset.csv

Attributes: Includes features like Age, Gender, CGPA, Sleep Duration, Depression, Anxiety, and Panic Attack.

Objective: Analyze the depression prevalence among students and identify statistical relationships.



student_depression_dataset.csv - Excel (Product Activation Failed)

id	Gender	Age	City	Profession	Academic	Work Pres	CGPA	Study Satis	Job Satisfa	Sleep Dura	Dietary Ha	Degree	Have you e	Work/Stuc	Financial S	Family Hist	Depression
2	Male	33	Visakhapa	Student	5	0	8.97	2	0	'5-6 hours'	Healthy	B.Pharm	Yes	3	1	No	1
8	Female	24	Bangalore	Student	2	0	5.9	5	0	'5-6 hours'	Moderate	BSc	No	3	2	Yes	0
26	Male	31	Srinagar	Student	3	0	7.03	5	0	'Less than	Healthy	BA	No	9	1	Yes	0
30	Female	28	Varanasi	Student	3	0	5.59	2	0	'7-8 hours'	Moderate	BCA	Yes	4	5	Yes	1
32	Female	25	Jaipur	Student	4	0	8.13	3	0	'5-6 hours'	Moderate	M.Tech	Yes	1	1	No	0
33	Male	29	Pune	Student	2	0	5.7	3	0	'Less than	Healthy	PhD	No	4	1	No	0
52	Male	30	Thane	Student	3	0	9.54	4	0	'7-8 hours'	Healthy	BSc	No	1	2	No	0
56	Female	30	Chennai	Student	2	0	8.04	4	0	'Less than	Unhealthy	'Class 12'	No	0	1	Yes	0
59	Male	28	Nagpur	Student	3	0	9.79	1	0	'7-8 hours'	Moderate	B.Ed	Yes	12	3	No	1
62	Male	31	Nashik	Student	2	0	8.38	3	0	'Less than	Moderate	LLB	Yes	2	5	No	1
83	Male	24	Nagpur	Student	3	0	6.1	3	0	'5-6 hours'	Moderate	'Class 12'	Yes	11	1	Yes	1
91	Male	33	Vadodara	Student	3	0	7.03	4	0	'Less than	Healthy	BE	Yes	10	2	Yes	0
94	Male	27	Kalyan	Student	5	0	7.04	1	0	'Less than	Moderate	M.Tech	No	10	1	Yes	1
100	Female	19	Rajkot	Student	2	0	8.52	4	0	'Less than	Unhealthy	'Class 12'	No	6	2	Yes	0
103	Female	19	Kalyan	Student	5	0	5.64	5	0	'Less than	Moderate	'Class 12'	Yes	4	5	Yes	1
106	Male	29	Srinagar	Student	3	0	8.58	3	0	'More thar	Moderate	M.Tech	Yes	10	2	Yes	1
120	Male	25	Nashik	Student	5	0	6.51	2	0	'Less than	Unhealthy	M.Ed	Yes	2	5	Yes	1
132	Female	20	Ahmedaba	Student	5	0	7.25	3	0	'5-6 hours'	Healthy	'Class 12'	Yes	10	3	No	1
139	Male	19	Chennai	Student	2	0	7.83	2	0	'7-8 hours'	Unhealthy	'Class 12'	No	6	3	No	0
145	Male	25	Kalyan	Student	3	0	9.93	3	0	'5-6 hours'	Moderate	B.Ed	No	8	3	Yes	1
161	Male	29	Kolkata	Student	3	0	8.74	4	0	'5-6 hours'	Moderate	B.Ed	Yes	1	1	No	0
162	Male	29	Kolkata	Student	3	0	6.73	3	0	'7-8 hours'	Moderate	M.Tech	No	0	1	No	0
166	Female	25	Ahmedaba	Student	3	0	5.57	3	0	'More thar	Unhealthy	MSc	Yes	10	5	No	1
172	Male	23	Thane	Student	1	0	8.59	4	0	'7-8 hours'	Healthy	BHM	No	11	3	No	0
173	Male	18	Bangalore	Student	4	0	7.1	3	0	'More thar	Unhealthy	'Class 12'	Yes	11	5	Yes	1
176	Female	20	Mumbai	Student	5	0	8.58	5	0	'7-8 hours'	Moderate	'Class 12'	No	2	2	Yes	1
186	Male	31	Ahmedaba	Student	2	0	6.08	5	0	'7-8 hours'	Moderate	LLB	Yes	3	3	Yes	1
193	Male	25	Lucknow	Student	3	0	7.25	3	0	'More thar	Unhealthy	M.Ed	Yes	10	5	No	1
208	Male	33	Indore	Student	5	0	5.74	2	0	'Less than	Moderate	M.Pharm	No	8	3	Yes	0
214	Male	28	Kalyan	Student	3	0	9.86	3	0	'7-8 hours'	Unhealthy	M.Pharm	Yes	11	2	No	1
222	Male	18	Surat	Student	4	0	6.7	5	0	'Less than	Moderate	'Class 12'	Yes	5	4	Yes	1
232	Male	18	Visakhapa	Student	2	0	6.21	3	0	'5-6 hours'	Unhealthy	'Class 12'	Yes	4	2	No	1

3. Data Preprocessing

Loading and Displaying Data: Used pandas to load the dataset and preview entries.

```
In [2]: import numpy as np
        import pandas as pd
        import matplotlib.pyplot as plt
        import seaborn as sns
        from scipy.stats import pearsonr

        df = pd.read_csv("D:/PAS/student_depression_dataset.csv")
        print(df.head())


        # numpy, pandas: Data manipulation.
        # matplotlib, seaborn: Visualization.
        # scipy.stats.pearsonr: Used for correlation.
        # df.head(): Shows the first 5 rows of the dataset.
```

```
   id  Gender   Age         City Profession  Academic Pressure  \
0   2    Male  33.0  Visakhapatnam   Student                5.0
1   8  Female  24.0     Bangalore   Student                2.0
2  26    Male  31.0     Srinagar    Student                3.0
3  30  Female  28.0     Varanasi    Student                3.0
4  32  Female  25.0       Jaipur    Student                4.0

   Work Pressure  CGPA  Study Satisfaction  Job Satisfaction  \
0            0.0  8.97                2.0               0.0
1            0.0  5.90                5.0               0.0
2            0.0  7.03                5.0               0.0
3            0.0  5.59                2.0               0.0
4            0.0  8.13                3.0               0.0

      Sleep Duration Dietary Habits   Degree  \
0          '5-6 hours'      Healthy  B.Pharm
1          '5-6 hours'     Moderate      BSc
2  'Less than 5 hours'      Healthy       BA
3          '7-8 hours'     Moderate      BCA
4          '5-6 hours'     Moderate  M.Tech
```

Handling Missing Values:

```
In [5]: # Handle Missing Values
        df.isnull().sum()

Out[5]: id                                    0
        Gender                                0
        Age                                   0
        City                                  0
        Profession                            0
        Academic Pressure                     0
        Work Pressure                         0
        CGPA                                  0
        Study Satisfaction                    0
        Job Satisfaction                      0
        Sleep Duration                        0
        Dietary Habits                        0
        Degree                                0
        Have you ever had suicidal thoughts ?  0
        Work/Study Hours                      0
        Financial Stress                      0
        Family History of Mental Illness      0
        Depression                            0
        dtype: int64
```

4. Methodology

Statistical Analysis and Visualization

Libraries: pandas, seaborn, matplotlib, scipy.stats

Techniques used:

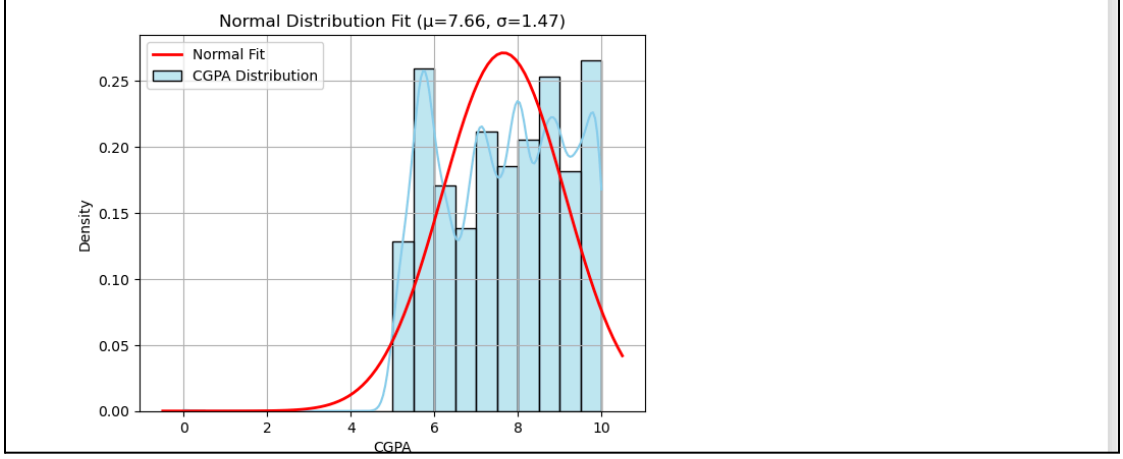
Histogram & KDE to visualize CGPA and fit a normal distribution

```
In [25]: # removes any NaN (missing) values
         cgpa_data = df['CGPA'].dropna()

         # Plot histogram with normal curve
         sns.histplot(cgpa_data, kde=True, stat="density", bins=20, color='skyblue', label="CGPA Distribution")
         # stat="density" normalizes the histogram, so the area under the curve equals 1.

         # Fit a normal distribution
         mu, std = norm.fit(cgpa_data)
         xmin, xmax = plt.xlim()
         x = np.linspace(xmin, xmax, 100)
         p = norm.pdf(x, mu, std)
         plt.plot(x, p, 'r', linewidth=2, label='Normal Fit')

         plt.title(f'Normal Distribution Fit (μ={mu:.2f}, σ={std:.2f})')
         plt.xlabel('CGPA')
         plt.ylabel('Density')
         plt.legend()
         plt.grid(True)
         plt.show()
```



5. Probability Concepts Implemented

a. Binomial Distribution

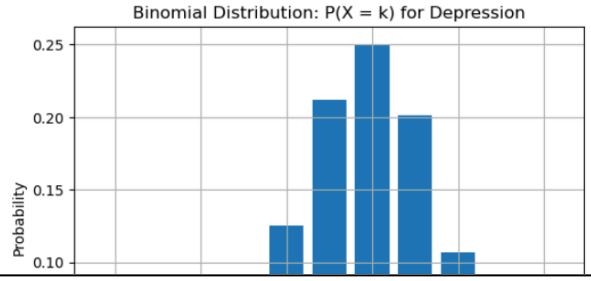
Scenario: Probability of students being depressed in a group of 10

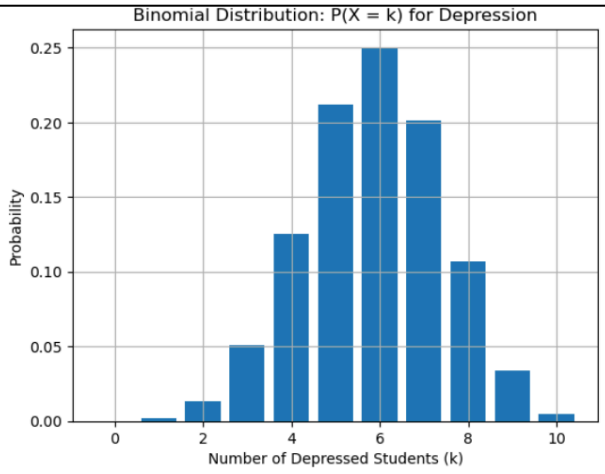
In [24]:
```
# Probability of depression
p = df['Depression'].mean()
# This calculates the proportion of students marked as depressed in the dataset. It assumes the 'Depression' column has binary values like 0 (not depressed) and 1 (depressed).
# So, p is the probability of depression for one student.
n = 10  # random sample of 10 students

# Binomial probability mass function
x = range(0, n+1)
probs = binom.pmf(x, n, p)
# binom.pmf(x, n, p) gives you the probability of exactly x students being depressed in a sample of n students, assuming the probability of any one student being depressed is p.

# Plotting
plt.bar(x, probs)
plt.title('Binomial Distribution: P(X = k) for Depression')
plt.xlabel('Number of Depressed Students (k)')
plt.ylabel('Probability')
plt.grid(True)
plt.show()

# Show calculated probabilities
for i, prob in zip(x, probs):
    print(f"P(X = {i}) = {prob:.4f}")
```






```
P(X = 0) = 0.0001
P(X = 1) = 0.0021
P(X = 2) = 0.0134
P(X = 3) = 0.0506
P(X = 4) = 0.1252
P(X = 5) = 0.2122
P(X = 6) = 0.2497
P(X = 7) = 0.2016
P(X = 8) = 0.1068
P(X = 9) = 0.0335
P(X = 10) = 0.0047
```

b. Normal Distribution Fit

Applied to: CGPA values

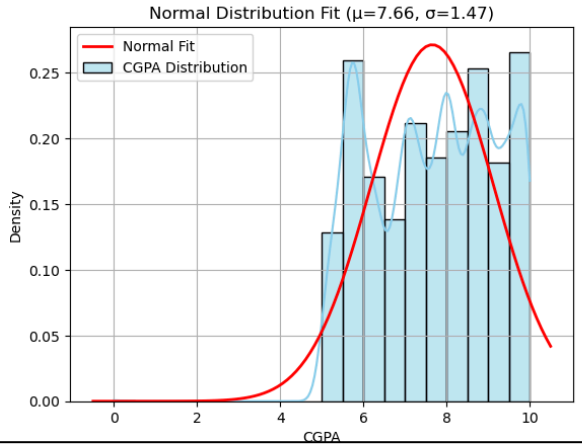
Purpose: Check if CGPA follows a normal distribution

```
In [25]: # removes any NaN (missing) values
cgpa_data = df['CGPA'].dropna()

# Plot histogram with normal curve
sns.histplot(cgpa_data, kde=True, stat="density", bins=20, color='skyblue', label="CGPA Distribution")
# stat="density" normalizes the histogram, so the area under the curve equals 1.

# Fit a normal distribution
mu, std = norm.fit(cgpa_data)
xmin, xmax = plt.xlim()
x = np.linspace(xmin, xmax, 100)
p = norm.pdf(x, mu, std)
plt.plot(x, p, 'r', linewidth=2, label='Normal Fit')

plt.title(f'Normal Distribution Fit (μ={mu:.2f}, σ={std:.2f})')
plt.xlabel('CGPA')
plt.ylabel('Density')
plt.legend()
plt.grid(True)
plt.show()
```



c. Descriptive Statistics

Included Measures:

Mean

```
In [32]: # Population vs Sample + Sample Mean
# Full data as population
population = df['CGPA'].dropna()

# Draw random sample of size 30
sample = population.sample(30, random_state=42)

# Calculate means
population_mean = population.mean()
sample_mean = sample.mean()

print(f"Population Mean (CGPA): {population_mean:.2f}")
print(f"Sample Mean (CGPA, n=30): {sample_mean:.2f}")
```

```
Population Mean (CGPA): 7.66
Sample Mean (CGPA, n=30): 7.58
```

6. Input of Work

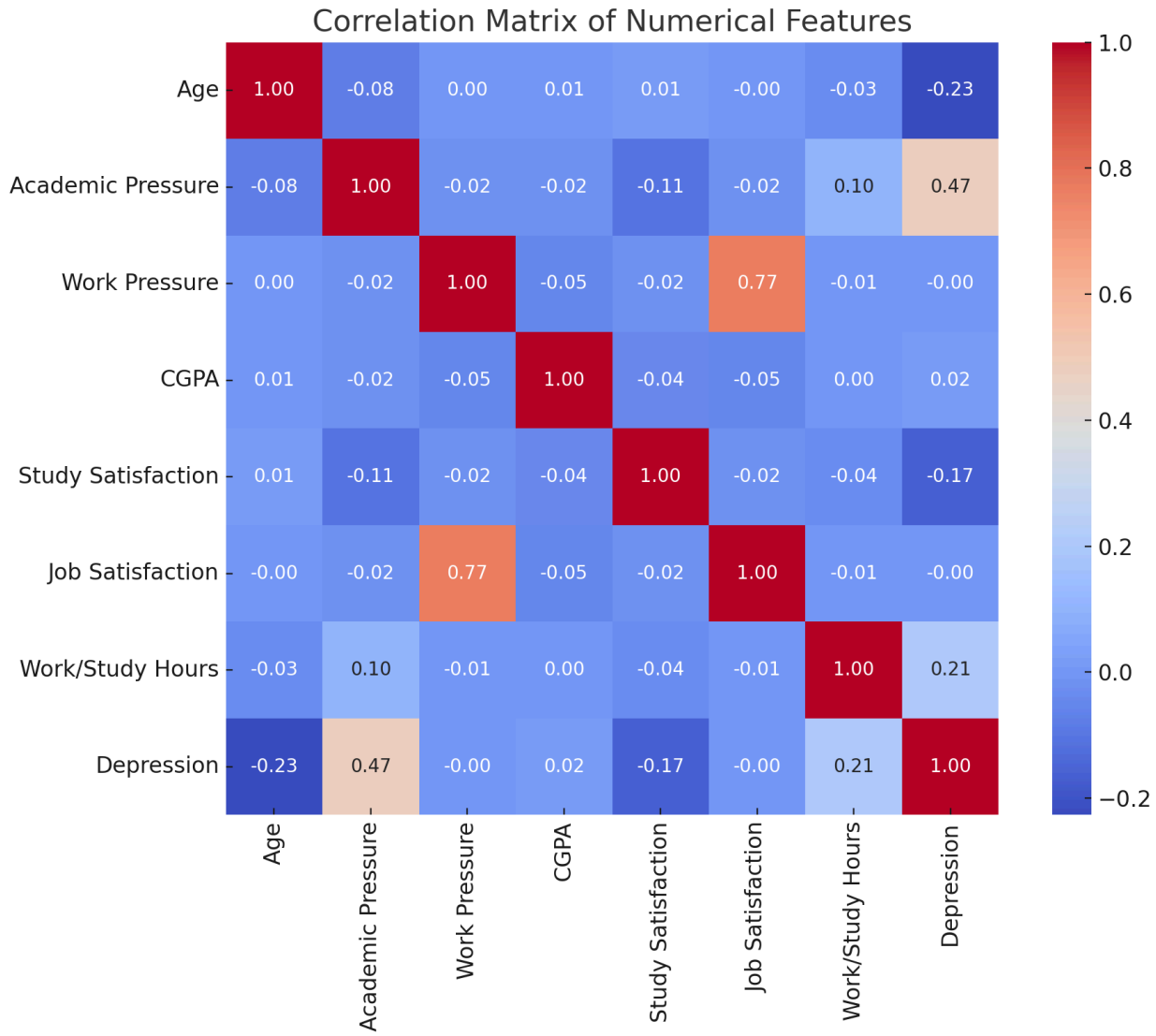
Tools Used: Jupyter Notebook, Python (NumPy, Pandas, Seaborn, SciPy)

Dataset Size: ~Around 28000 entries

Approach:

Preprocessing data → visualize → apply statistical models → interpret results

7. Observations & Analysis:



Depression shows some correlation with:

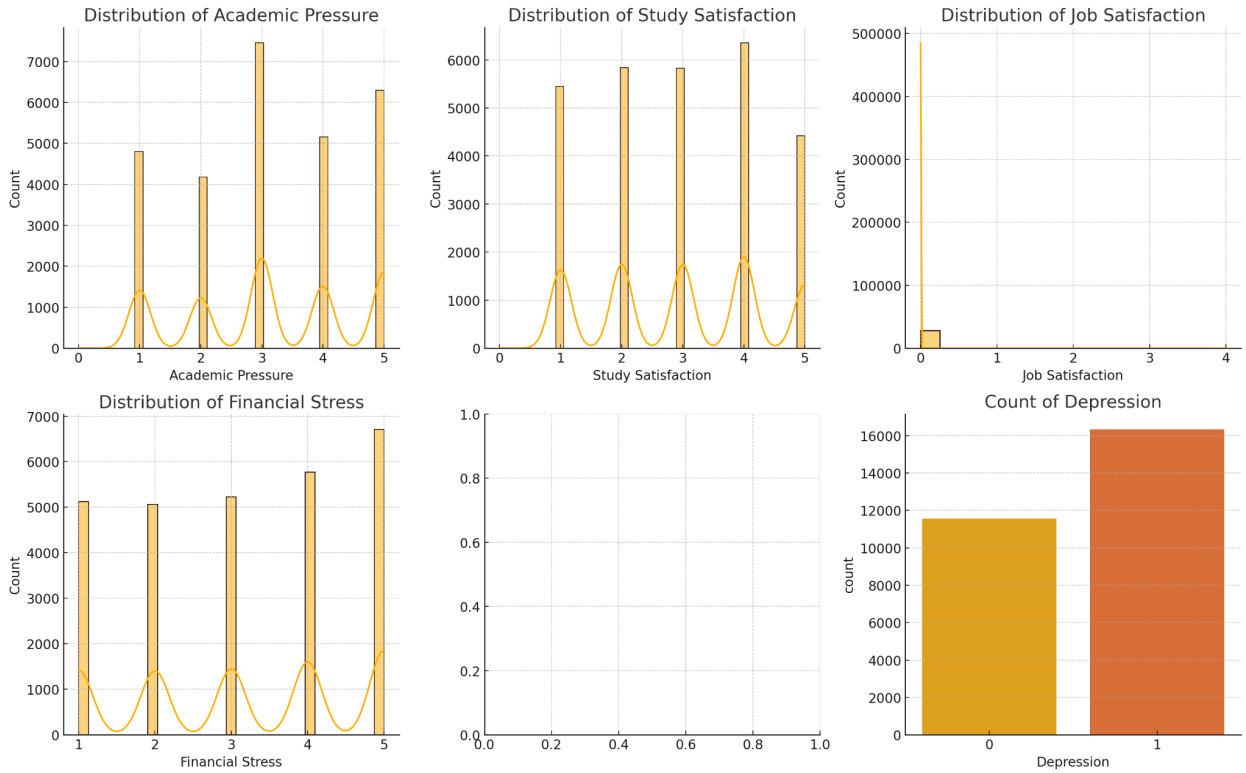
Study Satisfaction (negative)

Academic Pressure (positive)

Job Satisfaction (negative)

Financial Stress (mild positive)

Hence, these could be important variables for prediction.



Observations:

Academic Pressure is mostly skewed to the lower range, but varies.

Study and Job Satisfaction tend to be clustered around mid-to-low values.

Financial Stress has a varied distribution, with some concentration at the lower end.

Depression shows class imbalance (more 0s than 1s).

Choosing the right features for model performance and interpretability. Based on:

1. **Correlation analysis**
2. **Domain knowledge**
3. **Exploratory visualizations**

Features to Train the Models

Psychological & Lifestyle Indicators:

- **Academic Pressure** → Correlated with depression (+ve)
- **Study Satisfaction** → Negatively correlated (lower satisfaction = more depression)
- **Job Satisfaction** → Same as above
- **Financial Stress** → Makes logical sense and had some signal
- **Sleep Duration** → Sleep is a major mental health factor

Health and History:

- **Have you ever had suicidal thoughts?** → Strongly indicative
- **Family History of Mental Illness** → Known risk factor

Demographic Factors:

- **Gender** → Often shows differing depression patterns
- **Age** → Can impact life stressors and experiences
- **Degree** or **Profession** → May relate to academic/job pressure

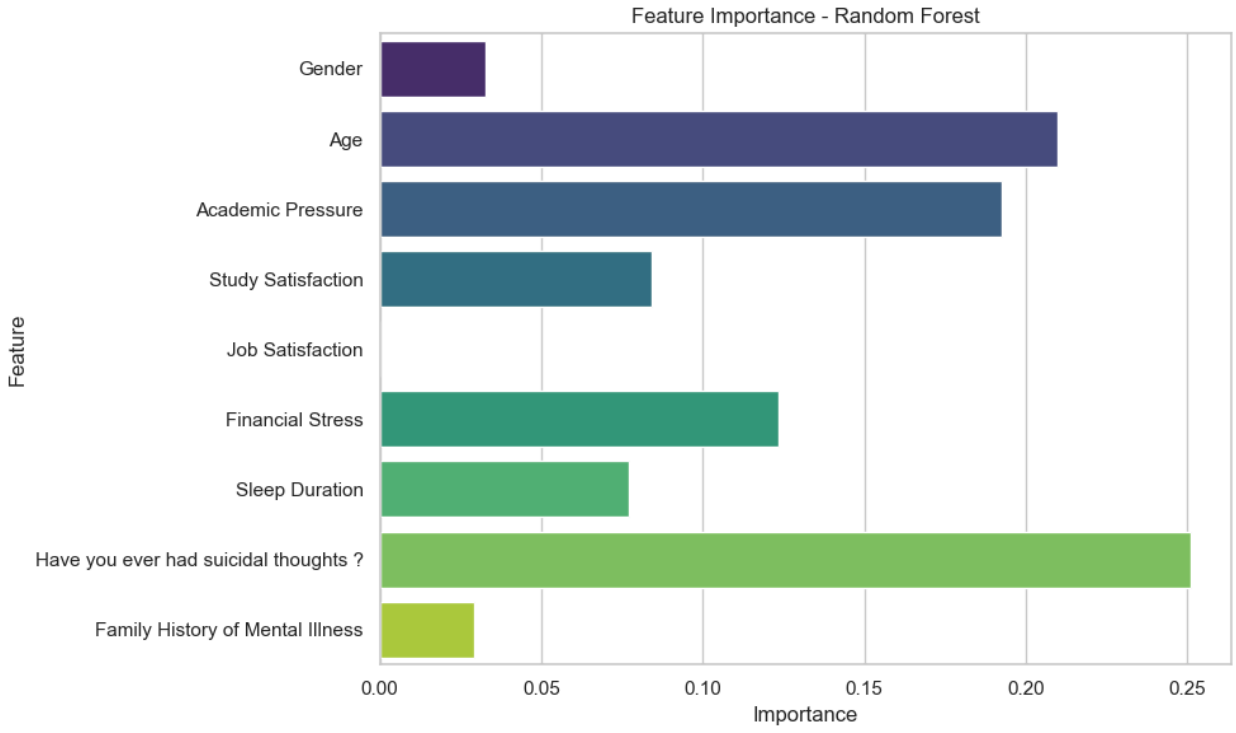
---

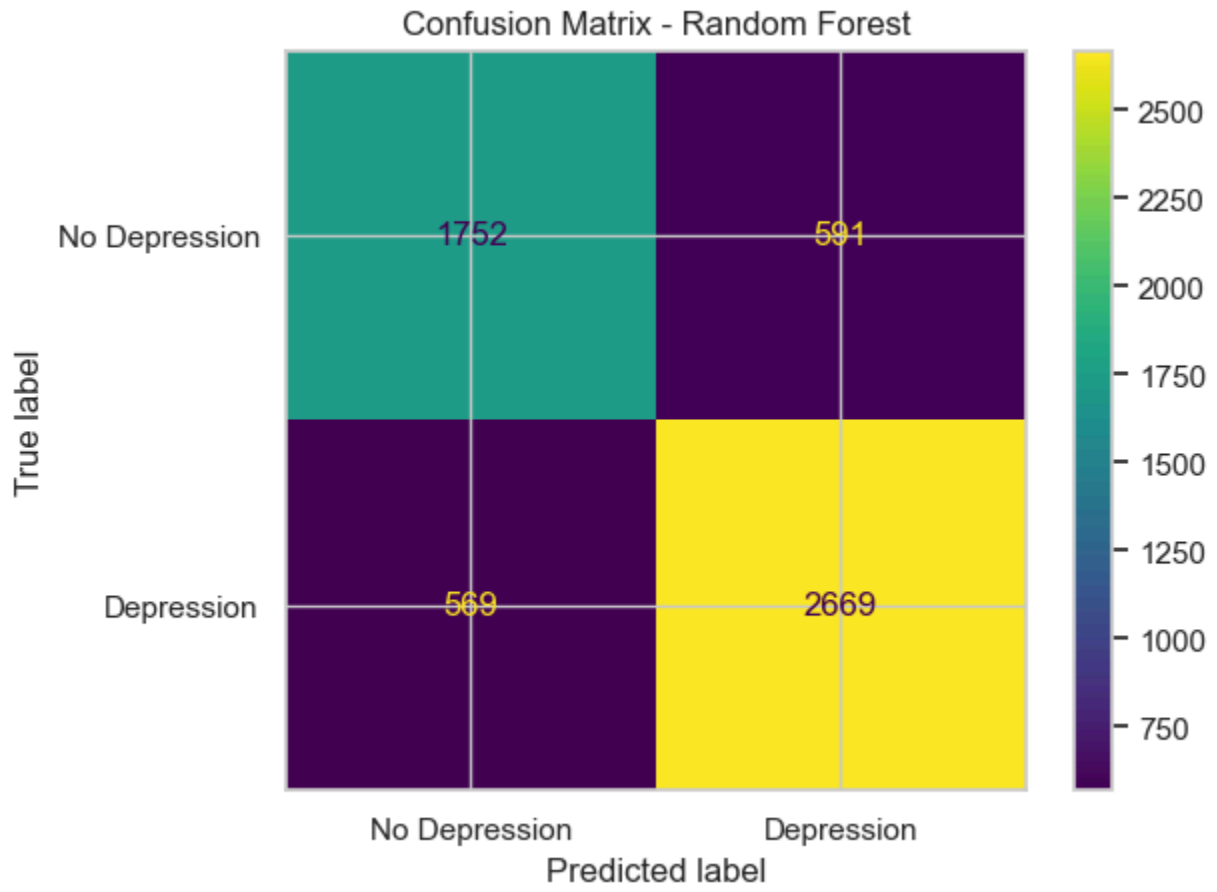
Features to Drop

- **City**: Too high-cardinality, needs encoding or grouping
- **id**: Just an identifier, remove it
- **CGPA**: Could be useful, but not very correlated in the matrix
- **Work/Study Hours**: Might add noise if not normalized

Feature Set

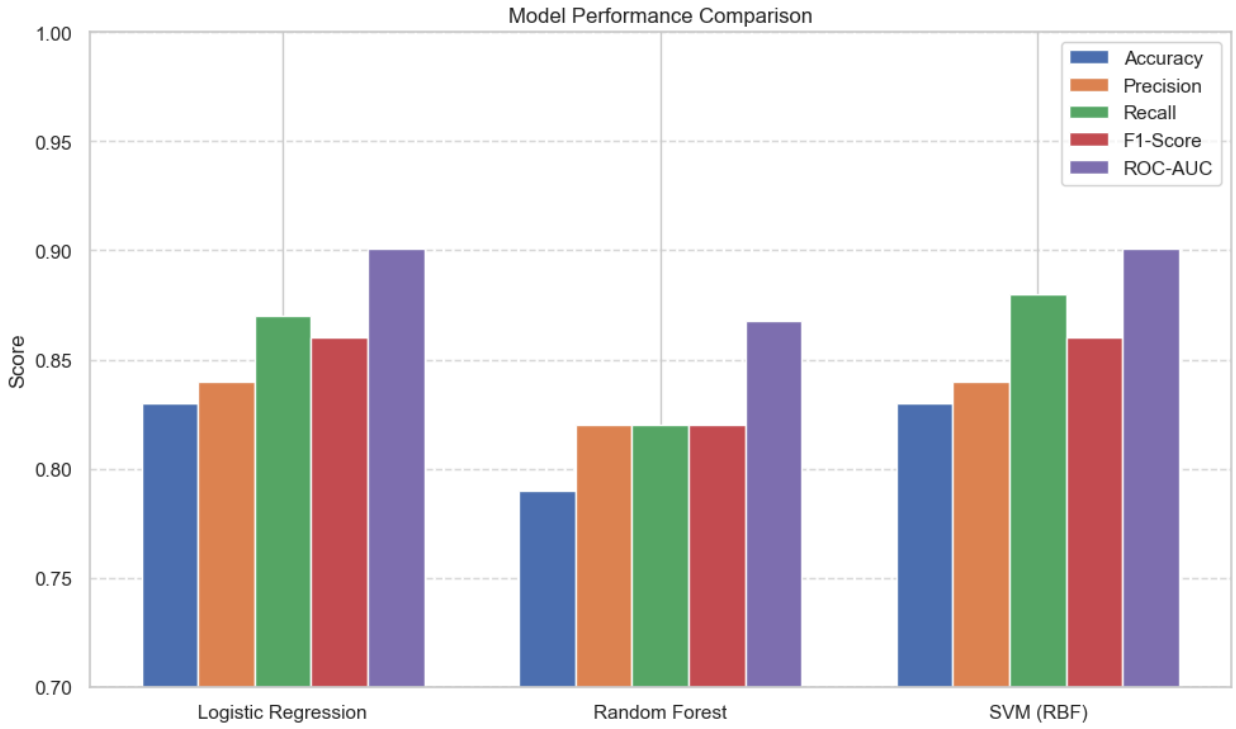
Feature	Type
Gender	Categorical
Age	Numeric
Academic Pressure	Numeric
Study Satisfaction	Numeric
Job Satisfaction	Numeric
Financial Stress	Categorical
Sleep Duration	Categorical
Have you ever had suicidal thoughts?	Categorical
Family History of Mental Illness	Categorical
Depression (Target)	Binary





Key Metrics:

- **Accuracy** = (TN + TP) / Total = (1752 + 2669) / 5581 ≈ **0.79**
- **Precision (Depression)** = TP / (TP + FP) = 2669 / (2669 + 591) ≈ **0.82**
- **Recall (Depression)** = TP / (TP + FN) = 2669 / (2669 + 569) ≈ **0.82**
- **F1 Score (Depression)** = Harmonic mean of precision and recall ≈ **0.82**



SVM and Logistic Regression perform similarly and better than Random Forest here.

Random Forest has slightly lower recall and ROC-AUC, meaning it misses more true depression cases.

All models are strong, but Logistic Regression is a good balance of interpretability + performance.

8. Conclusion

These insights help quantify the mental health landscape among students and encourage data-driven interventions.