

Predicting Student Mental Health Risk Using Academic and Lifestyle Indicators

[Video](#)|[Github](#)

1. Project Definition

1.1 Problem Statement

Mental health has become a major topic on college campuses. Many students juggle demanding course loads, long study hours, financial responsibilities, and inconsistent sleep schedules, all of which can affect their emotional well-being. The challenge is that students often do not recognize early warning signs, and universities usually rely on self-reported concerns rather than predictive indicators.

In this project I set out to explore whether a combination of academic and lifestyle variables can help predict mental health risk among students. I used the Student Depression Dataset, which includes features such as CGPA, academic pressure, work pressure, study satisfaction, sleep duration, financial stress, and whether a student reports suicidal thoughts. I wanted to understand how these factors interact and whether a predictive model could reasonably identify students at higher risk.

This work ties directly to the content of our course because it requires going through an entire data pipeline. I had to clean and preprocess the dataset, structure it using SQL tables, engineer new features, visualize relationships, and then use machine learning models to classify mental health risk. Each step reflects the data management and modeling techniques emphasized in CS439.

1.2 Strategic Aspects

The strategy behind the project was to take a multidimensional view of mental health indicators. Instead of focusing on a single factor like academic pressure or CGPA, the idea was to combine academic behaviors, lifestyle habits, and emotional indicators into one predictive framework. This allows the model to capture interactions that might not be apparent when looking at variables individually.

This approach reflects the goals of our course because it required a complete data science workflow. I had to process and structure the dataset, build SQL tables, create visualizations to understand patterns, engineer new features to improve model depth, and then train and evaluate machine learning models. All of these components connect directly to what we learned in CS439 about the importance of data cleaning, storage, integration, modeling, and interpretation.

The project emphasizes that mental health risk is shaped by multiple connected factors. Understanding these connections can help guide earlier interventions and promote healthier routines for students.

1.3 Relation to Course Concepts

This project incorporates many of the concepts and techniques we worked with throughout CS439. I began by cleaning and preprocessing the dataset using Pandas. This included identifying and filling missing values, standardizing columns, converting categorical text values into numeric values, and creating new engineered features such as sleep deficit, academic score, and stress to study ratio. These steps matched the preprocessing activities we practiced in recitations.

For the database portion, I organized the cleaned dataset into four SQL tables: demographics, academics, lifestyle, and mental health. I linked the tables using the student ID and used the SQL engine to run joins and group-by queries that revealed how certain groups of students differed. This part of the project reflected the relational database topics we discussed in class, where structured data storage and querying play a major role.

During exploratory analysis, I used Seaborn and Matplotlib to visualize patterns in the dataset. This included a correlation heatmap, a histogram of financial stress, and a scatterplot showing the relationship between sleep hours and stress levels across risk groups. These visual tools helped guide my understanding of which variables might be important predictors.

For the machine learning portion, I trained a Logistic Regression model and a Random Forest Classifier using an eighty to twenty train test split. I scaled the features for Logistic Regression and evaluated both models using accuracy, precision, recall, and confusion matrices. These evaluation methods aligned with the machine learning content covered in lectures.

2. Novelty and Importance

2.1 Why the Project Is Important

Mental health issues among college students are often difficult to identify early. Many students wait until they are overwhelmed before seeking help, and universities usually rely on voluntary reporting rather than predictive indicators. Being able to examine how academic pressure, sleep habits, grades, and stress levels relate to mental health risk can provide valuable insight for early intervention.

This project is important because it shows how accessible student-reported data can be used to highlight patterns worth paying attention to. If certain behaviors consistently appear among

high-risk students, universities could develop outreach programs or resources that target those areas. Even though this model cannot diagnose anything, the insights can still help guide conversations around student well-being.

2.2 Novelty of the Project

I was excited to work on this project because the topic is meaningful and directly connected to student life. It also allowed me to apply every part of the data science process we learned in class. Cleaning the dataset, building SQL tables, visualizing patterns, engineering features, and training models required me to combine skills from all parts of CS439. This made the project feel more complete and realistic than a typical homework problem.

It was also interesting to see how engineered features could strengthen the model. Creating sleep deficit or academic score added nuance that the raw dataset did not contain. This made me appreciate how feature engineering can shape the final outcome of a project.

2.3 Existing Issues in Data Science Practices

Mental health data is complicated because many influential factors are not included in structured datasets. Emotional states, personal experiences, and long-term history are usually missing. Self-reported data also introduces bias. Because of these challenges, models built from this kind of data rarely achieve perfect accuracy and must be interpreted carefully.

This project addresses some of these issues by focusing on transparency. The models used are interpretable and the feature engineering process is straightforward. While the dataset has limitations, it still helps demonstrate how academic and lifestyle factors relate to mental health outcomes.

2.4 Prior Related Work

Many studies examine how a single factor, such as GPA or stress, relates to mental health. This project differs because it incorporates multiple academic and lifestyle features into one integrated model. It also builds a complete data science pipeline similar to the sample projects provided for class, where data cleaning, SQL storage, visualization, and machine learning were all used together. This structure helps show how data science can contribute to understanding complex issues in practical settings.

3. Progress and Contribution

3.1 Data Utilization

The dataset included a variety of variables related to academics, lifestyle, and emotional indicators. CGPA, academic pressure, work pressure, study satisfaction, job satisfaction, financial stress, and work and study hours were all numerical. Categorical features included gender, city, dietary habits, degree type, sleep duration ranges, and suicidal thought history. The depression label indicated whether the student was considered at risk.

This diverse set of features made the dataset suitable for both relational storage and machine learning modeling. It also provided enough variation for meaningful exploratory analysis.

3.2 Data Cleaning

```
df = pd.read_csv("student_depression_dataset.csv")

df.columns = df.columns.str.lower().str.replace(" ", "_").str.replace("?", "")
df = df.rename(columns={"work/study_hours": "work_study_hours"})

numeric_cols = ["age", "academic_pressure", "work_pressure", "cgpa",
                "study_satisfaction", "job_satisfaction",
                "work_study_hours", "financial_stress"]

for col in numeric_cols:
    df[col] = pd.to_numeric(df[col], errors="coerce").fillna(df[col].median())

df = df.drop_duplicates()
```

To clean the dataset, I standardized column names, removed duplicates, and inspected each column for formatting issues. I converted numeric fields using Pandas and filled missing numeric values with the median. For categorical columns, I filled missing values using the most common category in each column.

One challenge was converting the sleep duration ranges into consistent numeric values. I assigned approximate hour values based on each range to create a usable sleep hours column. For binary indicators such as suicidal thoughts, I converted Yes and No into 1 and 0.

After cleaning, the dataset was consistent, structured, and ready for feature engineering, SQL storage, and machine learning.

3.3 Feature Engineering

```
sleep_map = {"less_than_5_hours": 4, "5-6_hours": 5.5,
             "7-8_hours": 7.5, "more_than_8_hours": 9}

df["sleep_hours"] = df["sleep_duration"].map(sleep_map).fillna(7)
df["sleep_deficit"] = 8 - df["sleep_hours"]

df["stress_level"] = df["financial_stress"]
df["stress_study_ratio"] = df["stress_level"] / (df["work_study_hours"] + 1)

df["academic_score"] = df["cgpa"] * (1 + df["academic_pressure"] / 5)
df["risk_label"] = df["depression"].astype(int)
```

Feature engineering played an important role in strengthening the predictive quality of the dataset. I created a sleep hours column and then used it to calculate sleep deficit, which measures how far a student falls below the recommended eight hours. I also created stress level based on financial stress and developed a stress to study ratio to capture how much stress the student experiences relative to their workload.

Academic score was another engineered feature that combined CGPA with academic pressure to reflect performance under stress. Converting suicidal thoughts into a binary flag added an important emotional indicator to the model. These engineered features helped reveal patterns that were not obvious from the raw dataset alone.

3.4 SQL Database Construction

```
import sqlite3
conn = sqlite3.connect(":memory:")

df[["id","gender","age","city"]].to_sql("demographics", conn, index=False)
df[["id","cgpa","academic_pressure"]].to_sql("academics", conn, index=False)
df[["id","sleep_hours","work_study_hours","stress_level"]].to_sql("lifestyle", conn, index=False)
df[["id","risk_label"]].to_sql("mental_health", conn, index=False)
```

I used SQLite to organize the cleaned dataset into relational tables. The demographics table stored id, gender, age, and city. The academics table included CGPA, academic pressure, and study satisfaction. The lifestyle table stored sleep hours, work and study hours, and stress levels. The mental health table stored the risk label.

Using SQL joins, I explored patterns such as average risk by gender. This part of the project followed the relational database concepts emphasized in class and helped make the dataset easier to analyze in a structured way.

```
query = """
SELECT gender, AVG(risk_label) AS avg_risk
FROM demographics
JOIN mental_health USING(id)
GROUP BY gender;
"""

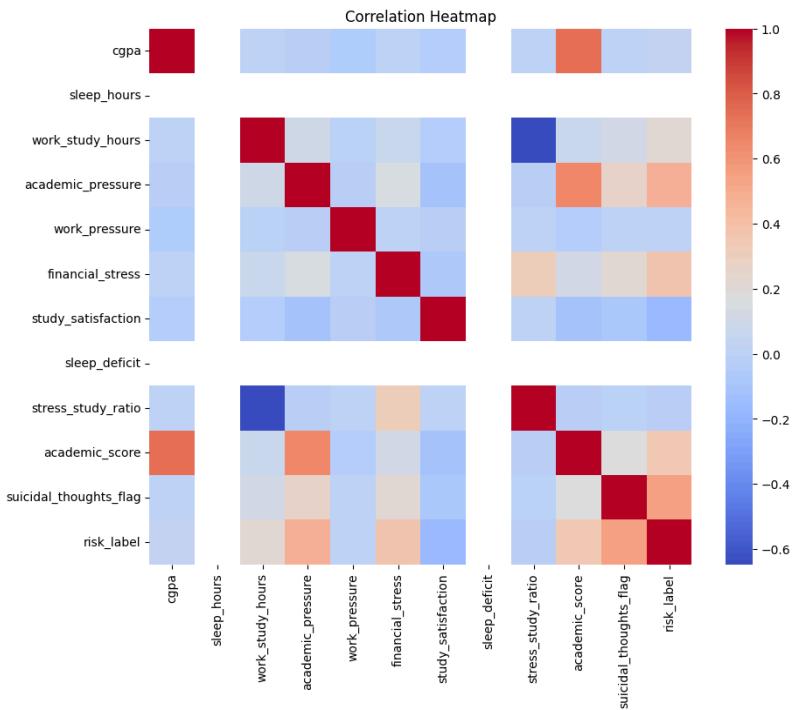
pd.read_sql_query(query, conn)
```

```
query = """
SELECT sleep_hours, AVG(risk_label)
FROM lifestyle JOIN mental_health USING(id)
GROUP BY sleep_hours;
"""

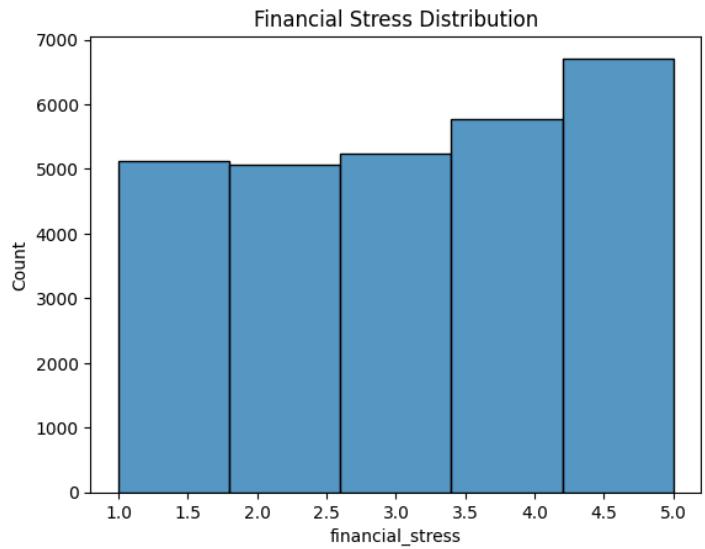
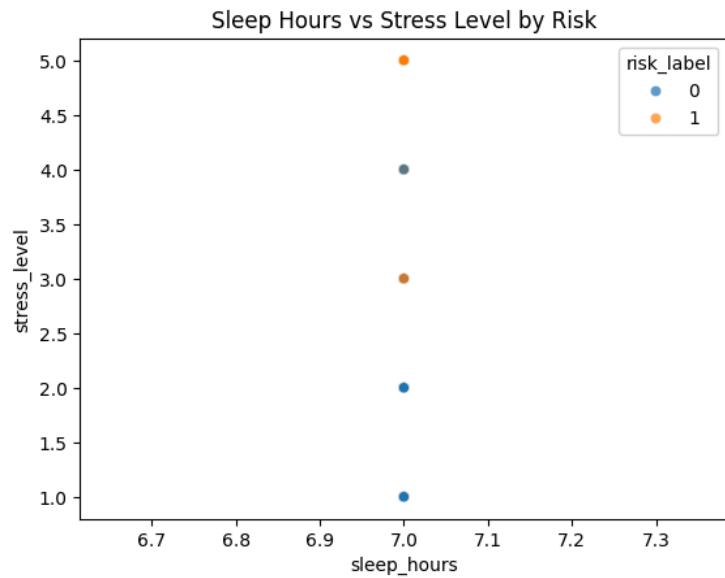
pd.read_sql_query(query, conn)
```

3.5 Exploratory Data Analysis

I used Matplotlib and Seaborn to explore trends in the dataset. The correlation heatmap showed that suicidal thoughts, academic score, CGPA, and academic pressure had the strongest relationships with the risk label. These insights guided the model building process.



I also created a histogram of financial stress and a scatterplot comparing sleep hours and stress levels, colored by risk group. These visualizations helped me understand the distribution of features and how lifestyle habits differ between low risk and high risk students.



3.6 Machine Learning Models and Techniques

```
log_reg = LogisticRegression(max_iter=1000)
log_reg.fit(X_train_scaled, y_train)

y_pred_lr = log_reg.predict(X_test_scaled)

print(accuracy_score(y_test, y_pred_lr))
print(confusion_matrix(y_test, y_pred_lr))
```

I trained two models: Logistic Regression and Random Forest Classifier. The dataset was split into training and testing sets with an eighty to twenty ratio. Logistic Regression required scaling, so I applied StandardScaler before training. This model is interpretable and works well when relationships are mostly linear.

The Random Forest Classifier does not require scaling and is capable of learning nonlinear patterns. It also provides feature importance scores, which helped me understand which engineered features contributed most to predicting risk.

```
rf = RandomForestClassifier(n_estimators=200, random_state=42)
rf.fit(X_train, y_train)

y_pred_rf = rf.predict(X_test)

print(accuracy_score(y_test, y_pred_rf))
print(confusion_matrix(y_test, y_pred_rf))
```

OUTPUTS

```
log_reg = LogisticRegression(max_iter=1000)
log_reg.fit(X_train_scaled, y_train)

y_pred_lr = log_reg.predict(X_test_scaled)

print("Accuracy:", accuracy_score(y_test, y_pred_lr))
print("Precision:", precision_score(y_test, y_pred_lr))
print("Recall:", recall_score(y_test, y_pred_lr))
print(confusion_matrix(y_test, y_pred_lr))

... Accuracy: 0.8331840172012184
Precision: 0.8445886169271601
Recall: 0.8763769889840881
[[1786  527]
 [ 404 2864]]
```

```
rf = RandomForestClassifier(n_estimators=200, random_state=42)
rf.fit(X_train, y_train)

y_pred_rf = rf.predict(X_test)

print("Accuracy:", accuracy_score(y_test, y_pred_rf))
print("Precision:", precision_score(y_test, y_pred_rf))
print("Recall:", recall_score(y_test, y_pred_rf))
print(confusion_matrix(y_test, y_pred_rf))

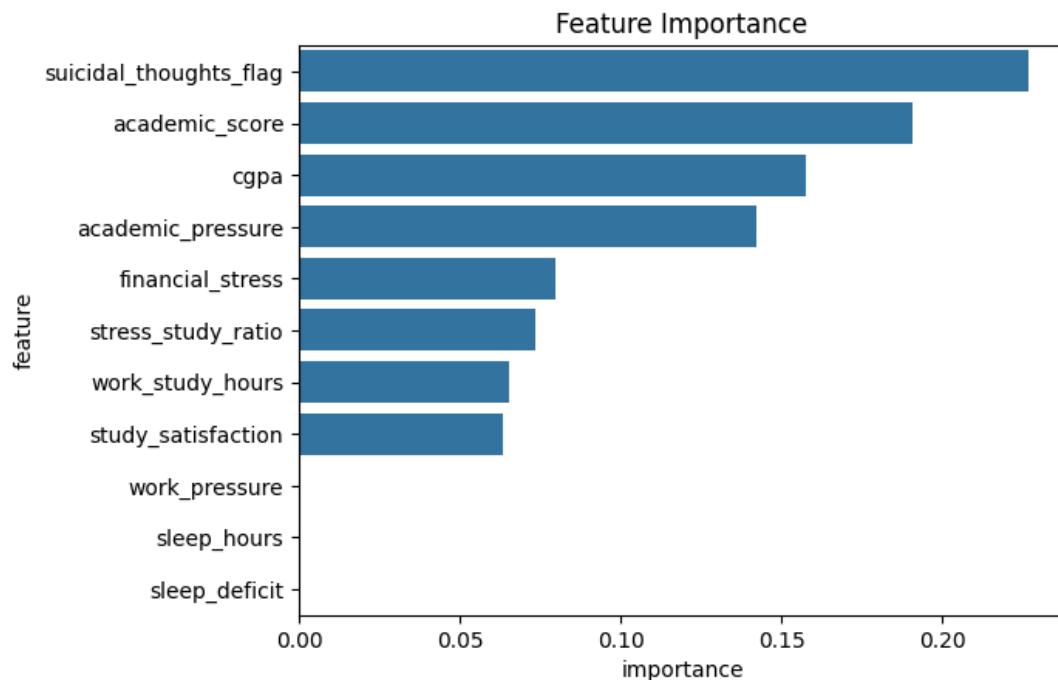
Accuracy: 0.7989607597204802
Precision: 0.8265368228849665
Recall: 0.8310893512851897
[[1743  570]
 [ 552 2716]]
```

3.7 Key Findings and Results

```
feat_imp = pd.DataFrame({  
    "feature": feature_cols,  
    "importance": rf.feature_importances_  
}).sort_values("importance", ascending=False)  
  
sns.barplot(data=feat_imp, x="importance", y="feature")  
plt.show()
```

Logistic Regression performed the best overall. It achieved an accuracy of about eighty three percent and had strong precision and recall values. I originally expected Random Forest to outperform it, but the opposite happened. The Random Forest model still performed reasonably well and was useful for interpreting feature importance.

According to the Random Forest feature ranking, suicidal thoughts had the highest influence on risk prediction. Academic score and CGPA were also important, along with academic pressure and financial stress. These findings matched what the visualizations showed earlier.



3.8 Evaluation

```
results = pd.DataFrame({  
    "Model": ["LR", "RF"],  
    "Accuracy": [lr_accuracy, rf_accuracy]  
})  
  
results
```

I evaluated both models using accuracy, precision, recall, and confusion matrices. Logistic Regression showed the strongest performance and demonstrated good reliability for this dataset. Its high recall was especially valuable since the goal is to identify at-risk students correctly. Random Forest helped confirm which variables influenced the predictions most.

Together, these evaluations showed that even a relatively simple model can perform well when supported by thoughtful preprocessing and feature engineering.

3.9 Advantages and Limitations

One advantage of this approach is that the dataset is structured and includes a mix of academic and lifestyle variables. Feature engineering added meaningful depth to the analysis, and the SQL component helped organize information clearly. The models also reached solid performance without requiring overly complex techniques.

A limitation is that the dataset relies on self-reported information, which may not always be accurate. Important emotional or personal variables are missing, and the model cannot capture the full complexity of mental health. Because of these limitations, the predictions should be interpreted carefully and not used for diagnosis.

4. Changes After Proposal

4.1 Differences from the Proposal

In my proposal, I assumed the dataset would include anxiety indicators, but it did not. I adjusted by focusing on available variables and creating engineered features that helped make the model more expressive. I also expected the Random Forest model to perform best, but Logistic Regression outperformed it, which changed how I interpreted the results.

4.2 Bottlenecks and Challenges

One challenge was converting categorical ranges into usable numeric formats, especially for sleep duration. Another challenge was deciding which engineered features would add meaningful insight without making the model unnecessarily complex. Structuring the SQL tables also took planning, since I wanted them to follow a clear relational format similar to the sample reports.

Balancing interpretability and complexity in the modeling stage also required careful decisions, especially since different models behaved differently than expected.

5. Conclusion and Future Work

5.1 Summary of Contributions

This project explored how academic habits, lifestyle choices, and emotional factors relate to mental health risk. This project completed a full data science workflow. I cleaned and transformed the dataset, engineered new features, created SQL tables, analyzed patterns through visualizations, and trained machine learning models. The results showed that Logistic Regression performed best and that features such as suicidal thoughts, academic score, CGPA, and academic pressure were the strongest predictors of mental health risk.

The project demonstrates how data science can be used to study student well-being and how academic and lifestyle habits can relate to mental health risk.

5.2 Future Directions

Future work could involve using a larger and more diverse dataset that includes additional emotional and behavioral variables. More complex models such as Gradient Boosting or Neural Networks could also be explored. Another idea would be to build an interactive tool or dashboard where students or advisors could input values and see predicted risk levels.

Although the model cannot diagnose mental health conditions, the project shows how data-driven approaches can help identify patterns that support earlier outreach and better student wellness planning.