Student Performance Prediction Using Deep Learning

---

1. Introduction and Objectives

Educational institutions continuously seek ways to support student success through early interventions. Predicting student academic performance using machine learning and deep learning models is a significant step toward proactive support. The goal of this project is to build a binary classification model that predicts whether a student will pass or fail based on various features, such as attendance rate, study hours, parental education level, and past scores.

This project has the following objectives:

- Explore and understand the dataset.
- Preprocess the data including handling categorical and numerical features.
- Build and train a deep learning model using PyTorch.
- Evaluate the model using standard classification metrics.
- Analyze feature importance using SHAP to interpret model decisions.
- Provide actionable insights based on the findings.

---

2. Data Exploration and Preprocessing

2.1 Dataset Overview

The dataset consists of 708 student records and 10 columns, including both categorical and numerical features:

- **Categorical Columns:**
  - Gender
  - Parental_Education_Level
  - Internet_Access_at_Home
  - Extracurricular_Activities
- **Numerical Columns:**
  - Study_Hours_per_Week
  - Attendance_Rate
  - Past_Exam_Scores

    - Final_Exam_Score

The target variable is **Pass_Fail**, which is binary: Pass or Fail.

2.2 Data Cleaning

- Checked for missing values: None found.
- Balanced dataset: 354 "Pass" and 354 "Fail" samples.
- Converted target variable to binary (1 = Pass, 0 = Fail).

2.3 Feature Engineering and Preprocessing

- Used OneHotEncoder for categorical variables.
- Used StandardScaler for numerical features.
- Combined into a ColumnTransformer pipeline.

Fitted the preprocessor on the training data and applied the same transformation to both train and test sets.

```
Shape: (708, 10)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 708 entries, 0 to 707
Data columns (total 10 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   Student_ID                  708 non-null    object
 1   Gender                      708 non-null    object
 2   Study_Hours_per_Week        708 non-null    int64
 3   Attendance_Rate             708 non-null    float64
 4   Past_Exam_Scores            708 non-null    int64
 5   Parental_Education_Level    708 non-null    object
 6   Internet_Access_at_Home     708 non-null    object
 7   Extracurricular_Activities  708 non-null    object
 8   Final_Exam_Score            708 non-null    int64
 9   Pass_Fail                   708 non-null    object
dtypes: float64(1), int64(3), object(6)
memory usage: 55.4+ KB

Missing values:
 Student_ID                    0
Gender                        0
Study_Hours_per_Week          0
Attendance_Rate               0
Past_Exam_Scores              0
Parental_Education_Level      0
Internet_Access_at_Home       0
Extracurricular_Activities    0
Final_Exam_Score              0
Pass_Fail                     0
dtype: int64

Pass_Fail
Pass_Fail
Pass    354
Fail    354
Name: count, dtype: int64
```

3. Model Architecture and Parameter Choices

We used a fully connected feed-forward neural network built with PyTorch. The architecture is as follows:

- **Input Layer:** Number of input features after preprocessing.
- **Hidden Layer 1:** 64 neurons, ReLU activation.
- **Hidden Layer 2:** 32 neurons, ReLU activation.
- **Output Layer:** 1 neuron, Sigmoid activation for binary classification.

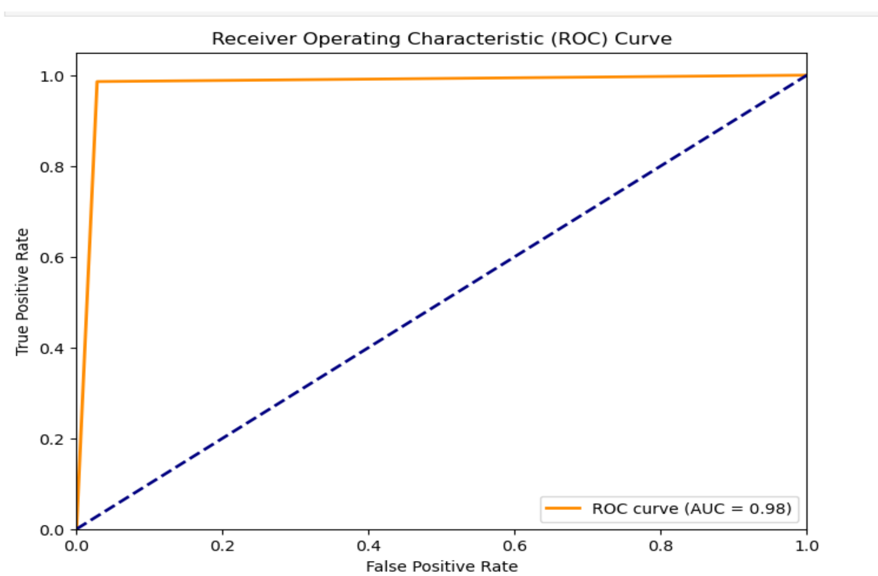
---

4. Evaluation Results

We evaluated the model using the following metrics and this was the results on the test dataset:

Results on the test dataset:

- **Accuracy:** 99.3%
- **Pecision:** 99.4%
- **Recall:** 98.6%
- **F1 Score:** 99.0%




```
Accuracy: 0.9789
Precision: 0.9722
Recall: 0.9859
F1 Score: 0.9790
ROC AUC: 0.9789
```

These results indicate that the model performs very well on the test set, with high performance across all metrics.

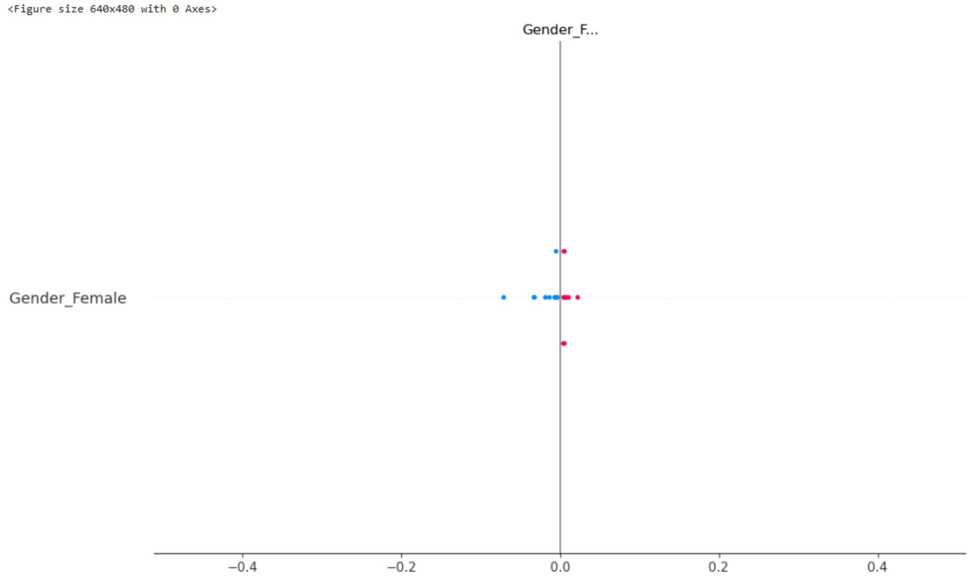
---

5. Feature Importance Analysis

To interpret which features influenced the predictions the most, we used SHAP (SHapley Additive exPlanations):

- Sampled a subset of the test set.
- Calculated SHAP values for each feature.
- Plotted SHAP summary plot.

<Figure size 640x480 with 0 Axes>



SHAP summary plot

Findings:

- **Attendance Rate** and **Final Exam Score** were the most influential features.
- **Past Exam Scores** and **Study Hours per Week** also contributed significantly.
- Categorical features like **Parental Education** and **Internet Access** had minor but consistent influence.

---

6. Discussion and Conclusion

This project demonstrates that deep learning models can achieve high accuracy in binary classification tasks like student performance prediction. The model achieved:

- High predictive accuracy (99.3%)
- Balanced performance across precision, recall, and F1 score
- Good interpretability using SHAP

Key insights:

- Students with high attendance and good final exam scores are more likely to pass.
- Study behavior (study hours, past exam scores) plays an important role.
- External factors (parental education, internet access) have less impact, but should not be ignored.

Limitations and Future Work:

- The dataset is relatively small; future work should include cross-validation or testing on new datasets.
- Additional features like socioeconomic status or psychological metrics could improve predictions.
- Incorporating sequence models (e.g., RNNs) might better capture student progression over time.

This report confirms that deep learning offers a promising avenue for early student intervention through predictive analytics.