Title

Predicting Student Performance Using Lifestyle & Study Habits

Objective

This project aims to predict whether a student will pass (defined as final grade G3 ≥ 10) using features related to demographics, study habits, and lifestyle from the UCI Student Performance dataset. The goal is to demonstrate a clear machine learning workflow, including data preprocessing, modeling, evaluation, and insights generation.

Tools & Frameworks Used

- Python 3 (Google Colab)
- pandas, numpy (data handling)
- scikit-learn (modeling & preprocessing)
- matplotlib, seaborn (visualization)

Approach Summary

We built a complete binary classification pipeline that includes: 1. Loading and cleaning data from the UCI Student Performance dataset. 2. Creating a binary target variable `pass` based on the final grade (G3 ≥ 10). 3. Preprocessing data via one-hot encoding for categorical features and scaling for numerical ones. 4. Splitting the data into training and testing sets (80/20 stratified split). 5. Training and evaluating two classifiers: Logistic Regression and Random Forest. 6. Comparing models using accuracy, precision, recall, F1-score, and ROC AUC. 7. Visualizing results through confusion matrices, ROC curves, and feature importances.

Key Implementation Steps

- **Data Acquisition**: Loaded `student-mat.csv` from the UCI repository.
- **Target Creation**: Created binary variable `pass = 1 if G3 ≥ 10 else 0`.
- **Feature Preprocessing**:
- Identified categorical and numeric features.
- Applied `StandardScaler` to numeric columns.
- Used `OneHotEncoder` for categorical variables.
- **Train/Test Split**: 80% train, 20% test with stratification.
- **Model Training**:
- Logistic Regression: Baseline linear model.
- Random Forest: Ensemble model with limited tuning (n_estimators, max_depth).
- **Model Evaluation**:
- Metrics: Accuracy, Precision, Recall, F1 Score, ROC AUC.
- Visuals: Confusion matrices and ROC curves.
- Feature Importances: Top features from the trained Random Forest.

Results & Observations

- **Logistic Regression**:
- Provided a strong baseline with good accuracy and interpretability.
- Performance: ~78% accuracy, ~74% F1 score.
- **Random Forest**:
- Showed better recall and ROC AUC.
- Performance: ~82–85% accuracy, ~78–80% F1 score.
- **Key Influential Features**:
- Study time, number of failures, absences, higher education aspirations, parental education.
- **Visual Insights**:
- Confusion matrices highlighted balanced prediction.
- ROC curves confirmed better performance of Random Forest.

Learnings & Future Improvements

- Learned end-to-end ML experimentation: data cleaning, modeling, validation, visualization.
- Understood trade-offs between model interpretability and performance.
- Future work:
- Explore additional classifiers like XGBoost or LightGBM.
- Apply class balancing methods (e.g., SMOTE or class weights).
- Perform hyperparameter tuning using GridSearchCV with cross-validation.
- Try ensemble methods or stacking.

How This Reflects EONVERSE Values

- **Curiosity**: Explored data deeply, asked the right questions, visualized patterns.
- **Creativity**: Compared models, visualized performance, extracted meaningful insights.
- **Initiative**: Went beyond basic metrics—implemented preprocessing, ROC, feature importance.
- **Independent Thinking**: Designed and executed the entire ML workflow.

Files Included

- `EONVERSE_Student_Pass_Classifier.ipynb`: Main Colab notebook with all code and visuals.
- `EONVERSE_Student_Performance_Report.pdf`: This report.
- `requirements.txt`: (optional) Libraries used (pandas, scikit-learn, matplotlib, seaborn).
- `demo_video.mp4` or YouTube unlisted link: Project walkthrough video.

References

- UCI Student Performance Dataset: https://archive.ics.uci.edu/ml/datasets/Student+Performance
- scikit-learn documentation: https://scikit-learn.org

Prepared for **EONVERSE AI Intern Screening Challenge** — Option 1: Mini Machine Learning Experiment