Student Performance Analysis Using Machine Learning

# Executive Summary

The Student Performance Analysis project applies machine learning techniques to understand and improve academic outcomes. Using a user-friendly Streamlit interface, the application facilitates visualization, supervised predictions, and unsupervised pattern discovery, making it a powerful tool for educators, researchers, and policymakers.

# Project Introduction

# Problem Statement

Identifying and understanding the factors that influence academic performance is essential for improving education systems. This project uses ML to reveal data-driven insights from student profiles and test scores.

# Objectives

* Develop an intuitive application for education data analysis
* Predict academic success using classification and regression models
* Reveal hidden insights via clustering and association rule mining
* Provide interpretive visualizations and model evaluation tools

# Dataset Description

* Gender, race/ethnicity, and parental education
* Lunch status and test preparation
* Math, reading, and writing scores

# System Architecture

The architecture includes separate components for data handling (`utils.py`), visualization (`visualization.py`), supervised learning (`supervised.py`), unsupervised learning (`unsupervised.py`), and the main interface (`app.py`).

# Data Exploration and Visualization

* Descriptive statistics and frequency distributions
* Distribution and box plots for performance metrics
* Heatmaps, scatter plots, and feature comparison charts

# Supervised Learning

# Classification Models

* Decision Tree, Random Forest, SVM, Logistic Regression, KNN
* Evaluation metrics: Accuracy, Precision, Recall, F1-Score
* Feature importance and confusion matrix

# Regression Models

* Linear, Decision Tree, Random Forest, SVR
* Evaluation metrics: RMSE, MAE, R² score
* Actual vs. predicted visualizations

# Unsupervised Learning

# Clustering

* K-Means, Hierarchical clustering
* Cluster count selection and visualization using PCA
* Silhouette score and Elbow method

# Association Rule Mining

* Apriori algorithm, rule filtering by lift, support, confidence
* Network graph visualization of associations

# Implementation Details

* Streamlit for UI, Scikit-learn for ML, Seaborn/Plotly for visualizations
* MLxtend for association mining, Statsmodels for advanced stats
* Modular codebase, dynamic UI elements, robust type handling

# Results and Insights

* Parental education, preparation courses, and gender significantly affect scores
* Classification accuracy reached 85% for pass/fail prediction
* Cluster analysis revealed performance trends across subgroups
* Association rules showed strong links between education level and high scores

# Challenges and Solutions

* Streamlit ID duplication → Resolved via unique widget keys
* Scatter plot issues → Fixed column mismatches
* Serialization errors → Converted data types appropriately

# Future Enhancements

* Deploy trained models via API endpoints
* Enable time-series and NLP analytics on student feedback
* Role-based dashboards with export capabilities
* Incorporate deep learning and ensemble techniques

# Conclusion

This project highlights how machine learning can transform raw educational data into actionable insights. The user-friendly Streamlit app empowers non-technical users to explore, predict, and discover patterns in student performance.

# Appendix

# Usage Instructions

* Install dependencies: pip install -r requirements.txt
* Run app: streamlit run app.py
* Access on browser: http://localhost:8501

# Key Files

* app.py – Main application entry point
* supervised.py – Contains classification/regression logic
* unsupervised.py – Clustering and association mining
* visualization.py – Exploratory data analysis and plotting
* utils.py – Data loading and preprocessing helpers