Group: ClassPass
Members: Phil Akagu-Jones (paa46@sfu.ca)
& Ariel Tyson (ajt11@sfu.ca)

Milestone 2 Progress Report

Project Summary

Our project aims to predict whether a student can pass a class or is at risk of dropping out using structured data from the UCI Predict Students' Dropout and Academic Success dataset. We are implementing our models from scratch, focusing on a custom kNN classifier and our goal is to make everything easily interpretable so students in need of extra support can be flagged.

Significant Accomplishments

Accomplishment 1 — Data Pipeline + EDA

We have implemented a loading and preprocessing pipeline that can handle the UCI data and structure it properly for our use. Our EDA generates: numeric summaries, frequency tables, class distribution plots and JSON with the data in a clear format.

Accomplishment 2 — Custom kNN Classifier

We have also implemented a custom kNN classifier, not using scikit. Ours can: compute euclidean and manhattan distances, provide neighbour based explanations, provide input validation, and most importantly easily integrates with our preprocessing pipeline.

Accomplishment 3 — Evaluations + Visualizations

Lastly, we also have implemented a full evaluation module that: computes accuracy and F1 scores, generates confusion matrices, plots F1 vs k, and saves all metrics and plots locally.

Proof of Accomplishments

All proof is in the attached zip files.

Proof 1: These are generated files containing the numeric statistics, categorical distributions and class balances. This confirms that the dataset was successfully cleaned, parsed, and analyzed.

Proof 2: This verifies the custom kNN model trains correctly, evaluates on validation sets, and performs hyperparameter selection. All of which is implemented by us.

Group: ClassPass
Members: Phil Akagu-Jones (paa46@sfu.ca)
& Ariel Tyson (ajt11@sfu.ca)

Proof 3: These validate the entire pipeline functions, including preprocessing, modeling, evaluation and plotting.

Challenges & Roadblocks

Many Categorical Features

Some features, like qualification or occupation, forced us to spend a lot of time one-hot encoding to avoid future problems. As this was not expected, we did not account any time for this and it slightly set us behind our original timeline.

Class Imbalance

An issue that we found with the data is that dropout cases are quite underrepresented, so this caused us to have to carefully validate our training to avoid overfitting which we have been able to fix.

Testing Our Custom Model

The issue that probably took up the most of our time was ensuring that our custom kNN classifier performed as expected which caused us to carefully unit test many outputs to ensure it did. This took us many iterations, but we believe we have it working now.

Changes from Original Plan

Nothing major has changed in terms of our plans besides adding more comprehensive testing.