Course 201
6.3 Mini-project
Date: July 2024

Predicting Student Dropout with Supervised Learning

By Joshua Dixon

Executive Summary

This report analyses student data to predict dropout rates using XGBoost and a neural network model. The most signidicant features were:

- **Credit Weighted Average:** Lower scores increase dropout likelihood.
- **Centre Name:** Location impacts dropout rates, possibly due to varying resources, support, and demographics.

Recommendations to Study Group

- **Enhance Academic Support:** Improve academic performance through tutoring, study groups, and support programs.
- **Strengthen Centre-Specific Support:** Address unique challenges of each centre, focusing on those like ISC Kingston, and provide tailored support services.
- **Monitor and Share Best Practices**: Regularly evaluate performance across centres, identify best practices, and standardise them to improve retention uniformly.

By implementing these targeted actions, Study Group can enhance student retention rates, ensuring students receive the support needed to succeed academically and remain enrolled.

Table of Contents

1 Introduction

High dropout rates can lead to revenue loss, diminished reputation, and lower satisfaction. This document outlines using XGBoost and a neural network with TensorFlow and Keras to predict student dropout.

Methodology

1.1 Data Source

The dataset has 25,060 rows, each representing a learner, including details such as nationality and performance. The dataset, used for the entire course, contains 36 features.

1.2 Key Steps in Analysis

1.2.1 The analysis was conducted on two datasets per model:

- X1 dataset: Original key features.
- X2 dataset: Original key features plus 'AttendancePercentage' and 'ContactHours'.

1.2.2 Methodology:

1) Initial Data Exploration and Assessment:
    a. Identified key features, data cleaning, feature engineering, scaling, and one-hot encoding.
    b. Split independent features into Train, Validation, and Test Datasets.
    c. Created X1 and X2 datasets.
2) Predicting Dropout with XGBoost:
    a. Modeled and compared X1 and X2 datasets with both default parameters and hyperparameter-tuned models.
3) Predicting Dropout with Neural Networks:
    a. Modeled and compared X1 and X2 datasets with both default parameters and hyperparameter-tuned models.

Key Libraries Used: Pandas, NumPy, Matplotlib, Seaborn, Scikit-learn, XGBoost, TensorFlow, Keras.Hyperparameter Tuning

1.2.3 Hyperparameter Tuning

F1-Score metric was used in hyper-parameter tuning, given the data is imbalanced and accuracy is less reliable.

- XGBoost Default Parameters:
    - learning_rate: 0.3
    - max_depth: 6
    - n_estimators: 100
- Model X1: XGBoost Best Tuned Parameters:
    - learning_rate: 0.1
    - max_depth: 7
    - n_estimators: 100
- Model X2: XGBoost Best Tuned Parameters:
    - learning_rate: 0.2
    - max_depth: 7
    - n_estimators: 300

- NN Initial Parameters:
    - neurons: 64
    - activation: relu
    - optimiser: adam
- Model X1: NN Best Tuned Parameters:
    - neurons: 128
    - activation: tanh
    - optimiser: adam
- Model X2: NN Best Tuned Parameters:
    - neurons: 64
    - activation: relu
    - optimiser: adam

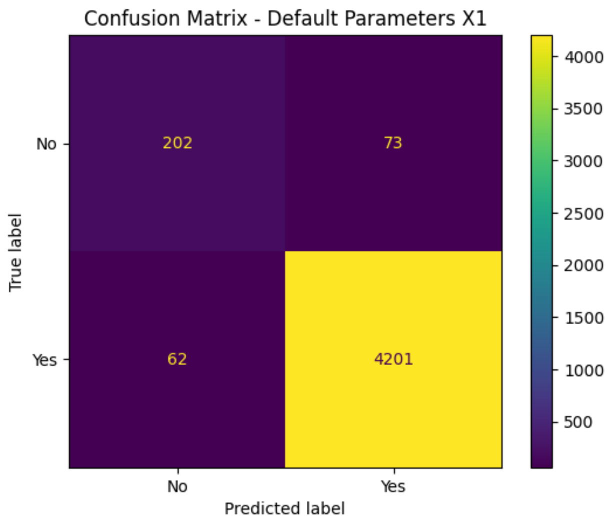
2 Predicting dropout with XGBoost

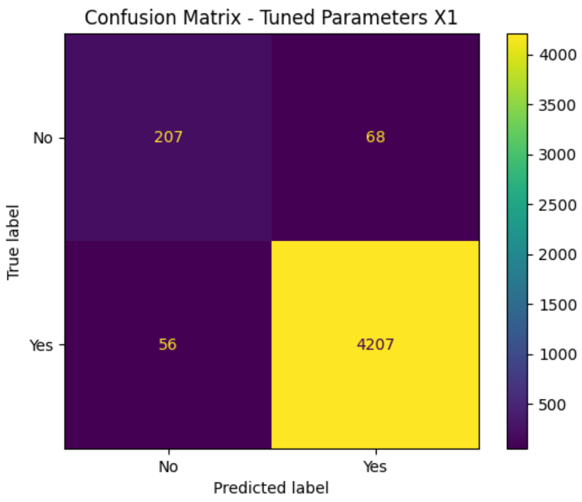
2.1 Results Summary

Model	Accuracy	Precision (No)	Recall (No)	F1-Score (No)	Precision (Yes)	Recall (Yes)	F1-Score (Yes)	AUC Score
Default Parameters X1	0.970251	0.765152	0.734545	0.749536	0.982920	0.985456	0.984186	0.974265
Tuned Parameters X1	0.972675	0.787072	0.752727	0.769517	0.984094	0.986864	0.985477	0.978470
Default Parameters X2	0.982151	0.859259	0.843636	0.851376	0.989925	0.991086	0.990505	0.990208
Tuned Parameters X2	0.981490	0.865900	0.821818	0.843284	0.988543	0.991790	0.990164	0.989515

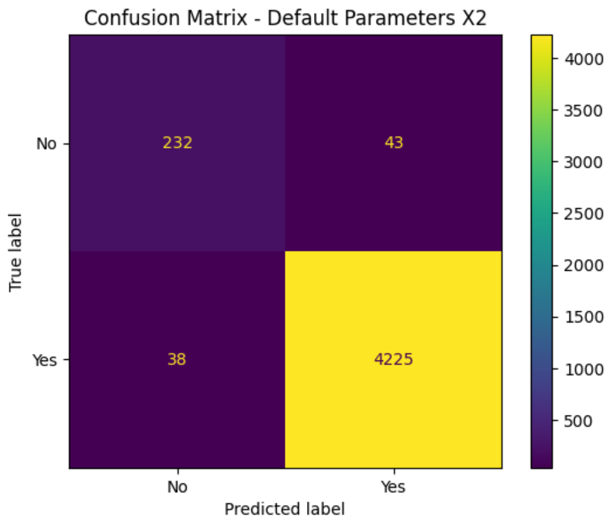
2.2 Confusion Matrices

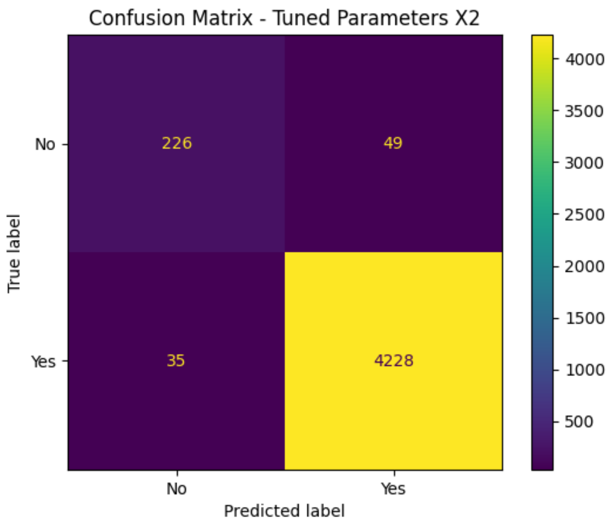
2.2.1 X1 Data


Confusion Matrix - Default Parameters X1

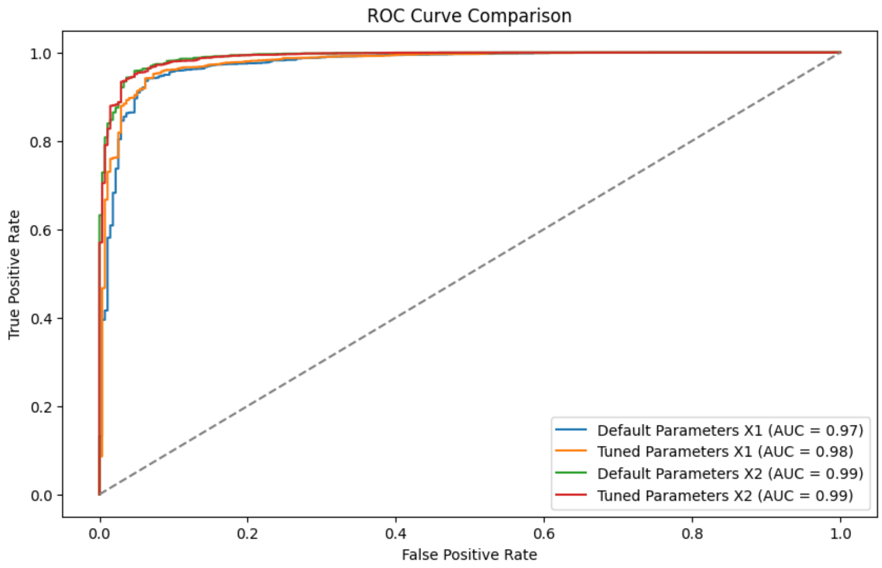

Confusion Matrix - Tuned Parameters X1

2.2.2 X2 Data

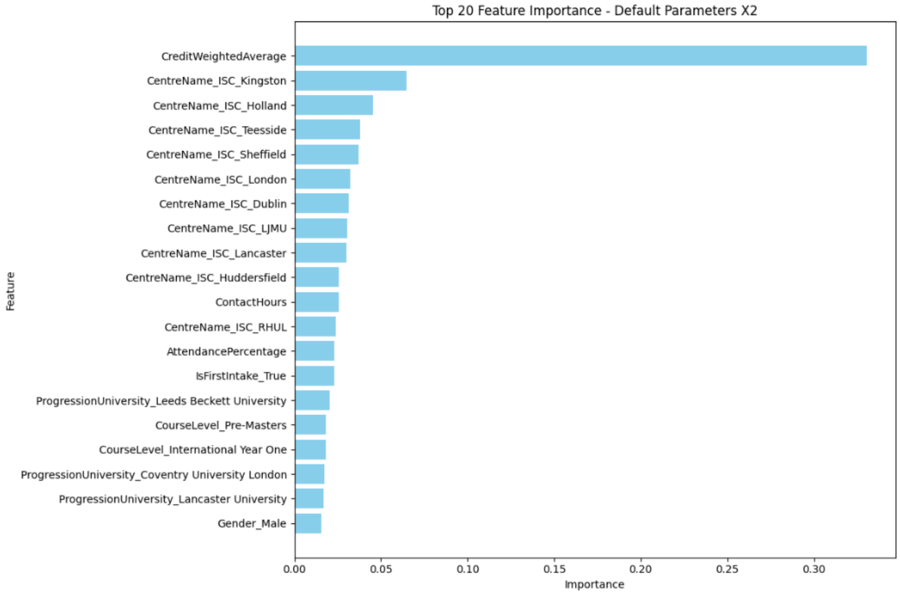

Confusion Matrix - Default Parameters X2


Confusion Matrix - Tuned Parameters X2

2.3 ROC Curves



2.4 Feature Importance



2.5 Overall Interpretation

- Confusion matrices show that the data is imbalanced.

- The XGBoost model with default parameters shows high accuracy for positive values, with over 98% for Precision, Recall, and F1-Score, likely due to data imbalance. Negative values were less accurate with 75% for the default X1.

- The ROC curve shows an AUC scores above 0.97, indicating near-perfect distinction between positive and negative cases, though influenced by the data imbalance, as false positives significantly impact the fewer negative observations.

- Tuned models improved Precision, Recall, and F1 Score for the "No" category by approximately 1.5-2%, while the "Yes" category saw minimal improvement for X1. It is noted that F1-Score slightly decreases for X2 data, but the difference is neglible.

- Adding features like AttendancePercentage and UnauthorisedAbsenceCount notably improved metrics for the "No" category, highlighting their importance in better predicting dropout rates.

- CreditWeighted Average was significantly more important than the remaining features. CentreName was second, with ISCKingston having the greated impact.

3 Predicting dropout using Neural Networks

3.1 Results Summary

Table 1: Results for "Yes" Category

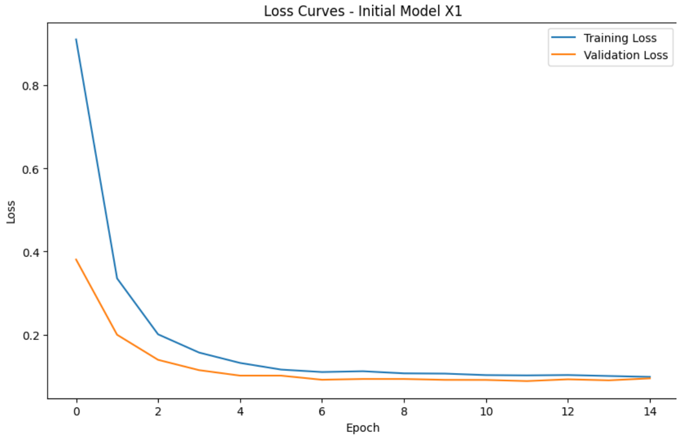
Model	Loss	F1-Score	Accuracy	Precision (Yes)	Recall (Yes)	F1-Score(Yes)	AUC
Initial Model X1	0.091680	0.983564	0.969590	0.982456	0.985222	0.983837	0.975841
Tuned Model X1	0.090244	0.982940	0.968488	0.982662	0.983814	0.983238	0.975572
Initial Model X2	0.079803	0.986762	0.975540	0.982791	0.991321	0.987037	0.983465
Tuned Model X2	0.072473	0.987543	0.977082	0.985978	0.989679	0.987825	0.985975

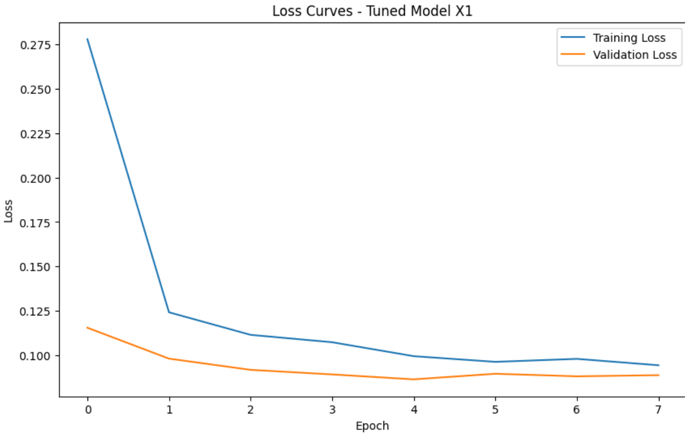
Table 2: Results for "No" Category

Model	Loss	F1-Score	Accuracy	Precision (No)	Recall (No)	F1-Score (No)	AUC
Initial Model X1	0.091680	0.983564	0.969590	0.767544	0.636364	0.695825	0.975841
Tuned Model X1	0.090244	0.982940	0.968488	0.744444	0.730909	0.737615	0.975572
Initial Model X2	0.079803	0.986762	0.975540	0.817460	0.749091	0.781784	0.983465
Tuned Model X2	0.072473	0.987543	0.977082	0.814126	0.796364	0.805147	0.985975

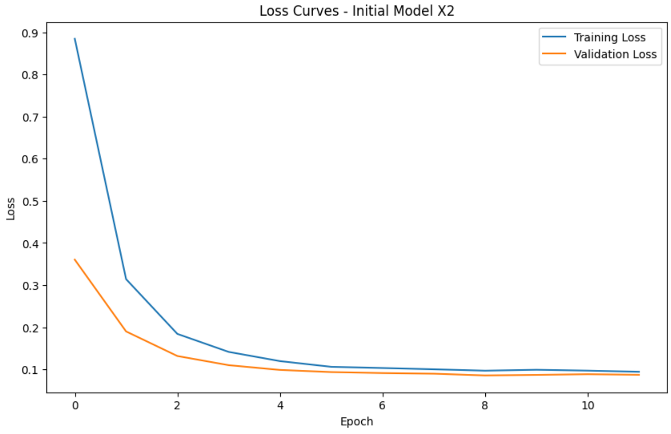
3.2 Loss Curves

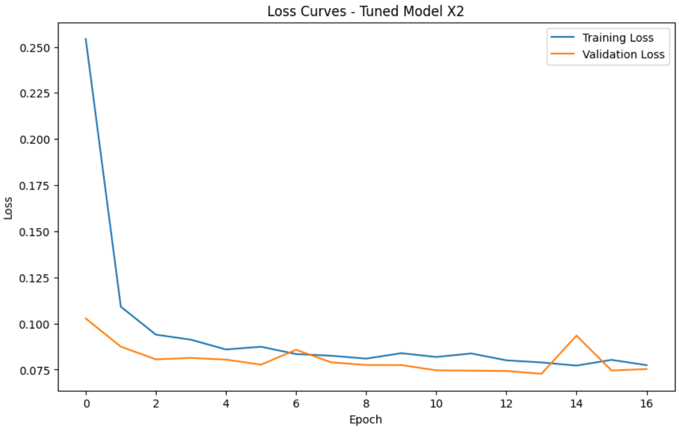
3.2.1 X1 Data





3.2.2 X2 Data





3.3 Overall Interpretation

- The loss curves show a reduction in both training and validation loss as epochs increase.
- Early stopping led to a varying number of epochs between models.
- Hyperparameter tuning did not significantly impact any of the models, with only slight improvement. The initial parameters appeared to produce optimal results for Model X2. Variation in the results is likely due to inherent randomness and the higher number of epochs in the tuned model.
- Adding features like AttendancePercentage and ContactHours has improved the performance metrics across both initial and tuned models.
- Initial curves appear to be smoother than hyperparameter-tuned curves.
- AUC results are similar to XGBoost model

4 Conclusion

- XGBoost generally outperformed the Neural Network, with final results similar after hyperparameter tuning and inclusion of ‘AttendancePercentage’ and ‘ContactHours’. The maximum F1-Scores for the “No” category were 0.85 (XGBoost) and 0.81 (Neural Network). Both models performed well for the “Yes” category, though the impact of imbalance was noted.
- Hyperparameter tuning alone had minimal impact on performance, with slight positive effects, particularly for the “No” category. The inclusion of ‘AttendancePercentage’ and ‘ContactHours’ had a greater isolated impact.
- CreditWeightedAverage and CentreName for specific centres were the most important features, prioritised in recommendations.