Supervised Learning: Predicting Student Dropout

1 Introduction

This project aims to identify students at risk of dropping out of their courses using two supervised learning techniques: XGBoost and Neural Networks. Predictions are made at three key stages of a student's academic journey (early, mid, and late), with each stage leveraging distinct metrics to enhance predictive accuracy. Accurately identifying at-risk students enables Study Group to implement timely and tailored interventions, providing effective support to those in need.

The primary objective of this project is to evaluate how the effectiveness of classifiers evolves as more student information becomes available over time. By analyzing predictive performance across the three stages, the project aims to determine when interventions are most impactful in preventing student dropouts.

2 Data Overview & Preprocessing

Study Group's dataset contains student information such as nationality, academic details, and gender, with additional data incorporated at each stage to enhance predictive accuracy. Stage 1 includes basic categorical and demographic features related to students and their courses. Stage 2 adds attendance data through authorised and unauthorised absence counts, while Stage 3 introduces academic performance metrics through module pass rates.

Data preprocessing involved removing irrelevant features, such as identifiers and marketing information, as they offered little predictive value. Features with over 50% missing data were excluded, while those with less than 2% missing values were cleaned by removing affected rows. This approach was particularly relevant in Stage 2, where some classes without attendance requirements were omitted. High-cardinality categorical variables with over 200 unique values were removed to prevent model overfitting and inefficiency. However, *Nationality* and *Course Name*, despite their high cardinality, were retained due to their predictive importance and encoded using *feature embedding* in the Neural Network to avoid the dimensionality explosion of one-hot encoding. This method assigns each category a dense vector, enabling the model to capture meaningful relationships between categories (see figure 1). After preprocessing, the dataset contained between **24,851 and 25,059 entries** and **8 to 12 features**, depending on the stage.

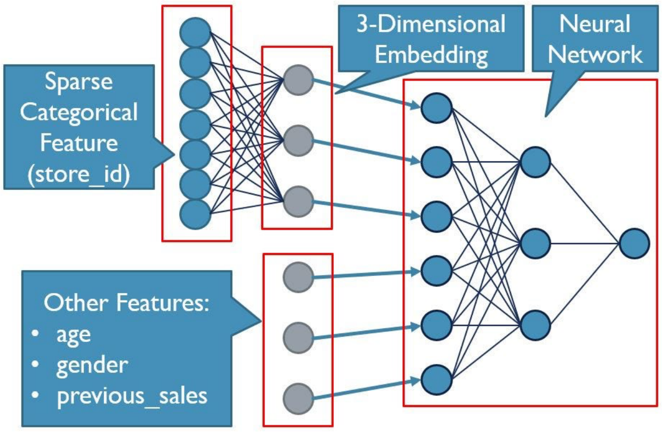

Figure 1: Feature Embedding (Malin, 2025)

3 Methodology

The models selected for this analysis (XGBoost and Neural Networks) were chosen for their complementary strengths. XGBoost effectively utilizes encoded categorical data without imposing artificial ordinal relationships, when appropriate encoding is applied. This method also enables feature importance analysis, offering valuable insights into both classification reliability and key indicators within student profiles. In contrast, Neural Networks excel at capturing complex, non-linear feature interactions, often uncovering patterns that simpler models may overlook, making them effective for nuanced classification tasks.

NEURAL NETWORK			
Hyperparameter	Stage 1	Stage 2	Stage 3
Embedding Dimension **(5-25; step size 5)**	10	25	15
Number of nodes (layer 1) **(16-128; step size 16)**	48	32	16
Activation (layer 1) **(ReLU, tanh, leaky ReLU)**	Tanh	ReLU	Leaky ReLU
Dropout layer 1 (rate) **(0.2-0.5; step size 0.1)**	0.2	0.3	0.4
Number of nodes (layer 2) **(16-128; step size 16)**	128	96	64
Activation (layer 2) **(ReLU, tanh, leaky ReLU)**	ReLU	ReLU	Leaky ReLU
Dropout layer 2 (rate) **(0.2-0.5; step size 0.1)**	0.4	0.2	0.4
Optimiser **(Adam, SGD, RMSProp)**	RMSProp	RMSProp	Adam

XGBOOST			
Hyperparameter	Stage 1	Stage 2	Stage 3
Learning Rate **(0.01, 0.1, 0.2)**	0.1	0.1	0.2
Max depth **(3, 5, 7, 10)**	5	5	3
# of Estimators **(50, 100, 200)**	50	100	200

Initially, both models were trained with preliminary hyperparameters selected through trial and error to achieve satisfactory baseline results. To enhance performance, hyperparameter tuning was conducted using distinct methods tailored to each model's computational characteristics. XGBoost, being relatively faster to train and having fewer hyperparameters to adjust, was optimized using grid search, allowing for an exhaustive but efficient exploration of the parameter space. Conversely, Neural Network tuning was performed using Bayesian optimization, as grid search would have been computationally prohibitive due to the larger number of hyper parameters and the longer training times associated with NNs. Bayesian optimization provided a probabilistic, more efficient approach, outperforming random search by prioritizing promising hyperparameter combinations while requiring fewer iterations than grid search. The final optimized hyper parameters for each model and stage are summarized in the table above.

4 Results

4.1 Model Performance Comparison

The models were evaluated using accuracy, F1-score, recall, precision, and AUC, with results shown in figure 2. Performance improved across stages, with stage 3 showing the most significant gains. Both models performed similarly, but XGBoost had better recall in stages 1 and 2, making it more effective at identifying at-risk students early on.

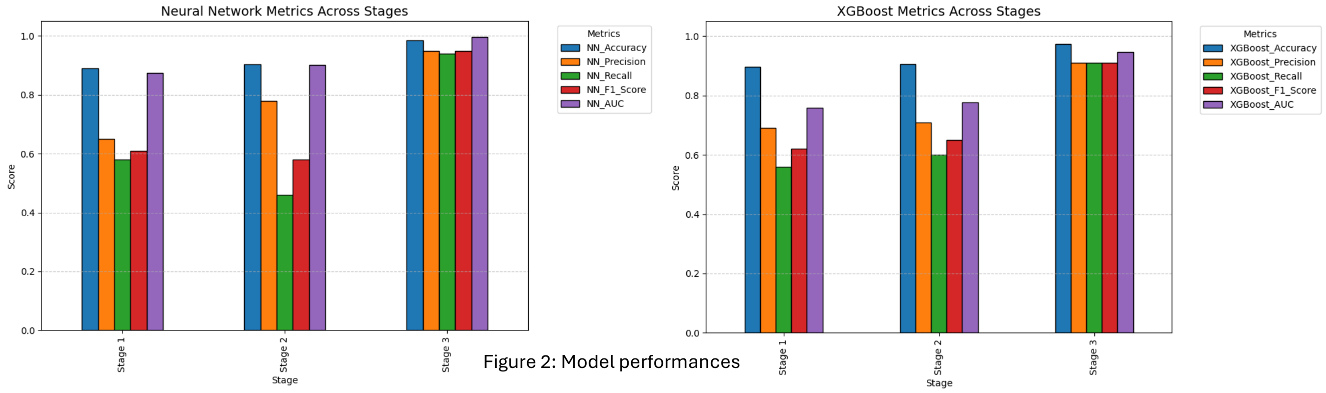

Figure 2: Model performances

The dataset was imbalanced (figure 3), with more students passing than failing. In such cases, accuracy can be misleading, which is why AUC was used for optimization. AUC prioritizes generalization and robust predictions, explaining its strong performance across all stages.

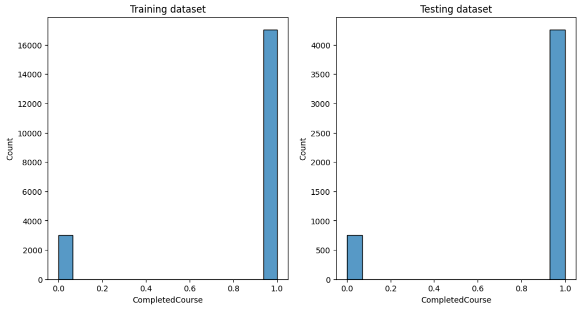

Figure 3: Class imbalance

4.2 Key Findings

In **stage 1**, performance was the lowest due to the limited data, which mainly included demographic and course information. Feature importance (figure 4) showed that nationality and course name were the strongest predictors, but since these are broad indicators, misclassifications were common.

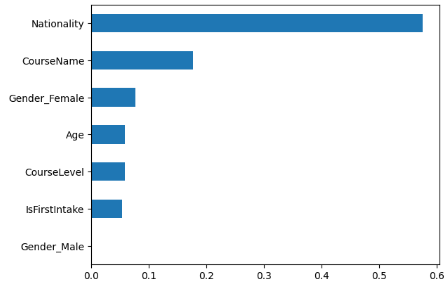

Figure 3: Feature importance (stage 1)

In **stage 2**, adding attendance data slightly improved performance. While nationality and course name remained dominant, absence counts emerged as important features, demonstrating that personal behavioral data enhances predictive accuracy.

Stage 3 saw a substantial performance jump with the inclusion of academic performance metrics. Initially, the model relied heavily on the number of courses passed, which, while predictive, risked overlooking other important features. To address this, the passed and assessed courses were combined into a normalized academic success indicator. This adjustment balanced feature importance and maintained metrics above 90 percent.

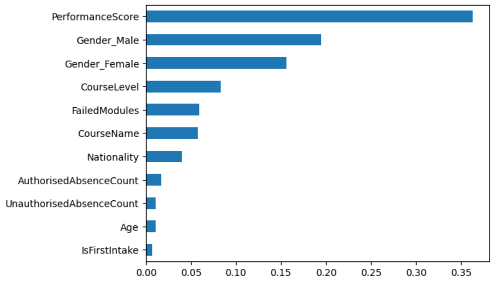

Figure 3: Feature importance (stage 3)

5 Discussion

Model performance improved progressively across stages as the availability and relevance of student data increased. In the early stage, predictions relied on indirect indicators like demographics and course information, which offered limited insight into individual risk. The inclusion of engagement data in stage 2 provided behavioral context, enabling more accurate predictions. Stage 3 saw the most substantial improvement with the addition of academic performance metrics, offering direct and highly predictive indicators of dropout risk.

Neural network performance significantly improved after hyperparameter tuning. Optimized embedding dimensions and dropout rates helped prevent overfitting, while smaller layers in stage 3 effectively captured essential patterns due to stronger features.

A key consideration was the over-reliance on the PassedModules feature in stage 3. This was addressed by creating a normalized academic success indicator, which balanced the influence of dominant features while preserving high model performance. Attempts to remove top features drastically reduced accuracy, emphasizing their critical predictive value.

6 Conclusions and Recommendations

Stage 3 models outperformed earlier stages, highlighting the strong predictive value of academic performance metrics. Both XGBoost and neural networks performed well before and after tuning, NN achieving a slightly higher AUC. Study Group should prioritize monitoring attendance and academic performance for early intervention. Future research should explore student feedback to further enhance prediction accuracy and support at-risk students more effectively.

Bibliography

Malin, M. (2025) *Why you should always use feature embeddings with structured datasets*, *Towards Data Science*. Available at: https://towardsdatascience.com/why-you-should-always-use-feature-embeddings-with-structured-datasets-7f280b40e716/ (Accessed: 16 February 2025).