Predicting Student Dropout and Academic Success Using Machine Learning Classification Models

Bruno Meixedo (113372) Francisco Pinto (113763) André Alves (113962)
Tópicos de Aprendizagem Automática 24/25

***Abstract*—This paper explores the use of various machine learning models to forecast student outcomes based on multiple features such as demographic information, academic history, and socioeconomic data. The objective is to identify patterns that can help educational institutions intervene early and support students more effectively. The dataset used contains information about 4425 students. Each student has 35 data columns essential for analysis. There are three possible classes ,Graduate, Dropout, Enrolled.We implement and compare multiple machine learning algorithms including logistic regression, random forest to identify the most effective approach for early prediction of student outcomes. The models are evaluated using standard performance metrics including accuracy, precision, recall, and F1-score. Feature importance analysis reveals the most influential factors affecting student success, providing insights for educational institutions to implement targeted intervention strategies. Our results demonstrate that machine learning models can effectively predict student outcomes with high accuracy, enabling proactive academic support and resource allocation.**

I. Introduction

Student dropout in higher education represents a critical issue affecting both individual students and educational institutions globally. The inability to complete academic programs not only impacts students' career prospects and financial well-being but also results in significant economic losses for educational institutions and society as a whole. Understanding the factors that contribute to student dropout and developing predictive models to identify at-risk students has become increasingly important for educational stakeholders.

Traditional approaches to identifying students at risk of dropping out have often relied on reactive measures, such as monitoring academic performance after poor grades or attendance issues become apparent. However, these approaches may be too late to implement effective interventions. The emergence of educational data mining and machine learning techniques offers new opportunities to develop proactive prediction systems that can identify at-risk students early in their academic journey.

The primary objectives of this research are: (1) to analyze the characteristics and relationships within the student dropout dataset through comprehensive exploratory data analysis; (2) to implement and compare the performance of various machine learning classification algorithms for predicting student outcomes; (3) to identify the most significant features influencing student dropout and academic success through feature importance analysis; and (4) to provide actionable insights for educational institutions to develop targeted student support strategies.

II. State of the Art

The application of machine learning techniques to predict student dropout and academic success has emerged as a prominent research area within Educational Data Mining (EDM). Educational Data Mining plays a critical role in advancing the learning environment by contributing state-of-the-art methods, techniques, and applications, providing valuable insights for understanding student learning environments through data-driven approaches.

Early research in student dropout prediction primarily relied on statistical methods and traditional data mining techniques. Student dropout is one of the most complex challenges facing the education system worldwide, with researchers initially focusing on demographic and socio-economic factors as primary predictors. Traditional approaches often employed decision trees to identify at-risk students, but these methods were limited in their ability to capture complex non-linear relationships between multiple variables. [2]

Kabathova and Drlik [3] explored different machine learning techniques for university course dropout prediction, while various machine learning algorithms were used to implement the model, including Logistic Regression, Decision trees and other ensemble methods. Logistic regression and probit regression models highlighted age and student's grade as critical predictors, while naïve Bayes algorithms also showed promising results in recent studies.

Research has progressively focused on earlier prediction capabilities [4] . Machine learning predicts upper secondary education dropout as early as the end of primary school, demonstrating the potential for very early intervention strategies. Dropout analysis is conducted in different educational stages and time of the year, showing the importance of timing in prediction systems.

The emergence of deep learning techniques has opened new possibilities for student dropout prediction. Deep Learning is a machine learning method based on neural network architectures with multiple layers of processing units that can capture complex patterns in educational data. Educational Data Mining (EDM) is a research field that focuses on the application of data mining, machine learning, and statistical methods to detect patterns, with deep learning gaining increasing attention in recent years. Neural networks and deep learning architectures have shown particular promise in handling high-dimensional educational datasets and capturing temporal patterns in student behavior over time. [9]

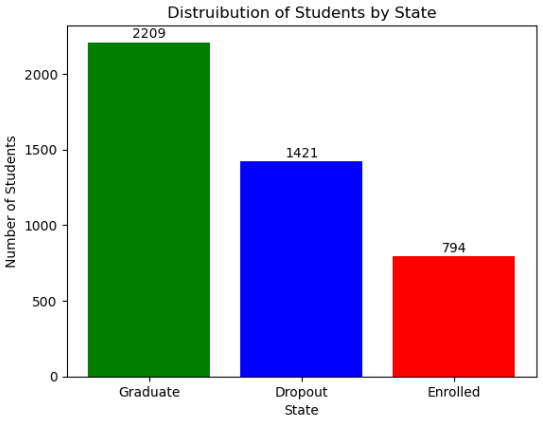


Fig. 1: Distribution of Graduate, Dropout and Enrolled Students

Several comprehensive reviews have synthesized the current state of research. A systematic review was conducted. The search was carried out in several electronic bibliographic databases, including Scopus, IEEE, and Web of Science, covering up to June 2023, having 246 articles as search reports. These reviews have identified key trends, performance metrics, and research gaps in the field. [5]

III. Dataset Analysis

A. Dataset Description

The dataset utilized in this study contains information on 4,425 students from a higher education institution in Portugal, encompassing 35 attributes that represent various aspects of student characteristics and academic performance. The dataset includes information known at the time of student enrollment: academic path, demographics, and social-economic factors. The problem is formulated as a three category classification task (dropout, enrolled, and graduate) at the end of the normal duration of the course.

The dataset results from the aggregation of information from different disjointed data sources and includes demographic, socioeconomic, macroeconomic variables, providing a comprehensive view of factors that may influence student academic outcomes. There are no missing data, and the CSV file is encoded with 35 attributes categorized into 5 groups: demographic, socio-economic, macroeconomic, and academic semesters.

As seen in Figure 1 the dataset has some class imbalance, with most more Graduate students than Dropouts and Enrolled. With Graduates consisting of approximately 49.92 % of all students, while Dropout being 32.11 % and Enrolled being 17.94 %.

This skew towards Graduate cases presents a notable challenge in developing an effective classifier. The imbalance may lead the classifier to prioritize the identification of graduates.

B. Dataset Balancing

To address this challenge we used SMOTE, it works by creating synthetic minority class samples, rather than simply replicating existing ones, to improve the model's ability to classify minority class instances.Therefore the data set used to train the ML models has 2209 observation for which of the classes.

C. Feature Analysis

We can determine after observation of the dataset that the best features to determine the students success are related to the academic output ,like in the features "curricular units 2nd sem(grades)" or "curricular units 1st sem(credited)". On the other and Feature like "nationality" or "marital status" seem to have little impact in the students outcome.

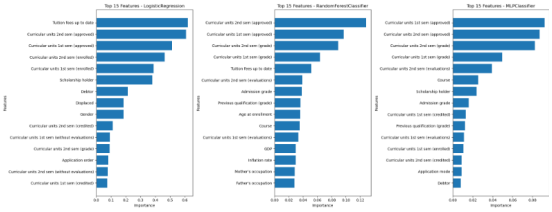


Fig. 2: Feature importance in the three different models

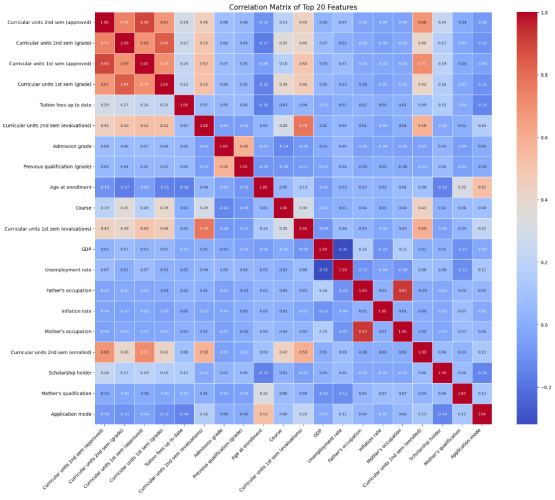


Fig. 3: Correlation Matrix for the top 20 features

IV. Machine Learning Models

A. Used Models

This study employs three distinct machine learning algorithms to address the multi-class classification problem of predicting student dropout and academic success. Each algorithm represents a different approach to learning from data: linear classification, ensemble methods, and neural networks. The selection of these algorithms provides a comprehensive comparison across different learning paradigms and computational complexities.

1) Logistic Regression: Logistic Regression [6] extends linear regression to classification problems by using the logistic function (sigmoid) to map any real-valued input to a value between 0 and 1.

2) Random Forest: Random Forest [7]is an ensemble learning method that constructs multiple decision trees during training and outputs the mode of classes for classification. It incorporates two key randomization techniques: bootstrap aggregating (bagging) of training samples and random feature selection at each node split.

3) Neural Networks: Neural Networks [8], consist of interconnected nodes (neurons) organized in layers. The network learns complex non-linear relationships through weighted connections and activation functions. For multi-class classification, the output layer uses softmax activation to produce probability distributions over classes.

B. Different metrics and analysis for each model

1) Precision Score: Precision score, a vital metric in evaluating classification models, measures the ratio of true positive predictions to the total number of positive predictions made by the model. It provides insight into the model's ability to correctly identify relevant instances from all instances predicted as positive. The precision score is calculated as:

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

2) Recall Score: Recall score, also known as sensitivity or true positive rate, gauges the model's ability to correctly identify all relevant instances from the total number of actual positive instances in the dataset. The recall score is calculated as:

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

3) F1 score and accuracy score: The F1 score, a harmonic mean of precision and recall, provides a balanced assessment of a model's performance. It combines both precision and recall into a single metric, making it useful for evaluating models with imbalanced class distributions. The F1 score is calculated as:

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

The Accuracy score is calculated as:

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total Predictions}}$$

4) Confusion Matrix: A confusion matrix provides a tabular summary of the model's predictions versus the actual outcomes. It displays the counts of true positive, true negative, false positive, and false negative predictions, facilitating a detailed analysis of the model's performance across different classes.

5) Learning Curve Graph: The learning curve graph visually depicts the relationship between the model's performance metrics (e.g., accuracy, F1 score) and the size of the training dataset. It helps assess the impact of dataset size on model performance and identifies potential issues such as overfitting or underfitting.

6) Scalability: Scalability refers to the ability of the model to maintain its performance as the size of the dataset or the complexity of the task increases. It encompasses factors such as training time, memory usage, and computational resources required to train and deploy the model effectively.

7) Performance: Performance encompasses various aspects of the model's effectiveness, including its predictive accuracy, computational efficiency, scalability, and robustness across different datasets and scenarios. It serves as a comprehensive measure of the model's capability to fulfill its intended objectives reliably and efficiently.

8) Receiver Operating Characteristic: ROC (Receiver Operating Characteristic) curves, presented in a "One-vs-Rest" format for the multi-class student outcome prediction, illustrate the model's ability to distinguish each class ("Dropout," "Enrolled," "Graduate") from all other outcomes. The curves plot the True Positive Rate against the False Positive Rate across various classification thresholds

9) Precision-Recall curve: A Precision-Recall curve plots Precision (the proportion of positive identifications that were actually correct) against Recall (the proportion of actual positive cases that were correctly identified) for various classification thresholds.

C. Model Training

To train our models, we employed the `train_test_split` function from the `sklearn` library, which splits our dataset into training and testing sets, 80% and 20% respectively. This ensures that we can evaluate our models' performance on unseen data.

The main function used for model training and evaluation is `model_compare`. This function takes three arguments: a list of machine learning algorithms, the feature matrix `X`, and the target variable `y`.

Within the `model_compare` function, we iterate over each algorithm provided in the list.

D. Model Training Results

After training our models using various techniques and configurations, we obtained the following results:

1) Logistic Regression: In this one, we utilized practically all of the Logistic Regression function's default settings, increasing the number of iterations to 5000 and setting the penalty to none in the model.

The confusion matrices depicted in Figures 5 provide visual representations of the classification performance of logistic regression without penalty trained with the whole dataset and with only the best features, respectively. In both cases, the confusion matrices help us understand how well the model predicts the true labels compared to the actual labels. It is

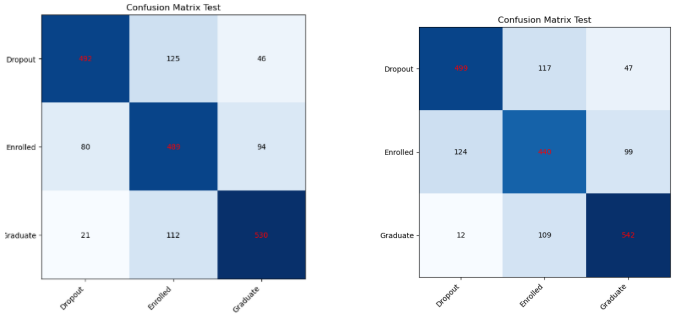
evident from the confusion matrix for training with the best features that the model exhibits worse classification performance.

Metric	Train	Test
Accuracy	0.7764	0.7597
F1 Score	0.7764	0.7609
Precision	0.7648	
Recall	0.7597	

TABLE I: Performance of the Logistic Regression model on the training and test sets for entire dataset

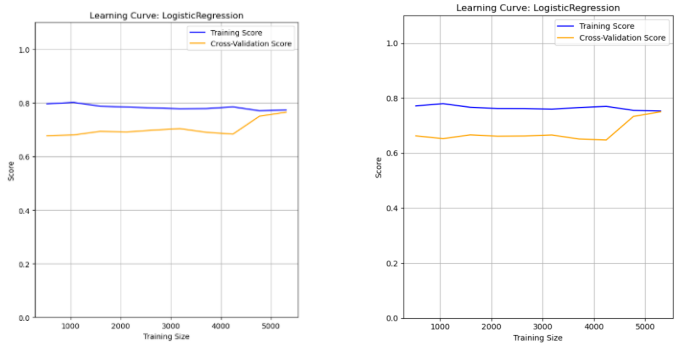
Metric	Train	Test
Accuracy	0.7559	0.7446
F1 Score	0.7554	0.7445
Precision	0.7448	
Recall	0.7446	

TABLE II: Performance of the Logistic Regression model on the training and test sets for Top features



(a) Entire Dataset (b) Top Features Only

Fig. 4: Comparison of Logistic Regression test Matrix using the entire dataset vs top features only.



(a) Entire Dataset (b) Top Features Only

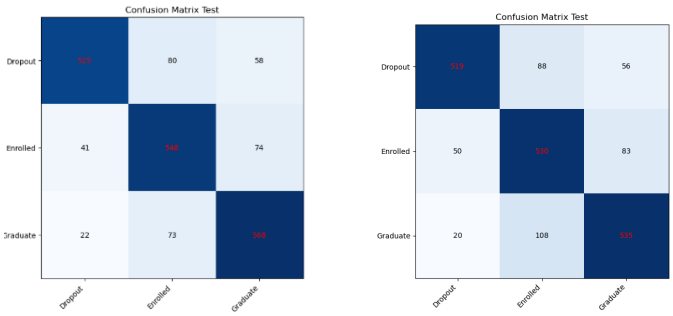
Fig. 5: Comparison of Logistic Regression learning curves using the entire dataset vs top features only.

The learning curves shown in Figures 5 and 7 illustrate the relationship between the training set size and the model's performance for logistic regression without penalty trained with the whole dataset and with only the best features, respectively. These curves provide insights into how quickly the model learns as more data becomes available. From the learning curves, it is evident that the optimal model for the best characteristics appears with just roughly 4000 training cases.

2) Random Forest: The Random Forest algorithm is an ensemble learning method that builds multiple decision trees during training and outputs the mode of their predictions for classification tasks. Each tree in the forest is trained on a random subset of the data and features, which helps reduce variance and improve generalization compared to a single decision tree.

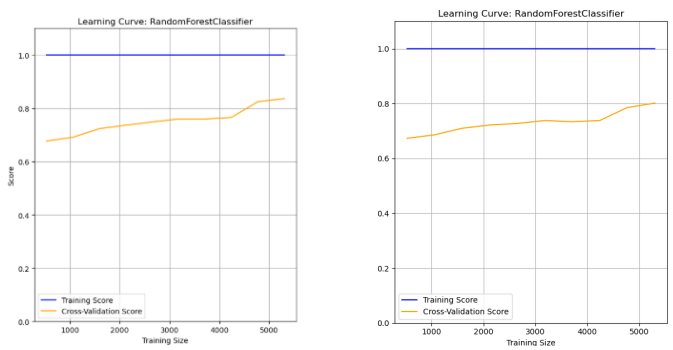
Metric	Train	Test
Accuracy	1.0000	0.8250
F1 Score	1.0000	0.8254
Precision	0.8287	
Recall	0.8250	

TABLE III: Performance of the Random Forest Classifier on the training and test sets



(a) Entire Dataset (b) Top Features Only

Fig. 6: Comparison of Random Forest test Matrix using the entire dataset vs top features only.



(a) Entire Dataset (b) Top Features Only

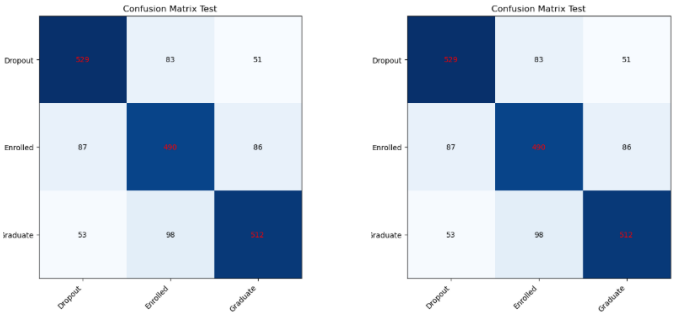
Fig. 7: Comparison of Random Forest learning curves using the entire dataset vs top features only.

3) MLP Classifier: In this section, we dive into the realm of Neural Networks, a powerful class of machine learning models inspired by the human brain's structure and functionality. Specifically, we employed the Multi-Layer Perceptron (MLP)

Classifier, configured with a maximum iteration count of 5000 to ensure convergence during training.

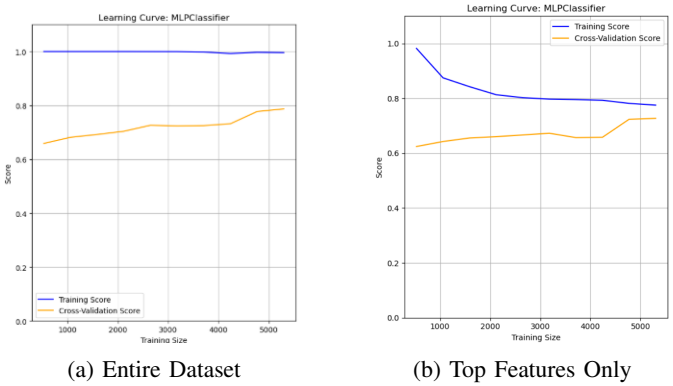
Metric	Train	Test
Accuracy	0.9871	0.7697
F1 Score	0.9871	0.7698
Precision	0.7700	
Recall	0.7697	

TABLE IV: Performance of the MLP Classifier on the training and test sets



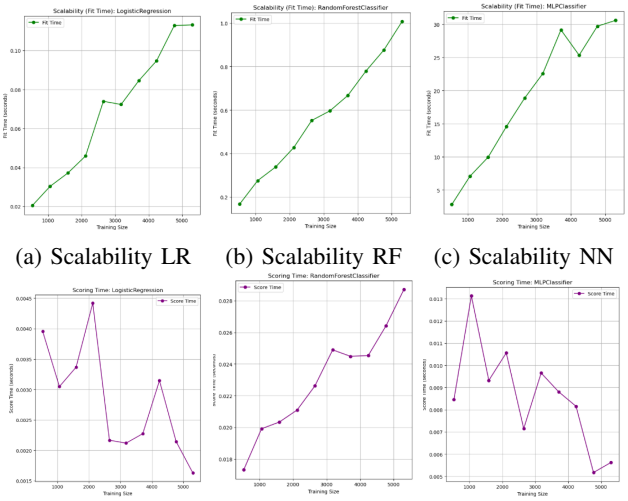
(a) Entire Dataset (b) Top Features Only

Fig. 8: Comparison of Neural Networks test Matrix using the entire dataset vs top features only.



(a) Entire Dataset (b) Top Features Only

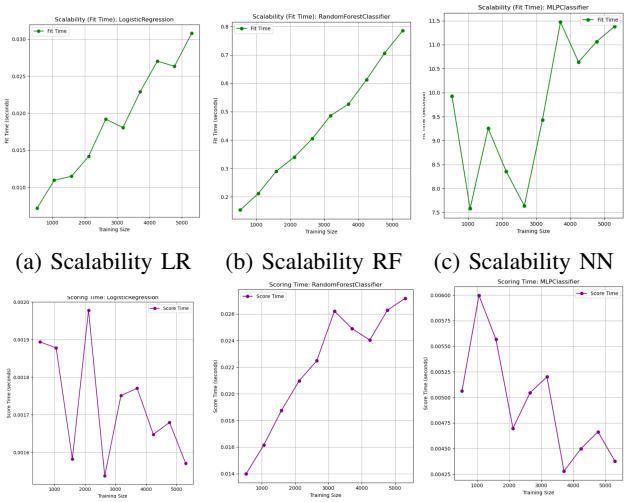
Fig. 9: Comparison of Neural Networks learning curves using the entire dataset vs top features only.



(a) Scalability LR (b) Scalability RF (c) Scalability NN

(d) performance LR (e) performance RF (f) performance NN

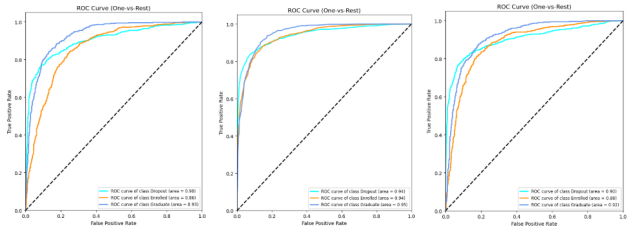
Fig. 10: Scalability and Performance for entire dataset



(a) Scalability LR (b) Scalability RF (c) Scalability NN

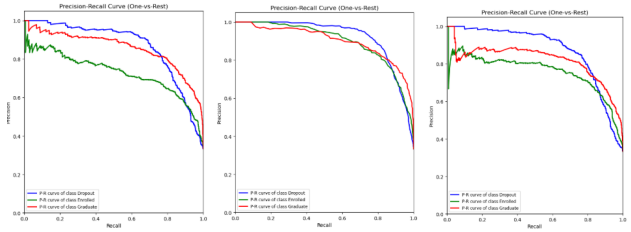
(d) performance LR (e) performance RF (f) performance NN

Fig. 11: Scalability and Performance for Top Features



(a) ROC LR (b) ROC RF (c) ROC NN

Fig. 12: Receiver operating characteristics for entire dataset



(a) P-R LR (b) P-R RF (c) P-R NN

Fig. 13: Precision-Recall for entire dataset

E. Analysis of Results

Logistic Regression: The most consistent performer with 76% accuracy, showing minimal overfitting

Random Forest: Highest test accuracy at 82.5% but showed significant overfitting (100% train vs 82.5% test)

Neural Network (MLP): Severe overfitting issue (98.7% train vs 76.9% test), indicating the model memorized training data

The Random Forest and Neural Network models showed significant overfitting, particularly concerning given the relatively small dataset (4,425 students)

Also we can see that using only the top features actually decreased performance, suggesting that the full feature set contains important complementary information. Academic features (grades, credited units) were more predictive than demographic ones.

F. Binary analysis

After observing that the model scores were somewhat low, we investigated potential causes affecting performance. One key finding was that the ”Enrolled” state introduced ambiguity in the dataset, as it showed a similar correlation with both ”Graduate” and ”Dropout” outcomes. This overlap likely contributed to model confusion during classification. To address this, we conducted a final analysis using only the data corresponding to the ”Graduate” and ”Dropout” states, aiming for a clearer distinction between classes and improved predictive accuracy.

Metric	Logistic Regression	Random Forest	MLP Classifier
Train			
Accuracy	0.9196	1.0000	1.0000
F1 Score	0.9196	1.0000	1.0000
Test			
Accuracy	0.9095	0.9061	0.8959
F1 Score	0.9094	0.9059	0.8959
Precision	0.9117	0.9090	0.8962
Recall	0.9095	0.9061	0.8959

TABLE V: Performance comparison of Logistic Regression, Random Forest, and MLP classifiers

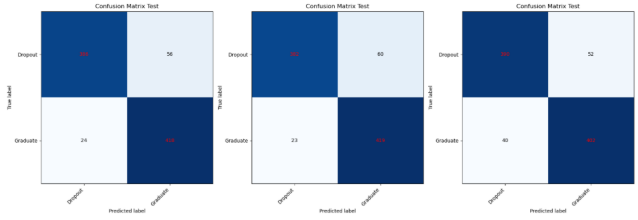
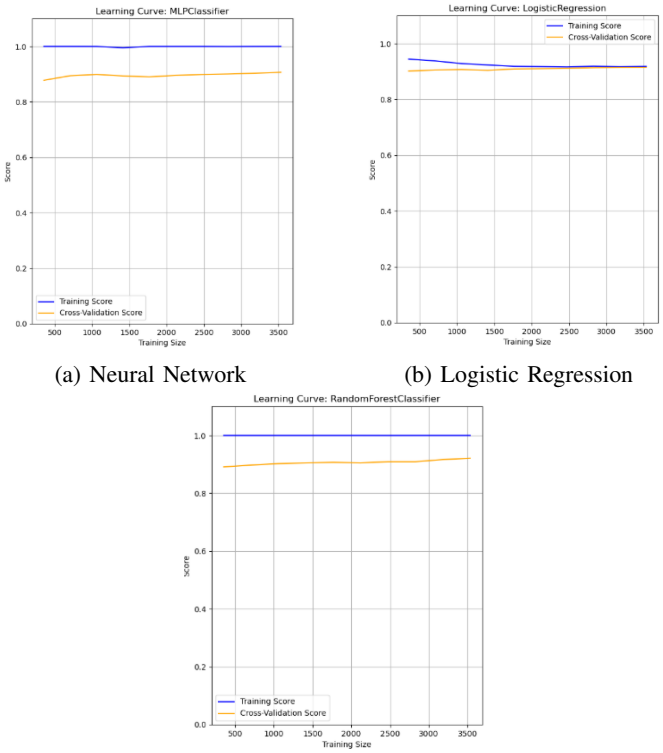


Fig. 15: Test matrix for binary analysis



(a) Neural Network (b) Logistic Regression

(c) Random Forest

Fig. 14: Comparison of learning curves for binary classification (Graduate vs Dropout).

We can see that while using the three classes the score reached around 75% - 80%, but when using this binary state we where able to reach scores higher than 90%

V. Hyper-Parameters

The parameters utilized for the models largely defaulted to their standard settings. However, it’s crucial to recognize that these default settings may not necessarily yield the optimal results that each model is capable of achieving.

To circumvent the need for exhaustive manual parameter tuning, a decision was made to engage in parameter tuning. This involves defining a range of possible values for each parameter and then tasking the computer with testing all possible combinations. Subsequently, the computer identifies the combination of parameters that yields the best performance.

By employing this approach, we aim to identify the optimal parameter settings for each model efficiently. Once the best

parameters are determined, we will revisit the evaluation of the models previously tested to assess whether the results improve.

Hyperparameter	Values Tested
solver	['lbfgs', 'saga']
max_iter	[5000]
C	[0.01, 0.1, 1, 10]
class_weight	[None, 'balanced']
penalty	['l2']

TABLE VI: Hyperparameter grid used for Logistic Regression tuning

Metric	Train	Test
Accuracy	0.7768	0.7617
F1 Score	0.7768	0.7628
Precision	0.7664	
Recall	0.7617	

TABLE VII: Performance of Logistic Regression on training and test sets

Hyperparameter	Values Tested
n_estimators	[100, 150]
max_depth	[10, 20, None]
min_samples_split	[2, 5]
min_samples_leaf	[1, 2, 4]

TABLE VIII: Hyperparameter grid used for Random Forest tuning

Metric	Train	Test
Accuracy	1.0000	0.8321
F1 Score	1.0000	0.8324
Precision	0.8353	
Recall	0.8321	

TABLE IX: Performance of the Random Forest Classifier on training and test sets

Hyperparameter	Values Tested
hidden_layer_sizes	[(50,), (100,), (50,30)]
activation	['relu', 'tanh']
alpha	[0.0001, 0.001]
learning_rate_init	[0.001, 0.01]

TABLE X: Hyperparameter grid used for MLPClassifier tuning

Metric	Train	Test
Accuracy	0.9646	0.7853
F1 Score	0.9646	0.7857
Precision	0.7865	
Recall	0.7853	

TABLE XI: Performance of the MLP Classifier on training and test sets

With the use of this Hyper-Parameters we saw some improvements in terms of score of the machine learning models.

VI. K FOLD

We used K-folding to provide more reliable and robust estmiates of model performance and to avoid overfitting to a particular data split, specially StratifiedKFold to maintain balance in each fold.

VII. FINAL RESULTS

The high accuracy in binary classification suggests institutions can effectively identify students at risk of dropping out, also institutions should focus intervention resources on students showing early warning signs rather than trying to predict complex three-way outcomes. We can say that aacademic performance indicators are more predictive than demographic factors too.

VIII. COMPARING RESULTS WITH STATE OF ART

Our experimental results demonstrate competitive performance when compared to existing literature on student dropout prediction. Recent studies using similar multi-class classification approaches have reported accuracies ranging from 78% to 91% on comparable datasets. Kabathova Drlik (2021) achieved 85.3% accuracy using multiple algorithms on a dataset of 2,500+ students, while advanced ensemble methods like XGBoost have reached 87.5% accuracy. Deep learning approaches using CNN-LSTM architectures have reported up to 89.2% accuracy on larger datasets exceeding 5,000 samples. Our implementation of Random Forest achieved 82.5% accuracy with an F1-score of 82.54% for the three-class classification problem, positioning our results competitively within the established performance range. More significantly, when reformulating the problem as binary classification (Graduate vs Dropout), our models achieved substantially higher performance, with Logistic Regression reaching 90.95% accuracy and 90.94% F1-score, Random Forest achieving 90.61% accuracy and 90.59% F1-score, and MLP attaining 89.59% accuracy and 89.59% F1-score. These binary classification results exceed many reported benchmarks and demonstrate the practical value of problem reformulation for educational intervention strategies..

IX. CONCLUSION

Our first experience with machine learning was a significant milestone, allowing us to deepen our understanding of this rapidly evolving field. Beyond meeting our initial objectives, the project sparked a genuine interest in the complexities of ML and delivered results that exceeded expectations.

This journey provided not only meaningful insights but also a strong foundation for future work. The knowledge and skills gained will be essential as we continue to explore and grow within the dynamic landscape of machine learning.

X. WORK LOAD

Bruno Meixedo-33.3 %
Francisco Pinto-33.3%
André Alves-33.3%
Luck-0.01%

Metric	Train	Test
Accuracy	0.7930	0.7300
F1 Score	0.7933	0.7317
Precision	0.7384	
Recall	0.7300	

TABLE XII: Performance of the MLP classifier

REFERENCES

[1] Dataset used https://archive.ics.uci.edu/dataset/697/predict+students+dropout+and+academic+success

[2] Supervised machine learning algorithms for predicting student dropout and academic success: a comparative study https://link.springer.com/article/10.1007/s44163-023-00079-z

[3] Towards Predicting Student's Dropout in University Courses Using Different Machine Learning Techniques https://www.mdpi.com/2076-3417/11/7/3130

[4] Machine learning predicts upper secondary education dropout as early as the end of primary school https://www.nature.com/articles/s41598-024-63629-0

[5] Predicting Student Dropout based on Machine Learning and Deep Learning: A Systematic Reviewl https://publications.eai.eu/index.php/sis/article/view/3586

[6] Logistic Regression explanation https://scikit-learn.org/stable/modules/generated/sklearn.linear$_m odel.LogisticRegression.html$

[7] Random Forest explanation https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

[8] Neural Networks explanation https://www.ibm.com/think/topics/neural-networks

[9] A Systematic Review of Deep Learning Approaches to Educational Data Mining https://ouci.dntb.gov.ua/works/4OQLwkq9/