24038714 | UFCFMJ-15-M | 2,181 words, excl. references | https://github.com/Koshaliya/ML-Coursework

Graduate dropout prediction and interpretation

***Abstract*— This report explores the predictive modeling approach to forecast student dropouts using the UCI Machine Learning Repository's real-world educational data. To identify at-risk students, six supervised learning algorithms - Random Forest, XGBoost, SVM, Logistic Regression, Decision Tree, and Neural Networks were applied after preprocessing. Models were evaluated using accuracy, F1-score, recall, ROC-AUC, training time, and interpretability. XGBoost achieved the best overall performance, while Random Forest and Logistic Regression offered strong interpretability. Feature importance and model coefficients highlighted key factors influencing dropout, supporting early intervention strategies.**

Keywords—dropout prediction; machine learning; xgboost; supervised learning; education analytics;

I. Introduction

Graduate dropout is one of the most critical problems in higher education [1]. While a high dropout rate causes financial loss for institutions, it also affects the students' prospects and leads to inefficient national education systems. Traditional methods like manual monitoring or threshold-based warnings often lack scalability and miss complex patterns, leading to delayed intervention. In contrast, ML enables continuous, scalable risk detection and allows earlier action before disengagement becomes irreversible [2].

The primary goal of this study is to incorporate machine learning to automate and increase the precision of dropout prediction, allowing institutions to quickly initiate focused support systems. Through analytical comparisons using suitable performance criteria, it assesses a collection of appropriate supervised learning algorithms. The knowledge gained from this approach is intended to show how predictive analytics may be used in the educational field to help with strategic decision-making and student achievement.

II. Dataset overview

The dataset used in this study is the "Predict Students Dropout and Academic Success" dataset, which is publicly available through the UCI machine learning repository [3]. It combines demographic, academic, financial, and institutional data linked to student performance from Portuguese higher education students.

A. *Dataset characteristics*

- Size: Around 4424 student records
- Features: 37 attributes, comprising 36 predictors and 1 target variable
- Target variable: The "Enrolled" category was eliminated in to predict whether a student will "Graduate" (1) or "Dropout" (0) for binary categorization purposes.

B. *Data preprocessing*

- The dataset is relatively clean; no missing values were found.
- Target variable was imbalanced since there were more graduates than dropouts. This was addressed using SMOTE [4]; by adding more dropout samples to the training set, recall increased from 0.80 to 0.82 without compromising evaluation fairness. However, generalization may be limited because synthetic cases might not accurately represent real behavior. Future research may investigate cost-sensitive learning as a more practical substitute.

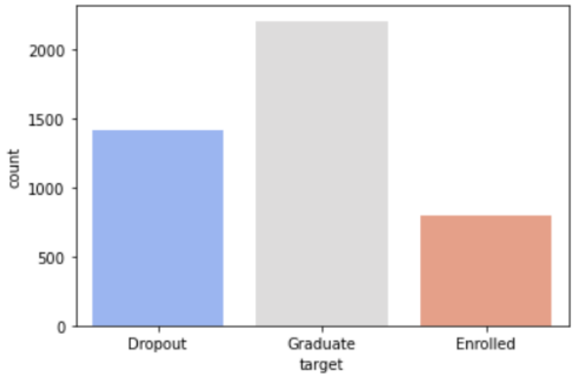

Figure 1: Distribution of target variable classes (Graduate, Dropout, Enrolled) before pre-processing

- Correlation analysis revealed several highly correlated (>0.85) features (e.g., 1st and 2nd semester curricular units and father and mother occupations). The redundant features were removed to avoid multicollinearity.
- Outliers in numerical variables were removed using the Interquartile Range (IQR) method to reduce their impact on model performance.

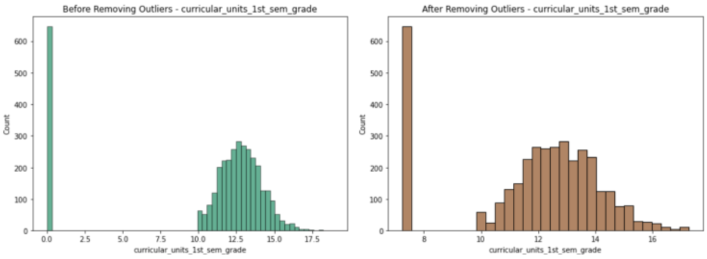

Figure 2: Outlier removal in curricular unit 1st sem grade shown as an example

The dataset is ideal for tasks involving classification. It offers a solid basis for comparing and assessing various machine learning models to forecast student success.

III. Bivariate vs Multivariate Analysis

In the context of predictive modeling, it is crucial to comprehend how input variables relate to the target. Two forms of analysis are commonly used:

- Bivariate analysis explores the relationship between two variables at once, usually between the target and an independent variable (e.g., target vs gender).
- Whereas multivariate analysis examines how several factors work together to affect the result, which helps identify more complex and logical patterns.

A. Bivariate analysis

To initiate the exploration of patterns within the dataset, bivariate analysis was performed to assess the relationship between different variables and the target variable (Graduate or Dropout). For better interpretability, the features were divided into categorical and numerical groups and analyzed accordingly.

- Categorical variables showed significant variations among target classes.
    - Students with unpaid fees or without scholarships had notably higher dropout rates.

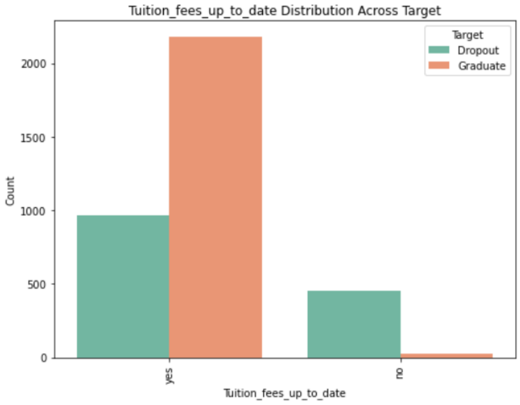

Figure 3: Tuition fees up to date feature distribution across target

- Numerical features were visualized using boxplots to compare distributions between the two classes. Key findings include:
    - Students who graduated had far higher admission grades and curricular unit grades in both semesters, indicating that success is mostly determined by academic performance.

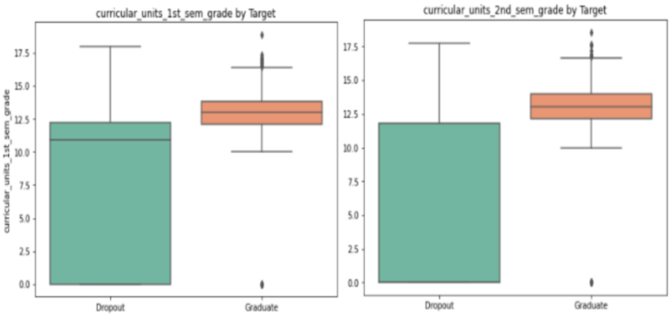

Figure 4: Boxplot of curricular unit 1st sem and 2nd sem grade vs target

-
    - Students who dropped out had slightly higher age at enrollment, which could reflect the influence of external responsibilities.

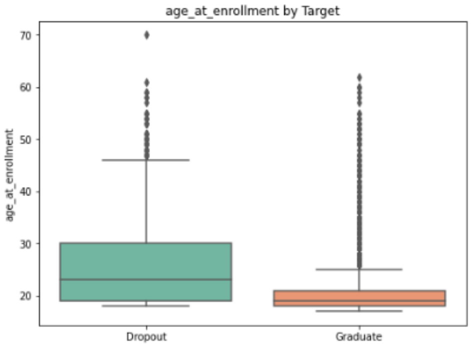

Figure 5: Age at enrollment vs target

1) Correlation Analysis

To find highly correlated pairs, Pearson correlation was computed across numerical features. Features with correlation above 0.85 were considered redundant and removed to reduce multicollinearity and improve model performance. To increase the model's generalization and efficiency, the following features were eliminated:

- mothers_occupation
- curricular_units_2nd_sem_enrolled
- curricular_units_2nd_sem_approved

While bivariate analysis is helpful in identifying initial trends, it did not adequately account for the interaction between variables. Student dropout is multifactorial; analyzing one feature at a time limits the understanding of its complexity.

B. Multivariate analysis

Multivariate analysis was used to uncover deeper patterns and dependencies between variables, which are crucial for building reliable predictive models. This analysis is particularly valuable when the target is influenced by combinations of various factors [6].

1) Principal Component Analysis (PCA)

To investigate dimensional structure, PCA was applied to standardized numerical characteristics.

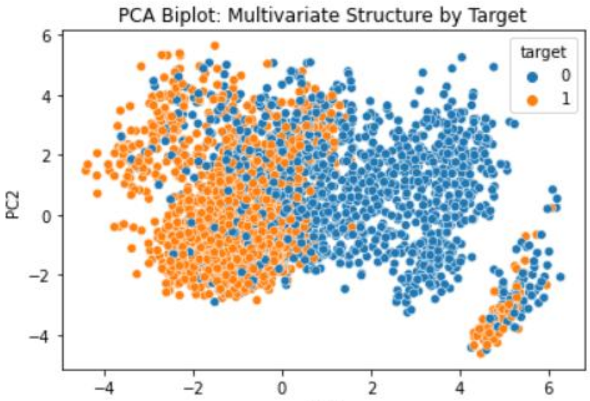

Figure 6: PCA of target variable

A biplot of the first two principal components shows a partial separation between the graduate and dropout classes indicating that multiple features contribute together to predict student outcomes. Though not used in training, PCA provided support for the choice of multivariate techniques.

While bivariate analysis offered useful initial insights multivariate analysis is more suited for this problem. Dropout prediction is rarely due to one factor; multivariate analysis better captures the combined effects of several variables. Therefore, for precise and useful predictions, machine learning models that can capture these interactions are required.

IV. Algorithm Selection and Categories

A. Algorithms selected

Dropout prediction is a supervised binary classification problem aiming to predict graduation based on input features. The optimal model should manage non-linear feature interactions, imbalanced classes [5], and mixed data types. Given the nature of the problem, the following supervised learning algorithms were chosen [7].

1) Decision Tree (Tree-based Learning) was included in evaluation to measure as a simple baseline model. It provides clear interpretability and insight into feature splits though it's prone to overfitting without pruning.

2) Random Forest (Ensemble Learning (Bagging)) was selected due to its capacity to identify feature importance, ability to manage diverse data and resistance to noise and outliers. The bagging technique lowers variance, improving generalization and resistance to overfitting.

3) XGBoost (Ensemble Learning (Boosting)) builds sequential tree with error correction to capture non-linear interactions making it ideal for handling class imbalance, non-linear interactions, and overfitting. On structured datasets, it regularly produces advanced results, which makes it perfect for dropout prediction.

4) SVM with RBF Kernel (Kernel-based Learning) by projecting data into higher dimensions it captures complex, non-linear decision boundaries and has ability to distinguish between graduate and dropout students. It was preferred over linear and polynomial kernels due to non-linearity of data and overfitting risk.

5) Logistic Regression was used as a baseline due to its simplicity and interpretability. Its coefficients provide clear insights into feature impact, making it useful for comparison, though it has limited flexibility with non-linear patterns.

6) Neural Networks capture complex, non-linear feature relationships. Though computationally intensive, they generalize well using techniques like dropout and batch normalization, effectively modeling patterns that influence dropout outcomes.

B. Algorithms not selected

1) Naive Bayes (Supervised Learning - Probabilistic classifier) was not chosen since this dataset does not support the assumption that all features are independent of one another as there are highly correlated features. Additionally, it is less appropriate for forecasting student dropout in situations where mixed and interdependent features are common because it has trouble handling continuous variables unless they follow certain distributions.

2) K-Nearest Neighbors (Supervised Learning - Instance-based learning) was not selected due to its sensitivity to high-dimensional data and computational inefficiency. Moreover, it generally performs weaker than ensemble models in imbalanced situations.

3) BaggingClassifier / AdaBoost (Supervised Learning - Ensemble methods) principles are already incorporated in Random Forest and XGBoost, which were found to perform better than them in terms of stability and accuracy.

4) Unsupervised or semi-supervised approaches were not considered due to label availability in the dataset.

Ensemble methods like Random Forest and XGBoost [9] are identified as the best algorithms due to their robustness and ability to manage diverse information. The model choices are supported by these findings as well as empirical results gathered during evaluation.

V. Evaluation

Six supervised learning algorithms were implemented and evaluated using a wide range of performance, efficiency, and interpretability parameters. The evaluation process covered the following metrics in Figure 7.

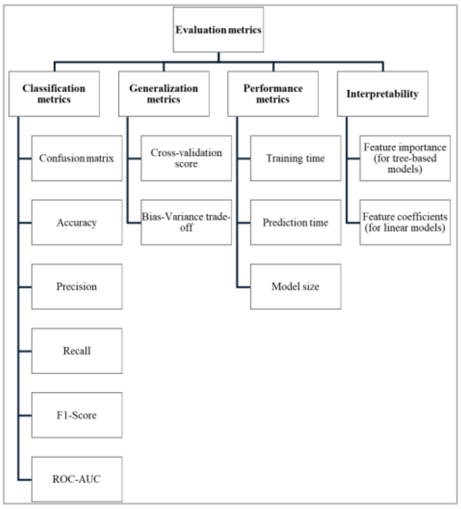

Figure 7: Evaluation metrics

In this case, high recall ensures that at-risk students are identified early, even at the expense of precision. False positives may lead to additional student support, but false negatives could mean a dropout is entirely missed, which is far costlier [11].

A. *Classification, generalization and performance analysis*

Metric	Decision Tree	Random Forest	XGBoost	SVM	Logistic Reg.	Neural network
F1-Score	0.881	0.901	0.905	0.887	0.895	0.888
Recall	0.950	0.923	0.930	0.934	0.937	0.914
Precision	0.822	0.879	0.882	0.845	0.857	0.863
Accuracy	0.865	0.893	0.898	0.875	0.885	0.879
ROC-AUC	0.861	0.891	0.896	0.872	0.882	N/A
CV Score	0.875	0.902	0.903	0.896	0.889	N/A
Train accuracy	0.892	1.000	0.963	0.926	0.894	0.955
Bias-variance gap	0.026	0.107	0.065	0.051	0.01	0.086
Overfitting	No	Yes	No	No	No	No
Training time (s)	0.066	2.199	0.513	3.948	0.029	19.71
Prediction time (s)	0.00	0.047	0.00	0.246	0.00	0.174
Model size (KB)	0.05	0.05	0.05	0.05	0.05	0.05

Table 1: Performance metrics of 6 supervised algorithms

XGBoost emerged as the best-performing model with strong recall (0.930), and highest F1-score (0.905) indicating excellent balance between precision and the ability to correctly identify dropouts. It obtained high accuracy (0.898) and a high cross validation score (0.903) showing good generalization to unseen data. It trained rapidly (0.513s) and prevented overfitting. These results along with the ability to handle nonlinear interactions and class imbalance make this most suitable model and dependable for early-warning systems.

Random Forest showed strong performance (F1-score: 0.901, recall: 0.923) and its feature importance adds interpretability. But significant overfitting limits generalization and its reliability in real-world deployment. In educational contexts, such unreliability could lead to misclassification of students and inappropriate interventions.

Logistic regression balanced performance (F1: 0.895, recall: 0.937) with excellent generalization and speed, making it ideal for transparent, low-resource environments with interpretability through coefficients.

SVM performed similarly (recall: 0.887) but required more time and lacked interpretability, limiting practical application.

Neural networks yielded competitive results (F1-score: 0.888, recall: 0.914), but longer training time and high variance limit suitability. With around 3,000 samples (after filtering “Enrolled” class) generalization may suffer, as NN typically need 5,000–10,000+ samples.

While decision trees achieved the highest recall (0.950), they had the lowest overall accuracy (0.865). Their simplicity and speed are strength, but they are best suited as a baseline model, not for production.

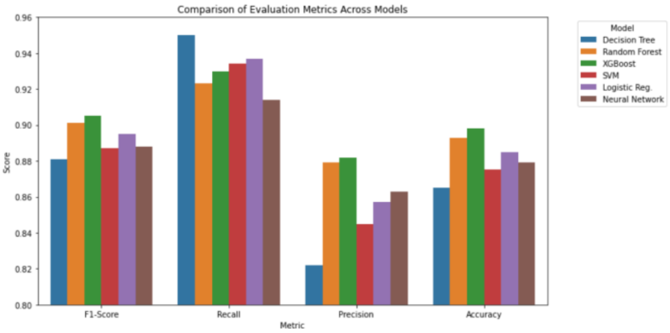

Figure 8: Comparison of model evaluation metrics across 6 supervised algorithms

Trade-Off analysis:

Each model presents trade-offs between performance, interpretability, and computational cost. Model size was identical (0.05 KB) across all models, so it had no impact on model selection. While models such as XGBoost require hyperparameter adjustment, they offer great accuracy and generalization. Logistic regression is interpretable but struggles with non-linear patterns. While neural networks are powerful but computationally demanding and less suitable for small datasets. Random forests and Decision trees are interpretable but prone to overfitting if they are not properly trimmed or regularized [10]. Hence, XGBoost was preferred for performance, while Logistic Regression suits transparent, low-cost deployment. Its interpretability enables trust, which is critical in education, where clarity often outweighs small accuracy gains.

B. *Feature Importance and Coefficients*

Understanding feature contributions improves interpretability and allows target intervention effectively [8].

1) *Feature Importance:*

- The most significant predictor was consistently curricular_units_1st_sem_approved, which was derived from tree-based models (Random Forest, XGBoost, and Decision Tree) highlighting early academic performance as critical.
- Other influential features including tuition_fees_up_to_date, curricular_units_2nd_sem_grade, age, course and curricular_units_2nd_sem_evaluations reflect financial stability and continued performance.

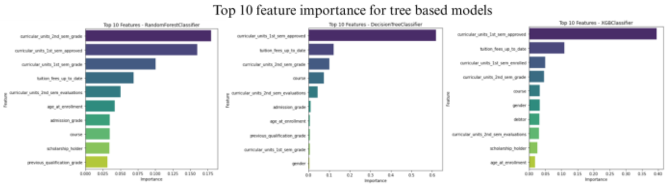

Figure 9: Feature importance comparison of Random Forest, Decision tree, XGBoost

2) *Feature Coefficients:*

- Logistic Regression further confirms the above patterns.
- Positive coefficients (e.g., curricular_units_1st_sem_approved) increase the chance of graduation. while negative coefficients (e.g., curricular_units_1st_sem_enrolled or course) may signal higher dropout risk. These insights align with academic expectations provide actionable variables for early intervention strategies.

VI. Conclusion and Future Work

This study applied machine learning to predict student dropout using real world data. Among the six models tested, XGBoost gave the best results with high accuracy and recall. Random Forest performed well in recall and feature importance but showed overfitting without tuning. Logistic Regression, though slightly less accurate, offered speed, strong generalization, and clear interpretability which is ideal for transparent educational use. Key features influencing outcomes included first-semester academic performance, tuition payment status, course and second-semester grades. Institutions can prioritize early academic support, course guidance, and financial aid monitoring to proactively reduce dropout rates.

Future work

The current models used simple number-based encoding for categorical variables. Using methods like one-hot encoding or target encoding can improve accuracy by preserving the structure, reducing misinterpretation in models like Logistic Regression. Tuning hyperparameters can enhance performance and reduce overfitting and make the models more reliable, as most models were tested with default settings. Further explore unsupervised and semi-supervised models to uncover latent student groupings and leverage partially labeled data, enabling deeper pattern discovery and broader generalization beyond fully supervised settings. These improvements can enhance early intervention for at-risk students.

References

[1] L. Aulck, N. Velagapudi, J. Blumenstock, and J. West, "Predicting Student Dropout in Higher Education," arXiv preprint arXiv:1606.06364, 2016. [Online]. Available: https://arxiv.org/abs/1606.06364

[2] M. Yağcı, “Educational data mining: prediction of students’ academic performance using machine learning algorithms,” Smart Learning Environments, vol. 9, no. 1, Mar. 2022, doi: https://doi.org/10.1186/s40561-022-00192-z.

[3] V. Realinho, M. Vieira Martins, J. Machado, and L. Baptista. "Predict Students' Dropout and Academic Success," UCI Machine Learning Repository, 2021. [Online]. Available: https://doi.org/10.24432/C5MC89.

[4] T. Tang, “Class Imbalance Strategies — A Visual Guide with Code,” Medium, Apr. 24, 2023. https://medium.com/data-science/class-imbalance-strategies-a-visual-guide-with-code-8bc8fae71e1a

[5] W.-J. . Lin and J. J. Chen, “Class-imbalanced classifiers for high-dimensional data,” Briefings in Bioinformatics, vol. 14, no. 1, pp. 13–26, Mar. 2012, doi: https://doi.org/10.1093/bib/bbs006.

[6] Z. Song, S.-H. Sung, D.-M. Park, andx B.-K. Park, “All-Year Dropout Prediction Modeling and Analysis for University Students,” Applied Sciences, vol. 13, no. 2, p. 1143, Jan. 2023, doi: https://doi.org/10.3390/app13021143.

[7] S. Tavasoli, “Top 10 Machine Learning Algorithms You Need to Know in 2020,” Simplilearn.com, Nov. 09, 2016. https://www.simplilearn.com/10-algorithms-machine-learning-engineers-need-to-know-article

[8] A. Mukherjee, “Feature Selection and Feature Importance- Section I,” Medium, Sep. 29, 2023. https://medium.com/@mukherjee.abhradeep/feature-selection-and-feature-importance-section-1-a9b460eeb997 (accessed Aug. 07, 2024).

[9] P. Sonawane, “XGBoost — How does this work,” Medium, Dec. 08, 2023. https://medium.com/@prathameshsonawane/xgboost-how-does-this-work-e1cae7c5b6cb

[10] B. Wohlwend, “Decision Tree, Random Forest, and XGBoost: An Exploration into the Heart of Machine Learning,” Medium, Jul. 23, 2023. https://medium.com/@brandon93.w/decision-tree-random-forest-and-xgboost-an-exploration-into-the-heart-of-machine-learning-90dc212f4948

[11] Vaibhav Jayaswal, “Performance Metrics: Confusion matrix, Precision, Recall, and F1 Score | Towards Data Science,” Towards Data Science, Sep. 14, 2020. https://towardsdatascience.com/performance-metrics-confusion-matrix-precision-recall-and-f1-score-a8fe076a2262/