Application of Machine Learning Algorithms to Predict Academic Success

Lorena Benavides Riano – PhD Student Engineering Education
John Alexander Yepes – Master Student Data Analytics
Mississippi State University

Keywords
Academic achievement, student success, machine learning

1. Introduction

Higher education institutions are constantly searching for methods to assess the effectiveness of their educational practices to prepare students for the workforce and academia after successfully completing the educational cycle. Recent studies in education have focused on understanding elements that contribute to student retention and success across different fields [1], [2]. Understanding student success as completing academic requirements, persistence, development of professional skills, goal attainment of learning outcomes, and career success [3]. One important indicator across universities is the balance of enrollment and dropout rates, which show the failure to retain students in formal education settings and represent a major concern in post-secondary education.

One way to address this problem is to detect students who are at risk of changing their decisions about completing their degrees early. With this information, educators can develop interventions to advise students and prevent negative outcomes. One way to address this is by studying available historical data and identifying pivotal factors that can help to inform these behaviors in advance. This study focuses on understanding academic achievement and persistence by applying machine learning techniques in order to separate students who might not complete their studies from the ones who are more likely to graduate. The main goal of this research is to identify any specific factor or set of attributes associated with academic success and student retention. Therefore, this study aims to contribute to student retention strategies as well as provide educators with more tools that could benefit their professional practice and, in the end, understand student challenges in college life.

The structure of this paper includes five sections. A general introduction to the importance of the research questions is developed in this section. Related works about data science techniques in predicting student academic achievement and persistence are in Section 2. Section 3 presents an overview of the methodology followed, explaining the science of the models such as KNN and Naïve Bayes - NB (Gaussian and Bernoulli), Decision Tree (DT), Random Forest (RF), Gradient Boosting (GB), and Support Vector Machine (SVM). Then, section 4 focuses on the experiment, data engineering, and results obtained before and after transforming the dataset, as well as the best model(s). Finally, Section 5 concludes the study findings and suggestions for future work.

2. Literature Review

Predicting academic achievement and dropout outcomes has been a matter of interest throughout higher education. Consequently, different techniques and tools have been applied, ranging from traditional statistical analysis [4] to data mining [5], artificial intelligence [6], and machine learning. Depending on each approach and research objective, each study has provided useful information. For this literature review, the focus is on understanding recent contribution to the research topic that involve machine learning models for classification problems.

Hussain and Khan [7] used a dataset (30 attributes) from a secondary school examination comprising 9th to 12th classes. A supervised K-NN regression model and a decision tree classifier were applied to build a prediction model of student performance. The DT classifier yielded 94.39% average accuracy, while the K-NN classifier obtained 85.74%. However, when these approaches relied on genetic algorithms, the performance increased by approximately 3% in each case. The results showed that when students had the same educational background or historical records, the educational background or historical records results were more reliable in predicting students' grades.

Mustafa [8] compared the performances of the random forests, nearest neighbors, support vector machines, logistic regression, Naïve Bayes, and k-nearest Neighbors techniques to predict the final exam marks of 1854 students with just four attributes. The predictions of models were evaluated using a confusion matrix. The classification accuracy ranged from 0.699 to 0.746, the highest classified correctly corresponding to Random Forest and K-Nearest Neighbors algorithms. The results highlight the power of machine learning methods in predicting academic performance even with few parameters.

One of the most recent studies showed that across 70 paper journals from 2022 to 2023 focused on predicting college student graduation in a variety of large datasets. Factors considered for the analysis varied from prior academic achievement, demographics, academic records, and psychological measures. Using more frequent machine learning techniques, SVM, RF, and LR, respectively, the prediction of student graduation accuracy could be as high as 90%. Moreover, models are also useful in analyzing student progression and are capable of handling binary and multi-class classification [9].

In general, machine learning tools are widely used in education research; however, there are still limitations from which algorithm led to more accurate results based on the size of the dataset, the reliance on one source of information, and the quality and number of parameters used for the forecast of student academic success.

3. Methodology

This is a supervised dataset, which will be analyzed using different models to solve this classification problem. These models are k-neighbors, Naive Bayes, Decision Tree, Random Forest, Gradient Boosting, and Support Vector Machine (non-escalated and escalated), and their hyperparameters will be tuned to obtain the model with the best performance.

3.1. k-Nearest Neighbors (KNN): this learning algorithm is quick when processing classification problems. KNN predicts the class of a new data point by identifying the closest points in the training set, which are stored in memory. The algorithm determines the "closeness" or similarity between points by calculating distances, typically using metrics such as Euclidean or Manhattan distance [10]. The Euclidian formula is:

$$d(x,y) = \sqrt{(\sum_{i=1}^{N}(x_i - y_i)^2)}. \qquad (1)$$

3.2. Naive Bayes (NB): The Naive Bayes Classifier is an algorithm based on the Bayes theorem, and it is known to be better than other classification methods. This is because the model is simple and easy to use, it assumes independence for each event, and it can be implemented for large datasets [11]. The Naive formula is:

$$P(C|x_1, x_2, \ldots, x_n) = \frac{P(C)P(x_1, x_2, \ldots, x_n|C)}{P(x_1, x_2, \ldots, x_n\ )} \qquad (2)$$

3.3. Decision Tree (DT): this is one of the most important algorithms in supervised learning. This model uses a top-down method, which is known as recursive splitting. It can overfit easily because if there is no defined deepness, so pruning can be used. The metric used to calculate the tree is called the Gini index, and the Entropy has a range of 0 and 1 to divide each node **[12]**. The Gini or Entropy formulas are the following:

$$\text{Gini}(p) = \sum_{k=1}^{m} p_k(1-p_k) = 1 - \sum_{k=1}^{K} p_k^2 \text{ for Gini, and } I(S_{1j}, S_{2j}, \ldots, S_m) = -\sum_{i=1}^{m} P_i \log_2 P_i \text{ for Entropy.} \qquad (3)$$

3.4. Random Forest (RF): It is a technique that combines predictions from many decision trees to reduce variance. This process makes it more robust than single decision trees, and it was initially introduced by Breiman (2001). Though RF can handle various types of response variables, this paper focuses primarily on binary classification and regression **[13]**. For classification, the formula used is the following:

$$\hat{p}_i = \frac{1}{T}\sum_{t=1}^{T} I(\hat{y}_{it} = 1) \text{ , where } \hat{y}_i = \begin{cases} 1 & \text{if } \hat{p}_i > 0.5, \\ 0 & \text{otherwise.} \end{cases} \qquad (4)$$

3.5. Gradient Boosting Decision (GBD): this algorithm combines multiple classifiers and takes regression decision trees to improve classification accuracy. By using a weighted approach, each classifier in GBDT is trained to reduce the residual errors from the previous iteration, to gradually improve the model. This process is known as the forward stagewise algorithm, and it builds the model step-by-step by sequentially adding each classifier at a time, with each step relying on all prior classifiers [14]. The formula is the following:

$$f_m(x) = f_{m-1}(x) + T(x;\Theta_m) \qquad (5)$$

3.6. Support Vector Machine (SVM): It is a linear classifier that separates a dataset by creating many hyperplanes that separate the data. SVM identifies the optimal separating hyperplane based on the closest support vectors (points) to each group. SVM has a hyperparameter (C) that controls the trade-off between maximizing the margin and minimizing classification errors. Another hyperparameter called Kernel enables SVM to work in higher-dimensional space, allowing for non-linear decision boundaries [15]. The formula is:

$$W(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \left(\mathrm{x_i^T x_j} \right) \tag{6}$$

The methodology **(see figure 1)** applied to analyze the dataset, identify issues found or their characteristics, and determine the best approach to build the models to obtain better performance and predict the data. **1)** Descriptive analysis to get insights into the dataset. **2)** Splitting the data in training and testing to assess the model's performance. **3)** Identifying the variables with multicollinearity or correlations. **4)** Run all models after eliminating the variables with multicollinearity **5)** Comparing results to select the best model.

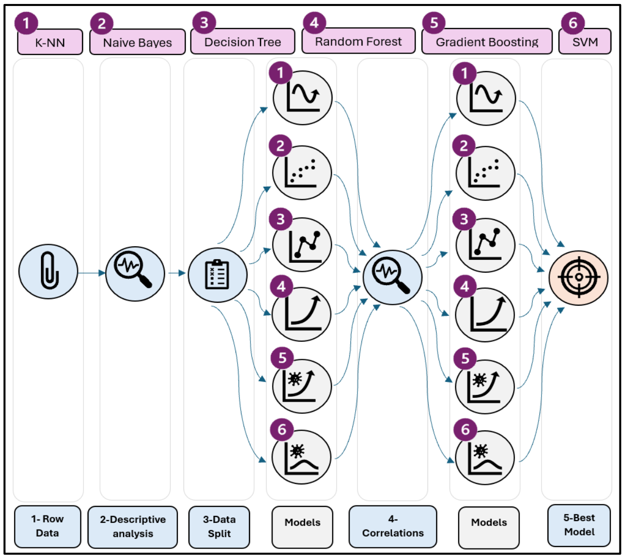


Figure 1: Methodology

4. Experiment and Results

4.1. Raw Data

The dataset used in this work comes from the Kaggle open-source dataset, which includes observations from 4,424 students enrolled in various bachelor's degrees offered at a higher education institution in Portugal (See the variables/features in Table 1). It contains 35 variables/features, which include academic records, sociodemographic information, and macroeconomic data encompassed within the students' context. [16]. Additional information can be found in Table 2.

Table 1: Type of variables in the dataset.

Class of Variable		Variable	Type	Class of Variable		Variable	Type
Demographic data	1	Marital status	Numeric/discrete	**Academic data at enrollment**	18	Application mode	Numeric/discrete
	2	Nacionality	Numeric/discrete		19	Application order	Numeric/ordinal
	3	Displaced	Numeric/binary		20	Course	Numeric/discrete
	4	Gender	Numeric/binary		21	Daytime/evening attendance	Numeric/binary
	5	Age at enrollment	Numeric/discrete		22	Previous qualification	Numeric/discrete
	6	International	Numeric/binary				
				Academic data at the end of 1st semester	23	Curricular units 1st sem (credited)	Numeric/discrete
Socioeconomic data	7	Mother's qualification	Numeric/discrete		24	Curricular units 1st sem (enrolled)	Numeric/discrete
	8	Father's qualification	Numeric/discrete		25	Curricular units 1st sem (evaluations)	Numeric/discrete
	9	Mother's occupation	Numeric/discrete		26	Curricular units 1st sem (approved)	Numeric/discrete
	10	Father's occupation	Numeric/discrete		27	Curricular units 1st sem (grade)	Numeric/continuous
	11	Educational special need	Numeric/binary		28	Curricular units 1st sem (without evaluations)	Numeric/discrete
	12	Debtor	Numeric/binary				
	13	Tuition fees up to date	Numeric/binary	**Academic data at the end of 2nd semester**	29	Curricular units 2nd sem (credited)	Numeric/discrete
	14	Scholarship holder	Numeric/binary		30	Curricular units 2nd sem (enrolled)	Numeric/discrete
					31	Curricular units 2nd sem (evaluations)	Numeric/discrete
Macroeconomic data	15	Unemployment rate	Numeric/continuous		32	Curricular units 2nd sem (approved)	Numeric/discrete
	16	Inflation rate	Numeric/continuous		33	Curricular units 2nd sem (grade)	Numeric/continuous
	17	GDP	Numeric/continuous		34	Curricular units 2nd sem (without evaluations)	Numeric/discrete
				Target/Respon	35	Dropout / Graduate / Enrolled	Categorical

Table 2: Information for students.

Category	Sample	
	n	%
Gender		
Male	1556	35.2
Female	2868	64.8
Displaced	2426	54.8
Educational special needs	51	1.2
Tuition fees up to date (No)	528	11.9
Scholarship holder	1099	24.8
International	110	2.5

The data set covers the period from 2009 to 2019 and includes information from 17 undergraduate programs (Figure 1a). The mean age of students at enrollment was 23, and the majority were single.

4.2. Descriptive Analysis & Visualizations

The dataset preparation involved importing all the libraries necessary to set a base for performing all the analyses. Understanding the type of features, distribution (plots and statistics), meaning, and origin helped us focus our analysis.

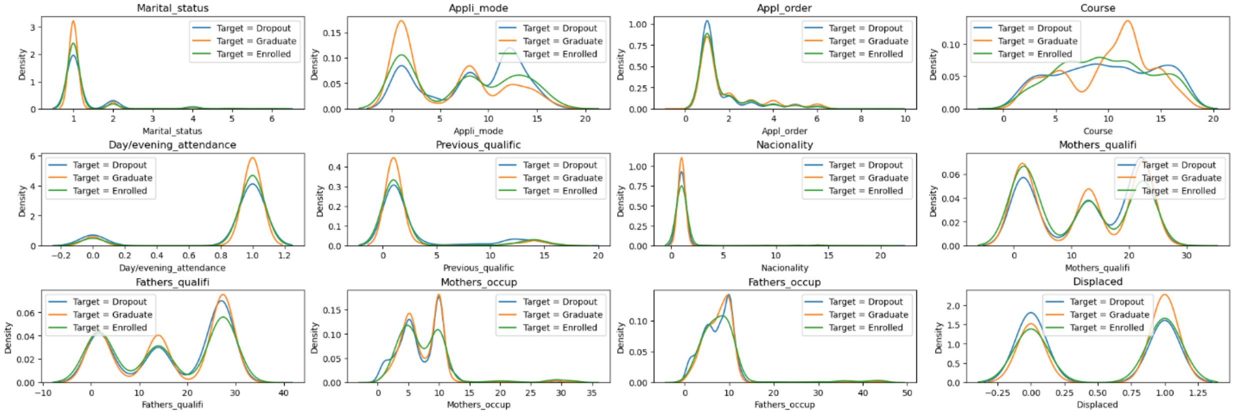
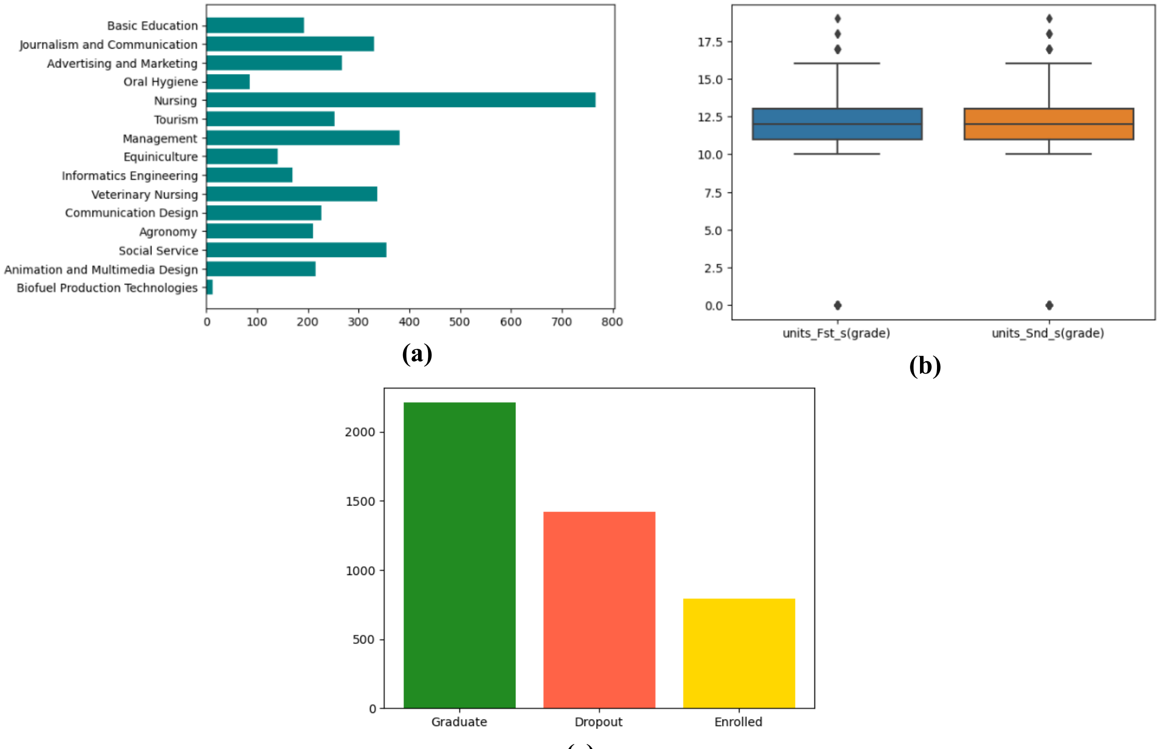


Figure 2: Density plot Target variable by some features

As shown in figure 2, the feature Target (response) classes overlap with all the features. This confirms that this is a classification case because the three target classes (Dropout, Graduate, and Enrolled) are not linearly separable.



(a)

(b)

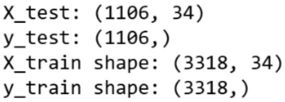
(c)

Figure 3: Descriptive plots of the dataset **(a)** distribution of bachelor's degrees, **(b)** boxplots of grades from 1st semester and 2nd semester, and **(c).** Distribution of target variable.

4.3. Data split

The machine learning process started by separating the raw data into predicted variables (34 as X-variables) and the response (1 as Y-variable). Then, these two groups were again split into training (x-train and y-train) and testing sets (x-test and y-test). Once it was confirmed that the data split shape matched, the modeling process started.

Table 3: Data Shape



```
X_test: (1106, 34)
y_test: (1106,)
X_train shape: (3318, 34)
y_train shape: (3318,)
```

4.4. Initial Models

The hyperparameters belonging to each model were identified to obtain the best performance. For this purpose, a for-loop was developed for all models, excluding Naïve Bayes. After that, it was determined whether the models were overfitted or underfitted by comparing the training and testing scores and plotting them.

As it was the raw data, some results obtained were expected due to the type of features (binary, ordinal, discrete, among others). A cross-validation analysis was also run in some models to check the best model's performance and compare it with other results. See the values in Table 4.

Table 4: Accuracy and Cross Validation for models**.**

Model	Training set	Testing set	Cross-validation
KNN	0.705	0.705	0.699
NB-Gaussian	0.68	0.67	
NB-Bernoulli	0.68	0.69	
DT	0.762	0.741	0.742
RF	--	--	**
GBD	0.792	0.789	0.770
SVM – scale	0.87	0.77	0.763
SVM – rbf	1	0.51	0.763

**Testing results are higher than Training results. The data has yet to be transformed at this stage.

Overall, some good results were obtained at this stage, which shows that some models work well with the full data. However, after transforming the data, this analysis wants to show the effect obtained on the models.

4.5. Data Engineering

As mentioned, for-loops or manual adjustments were performed in most models to obtain the best accuracy on the training and testing sets. However, GridSearch was applied to all the models to ensure the best tradeoff between both sets. Still, it was necessary to extract and transform some features before.

4.5.1. Feature Selection

Previously, when applying the model, it was important to identify which features present a redundancy or dependency. Thus, a correlation analysis was conducted to determine which attributes could be omitted from the prediction. The correlation matrix (Figure 4) shows a significant influence (>0.8) between the academic indicators for Curricular Units (1st and 2nd Semesters) and the features related to the origin of the student (International and Nationality). After careful analysis, it was decided to remove the nationality variable, which contained the same information as whether the student is international. On the other hand, the academic indicators are preserved as they are important measures of academic achievement. Therefore, 33 features are used as predicted variables, plus one feature as a target with three possible categories (Dropout, enrolled, and graduate).

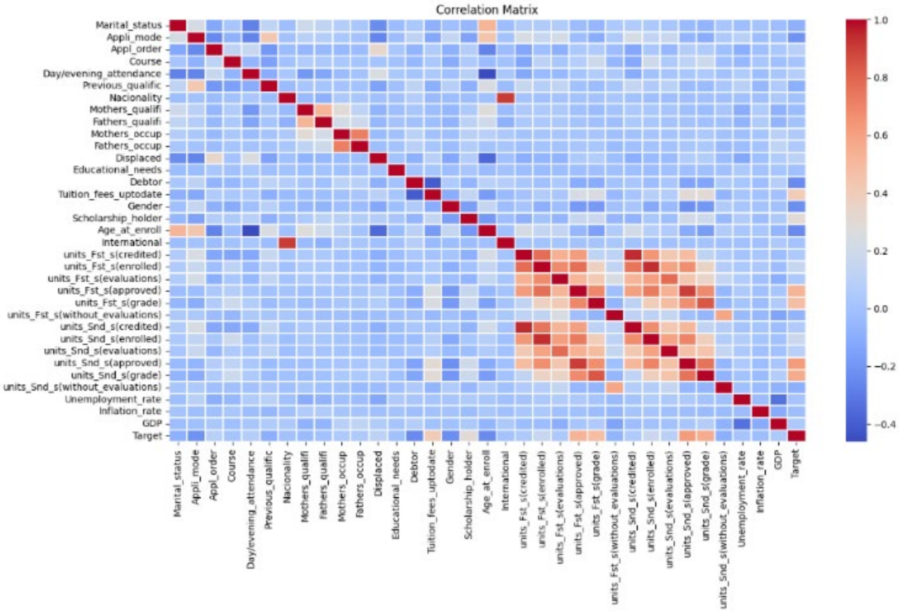


Figure 4: Correlation Matrix for predicted features.

4.5.2. Feature Transformation

The dataset contains mainly discrete data. Thus, the numbers in each column represent a value, but the magnitude does not have any meaning. In this case, the predicted features were scaled using different approaches based on the type of value. For categorical values, one-out-of-N encoding was used. This method adds new features that replace categorical values with 0 and 1. For continuous values, the StandarScaler transformation was used, which ensured that for each feature, the mean is 0 and the variance is 1. The third method applied for ordinal values, the OrdinalEncoder, ensures that the order is preserved. The models are evaluated in two conditions: with raw data (without the transformation mentioned in this paragraph) and then with the data transformed to see if, by doing this, the results can be improved.

4.6. New Models

After running GridSearch, some changes were found in the models. For example, Decision tree, Gradient Boosting, and SVM had improvements. However, Naïve Bayes had a significant decrease in its score for Gaussian analysis but an improvement for Bernoulli as the OnehotEncoder transformation makes categorical values as binary data counting how often every feature of each class is not zero.

Table 5: Comparison of Accuracy before and after transformations

Model		Topic	Result before	Result after	Model		Topic	Result before	Result after
KNN		Training set	0.705	0.705	Gradient Boosting		Training	0.792	0.83
		Testing set	0.705	0.728			Testing	0.789	0.79
		Cross-validation	0.699				Cross-validation	0.794	
Decision Tree		Training set	0.762	0.8	Forest Random		Testing results are higher than Training results. The data has yet to be transformed at this stage.		0.74
		Testing set	0.741	0.755					0.74
Naïve Bayes	Gaussian	Training set	0.68	0.27	SVM	Kernel=rbf	Training set	1	0.89
		Testing set	0.67	0.23			Testing set	0.51	0.81
	Bernoulli	Training set	0.68	0.72		Kernel=scale	Training set	0.87	-
		Testing set	0.69	0.71			Testing set	0.77	-

4.6.1. Evaluation of the model performance

In the dataset, information for all features is present for each student, which is associated with one of the three classes: graduate, enrolled, or dropout. The performance of the models was evaluated with a confusion matrix, classification accuracy, precision, recall, f-score (F1), and area under roc curve (AUC) metrics. The prediction accuracy was evaluated using 5fold cross-validation to estimate the test error in each iteration. The results are given in the Table 6

Table 6: Metrics of the models

Model	Accuracy	Precision	Recall	F1	AUC
KNN	0.731	0.705	0.731	0.703	0.858
NB	0.713	0.733	0.713	0.721	0.861
DT	0.755	0.742	0.755	0.739	0.820
RF	0.740	0.711	0.740	0.705	0.859
GBD	0.794	0.783	0.794	0.786	0.904
SVM	0.807	0.800	0.807	0.801	0.897

The performance comparison of classification algorithms shows that the Support Vector Machine (SVM) achieved the highest accuracy (80.7%) and F1-score (0.801), indicating it as the top-performing model overall. Gradient Boosting (GBD) followed closely with an accuracy of 79.4% and F1-score of 0.786, along with the highest AUC score of 0.904, reflecting excellent classification. The Decision Tree (DT) and Random Forest (RF) models showed moderate performance, with accuracies of 75.5% and 74.0%, respectively, and F1-scores around 0.74. The k-Nearest Neighbors (KNN) and Naive Bayes (NB) algorithms had lower accuracies of 73.1% and 71.3%, respectively, with NB achieving a slightly higher precision (0.733) and KNN having a higher AUC score (0.858). Overall, SVM and GBD demonstrated higher performance, with SVM excelling in accuracy and F1-score, and GBD standing out in AUC. In GBD the most important feature is Units Approved in the second Semester, followed by Tuition Updates, which aligns with the expectations of pursuing the degrees when good academic outcomes are achieved, and financial obligations are up to date (Figure 5).

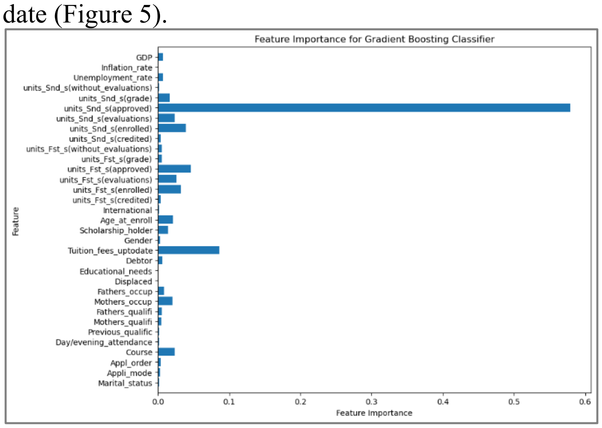


Figure 5: Feature Importance for GBD.

Tables 7 and 8 show the confusion matrix for the GBD and SVM algorithms. The main diagonal of the matrix exhibits the percentage of correctly predicted instances, and the other elements, the percentage of error predicted

Table 7: Confusion matrix of the SVC Model.

Class	Actual	Predicted			F1-score
Dropout	353	**276**	37	20	0.80
Enrolled	193	38	**97**	20	0.56
Graduate	560	39	59	**520**	0.8

Table 8: Confusion matrix of the GBD Model.

Class	Actual	Predicted			F1-score
Dropout	353	**274**	47	22	0.79
Enrolled	193	41	**86**	20	0.51
Graduate	560	38	60	**518**	0.88

Comparing the confusion matrices of SVC and GBD models reveals differences in the classification performance for the three groups. Both models performed well for the Dropout and Graduate classes, achieving similar F1-scores of 0.80 and 0.88 (SVC and GBD, respectively) for Graduates and around 0.79–0.80 for Dropouts. However, for the Enrolled class, both models struggled, with GBD showing slightly lower performance (F1-score of 0.51) than SVC (F1-score of 0.56). The low performance for the Enrolled class highlights the imbalance in the dataset, being that this class is the most challenging for the model. However, the SVC shows better performance for the minority class and suggests a more balanced choice.

5. Conclusion and Future Work

In conclusion, the evaluation of six supervised classification algorithms revealed that Gradient Boosting (GBD) and Support Vector Machine (SVM) models performed better for the dataset, demonstrating strong predictive accuracy and F1-scores, particularly for identifying students at risk of dropping out, which is the main interest. The feature selection process, which reduced the dataset to 33 relevant features, and the application of data scaling significantly improved the overall model performance and mitigated initial underfitting problems. However, both GBD and SVM struggled to accurately classify the Enrolled class, highlighting a dataset imbalance that remains a challenge for these assessed models. Despite this, the proposed approach successfully predicted student dropout risk with an average accuracy of 79%, demonstrating its potential as a valuable tool for early intervention strategies in post-secondary institutions. Further improvements, such as addressing class imbalances and refining model parameters, could enhance its predictive capability in the future.

6. References

[1] "P_R_Dashboard_2022Cohort," Tableau Software. Accessed: Oct. 25, 2024. [Online]. Available: https://public.tableau.com/views/P_R_Dashboard_2022Cohort/PR_Dashboard?:embed=y&:showVizHome=no&:host_url=https%3A%2F%2Fpublic.tableau.com%2F&:embed_code_version=3&:tabs=no&:toolbar=yes&:animate_transition=yes&:display_static_image=no&:display_spinner=no&:display_overlay=yes&:display_count=yes&:language=en-US&:loadOrderID=0

[2] A. W. Astin, *Assessment for excellence: The philosophy and practice of assessment and evaluation in higher education*. Rowman & Littlefield Publishers, 2012.

[3] C. Gibson, S. Rankin, and T. York, "Defining and measuring academic success," *Pract. Assess. Res. Eval.*, vol. 20, no. 5, p. 20, 2015.

[4] R. Los and A. Schweinle, "The interaction between student motivation and the instructional environment on academic outcome: a hierarchical linear model," *Soc. Psychol. Educ.*, vol. 22, no. 2, pp. 471–500, Apr. 2019, doi: 10.1007/s11218-019-09487-5.

[5] A. I. Adekitan and E. Noma-Osaghae, "Data mining approach to predicting the performance of first year student in a university using the admission requirements," *Educ. Inf. Technol.*, vol. 24, pp. 1527–1543, 2019.

[6] O. Zawacki-Richter, V. I. Marín, M. Bond, and F. Gouverneur, "Systematic review of research on artificial intelligence applications in higher education–where are the educators?," *Int. J. Educ. Technol. High. Educ.*, vol. 16, no. 1, pp. 1–27, 2019.

[7] S. Hussain and M. Q. Khan, "Student-Performulator: Predicting Students' Academic Performance at Secondary and Intermediate Level Using Machine Learning," *Ann. Data Sci.*, vol. 10, no. 3, pp. 637–655, Jun. 2023, doi: 10.1007/s40745-021-00341-0.

[8] M. Yağcı, “Educational data mining: prediction of students’ academic performance using machine learning algorithms,” *Smart Learn. Environ.*, vol. 9, no. 1, p. 11, Mar. 2022, doi: 10.1186/s40561-022-00192-z.

[9] L. R. Pelima, Y. Sukmana, and Y. Rosmansyah, “Predicting University Student Graduation Using Academic Performance and Machine Learning: A Systematic Literature Review,” *IEEE Access*, vol. 12, pp. 23451–23465, 2024, doi: 10.1109/ACCESS.2024.3361479.

[10] A. Singh, M. N., and R. Lakshmiganthan, “Impact of Different Data Types on Classifier Performance of Random Forest, Naïve Bayes, and K-Nearest Neighbors Algorithms,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 12, 2017, doi: 10.14569/IJACSA.2017.081201.

[11] N. Salmi and Z. Rustam, “Naïve Bayes classifier models for predicting the colon cancer,” in *IOP conference series: materials science and engineering*, IOP Publishing, 2019, p. 052068.

[12] Y. Lu, T. Ye, and J. Zheng, “Decision Tree Algorithm in Machine Learning,” in *2022 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA)*, Aug. 2022, pp. 1014–1017. doi: 10.1109/AEECA55500.2022.9918857.

[13] P. Probst and A.-L. Boulesteix, “To Tune or Not to Tune the Number of Trees in Random Forest”.

[14] Z. Tian, J. Xiao, H. Feng, and Y. Wei, “Credit Risk Assessment based on Gradient Boosting Decision Tree,” *Procedia Comput. Sci.*, vol. 174, pp. 150–160, Jan. 2020, doi: 10.1016/j.procs.2020.06.070.

[15] D. Isa, L. H. Lee, V. P. Kallimani, and R. RajKumar, “Text Document Preprocessing with the Bayes Formula for Classification Using the Support Vector Machine,” *IEEE Trans. Knowl. Data Eng.*, vol. 20, no. 9, pp. 1264–1272, Sep. 2008, doi: 10.1109/TKDE.2008.76.

[16] “Find Open Datasets and Machine Learning Projects | Kaggle.” Accessed: Oct. 25, 2024. [Online]. Available: https://www.kaggle.com/datasets