10-21-2023

Curbing Dropout: Predictive Analytics at the University of Porto

Luís Blanquet
Universidade do Porto, luis.blanquet@gmail.com

João Grilo
Universidade do Porto, ggrilo2002@gmail.com

Pedro Strecht
Universidade do Porto, pstrecht@uporto.pt

Ana Camanho
Universidade do Porto, acamanho@fe.up.pt

Curbing Dropout: Predictive Analytics at the University of Porto

Luís Blanquet, Universidade do Porto, Portugal, luis.blanquet@gmail.com
João Grilo, Universidade do Porto, Portugal, ggrilo2002@gmail.com
Pedro Strecht, UPdigital, Universidade do Porto, Portugal, pstrecht@uporto.pt
Ana Camanho, Universidade do Porto, Portugal, acamanho@fe.up.pt

Abstract

This study explores data mining techniques for predicting student dropout in higher education. The research compares different methodological approaches, including alternative algorithms and variations in model specifications. Additionally, we examine the impact of employing either a single model for all university programs or separate models per program. The performance of models with students grouped according to their position on the program study plan was also tested. The training datasets were explored with varying time series lengths (2, 4, 6, and 8 years) and the experiments use academic data from the University of Porto, spanning the academic years from 2012 to 2022. The algorithm that yielded the best results was XGBoost. The best predictions were obtained with models trained with two years of data, both with separate models for each program and with a single model. The findings highlight the potential of data mining approaches in predicting student dropout, offering valuable insights for higher education institutions aiming to improve student retention and success.

Keywords: Educational Data Mining; Classification; Academic Performance; Dropout Prediction.

1. INTRODUCTION

In this paper, a thorough comparative analysis of multiple machine learning algorithms is presented to predict student dropout at the University of Porto. The analysis also encompasses a performance evaluation and a comparison between the effectiveness of a single model and models designed for individual programs. Additionally, it explores the alternatives for the number of previous academic years to include in the training dataset, as well as the selection of suitable data treatment methodologies. With the results of this comparative analysis, we aim to empirically increase the understanding of the most effective machine learning algorithms for predicting student dropout. This research also contributed to developing targeted intervention strategies to improve student outcomes at the University of Porto.

This paper is structured as follows: Section 2 reviews the literature on Educational Data Mining; Section 3 details the methodology proposed for the creation of the predictive models; Section 4 presents the results and discusses their implications; Section 5 highlights the main insights, limitations, and future research opportunities following from this study.

ISSN 2183-489X
http://dx.doi.org/10.18803/capsi.v23.143-159

2. LITERATURE REVIEW ON EDUCATIONAL DATA MINING

2.1. Variables used in Education Data Mining studies

Education Data Mining studies with a similar purpose tend to show analogous conclusions regarding variable importance and relevance despite using datasets from different higher education institutions. Furthermore, the process of categorizing the variables used and identifying the clusters of variables most frequently employed for specific problems is important to guide the design of new research informed by the previous findings of the literature.

The study by Strecht et al. (2015), investigated the academic performance of students at the University of Porto and identified a number of demographic variables (age, sex, marital status, nationality, displacement status), along with information on scholarship status, special needs, type of admission, student status, and debt situation as important predictors of academic performance. Moreover, the performance of students in the first year of computer science courses was found to be a significant factor in predicting their academic performance upon completing the degree. A literature review by Alyahyan and Dustegor (2020) focused on factors like prior academic achievement, e-learning activity, psychological attributes, and environmental conditions as independent variables for predicting academic performance. Zimmermann et al. (2011) found that performance in the third year of a B.Sc. program was more effective for predicting future M.Sc. program performance compared to using academic results from the first year. Asif et al. (2014) concluded that socioeconomic data did not significantly contribute to predicting academic performance. Huang and Fang (2013) demonstrated that students' midterm exam grades and previously obtained grades were important predictors of their final exam grades. Table 1 summarizes the most influential categories of factors considered in the literature for predicting academic results in higher education.

2.2. Number of academic years used in historical data

Adekitan and Salau (2019) achieved an 89% accuracy in predicting academic performance for engineering students using three years of academic data. On the other hand, Garg (2018) obtained a 93% accuracy by creating individual models for each degree at Punjab University in Pakistan. The dataset comprised one year of data for 400 students. These studies show that a relatively short time frame of historical data can still be valuable for predicting academic performance. Furthermore, according to Svolba (2022), increasing the amount of data using a longer time frame does not necessarily improve prediction quality, especially for simulation cases with a short optimal length. Consequently, the number of years to consider in historical data remains a topic of debate in the literature, with no consensus, being highly dependent on the specific dataset and project.

Articles	Prior academic achievement	Socio-economic and demographic factors	E-learning activity	Psychological attributes
Adekitan & Salau (2019)	×			
Ahmad et al. (2015)	×	×		
Al-Barrak & Al-Razgan (2016)	×			
Almarabeh (2017)	×			
Anuradha & Velmurugan (2015)	×	×		
Asif et al. (2014)	×			
Garg (2018)	×	×		×
Khalaf et al. (2018)	×			×
Mesaric & Šebalj (2016)	×			
Mohamed & Waguih (2017)	×	×		
Mueen et al. (2016)	×	×	×	×
Aluko et al. (2018)	×			
Putpuek et al. (2018)		×		×
Singh & Kaur (2016)	×	×		
Sivasakthi (2017)	×	×		
Strecht et al. (2015)	×	×		
Yassein et al. (2017)	×			
Total (% in papers reviewed)	**16** (94%)	**9** (53%)	**1** (6%)	**4 (**24%)

Table 1 – Categories of factors used in the literature for the prediction of student academic success

Source: Adapted from Alyahyan & Dustegor (2020).

2.3. Data treatment methodologies

Asif et al. (2014) conducted a study to predict student academic performance at the degree level using data from four academic cohorts consisting of 347 undergraduate students. Data transformation techniques such as normalization, discretization, conversion to numeric values, and combining levels were applied. Additionally, the study used feature selection methods, such as feature and wrapper methods, to identify the most important features. New variables were derived calculating evolutions, like the difference in Grade Point Average (GPA) between consecutive semesters.

A more recent study (Altaf et al., 2019), found that sample size did not significantly affect non-satisfactory accuracy. A feature importance analysis showed that previously obtained grades were the most valuable independent variables for individually trained classifiers, and the study concluded that there was no need to drop courses with small sample sizes. Regarding data balancing, Strecht et

al. (2015) addressed the issue by using stratified sampling to consider the proportion of positive and negative cases in the target variable. Additionally, courses with less than 100 students were not considered and the results were validated using the Friedman test. The challenge of dealing with imbalanced data in prediction models was acknowledged, and various strategies were proposed, including preprocessing techniques such as resampling and feature selection/extraction (Haixiang et al., 2017). Resampling is a technique used to address the issue of imbalanced datasets by rebalancing the sample space and reducing the impact of skewed class distribution during the learning process. There are three main types of resampling methods: over-sampling, under-sampling, and hybrid methods. Over-sampling involves creating new minority class samples, either by randomly duplicating existing minority samples or using methods like Synthetic Minority Over-sampling Technique (SMOTE). Under-sampling, on the other hand, involves discarding some of the majority class samples to achieve a more balanced dataset, typically through random under-sampling. Hybrid methods combine both over-sampling and under-sampling techniques to achieve better data balance and improve the performance of prediction models.

Fewer papers considered feature selection compared to resampling methods (Haixiang et al., 2017). Feature selection reduces the risk of mistaking minority class samples as noise and aims to choose optimal features for classifier performance. Filter and wrapper methods are commonly used and proved to be effective in real-world problems. Table 2 reports the sample size of academic performance research in higher education.

2.4. Algorithms used in data mining applications

Different studies used a variety of classification algorithms including k-Nearest Neighbors (Silverman & Jones, 1989), Random Forest (Breiman, 2001), AdaBoost (Dietterich, 1997), Classification and Regression Trees (Breiman et al., 1984), C5.0 (Salzberg, 1994), Support Vector Machines (Vapnik, 2013), and Naïve Bayes (Lewis et al., 1996). Arguably, for each specific problem, some algorithms may be more suitable than others. For instance, the study of Asif et al (2014) found that Naïve Bayes and Neural Networks were the best algorithms in terms of accuracy for predicting student success using e-learning data, while Neural Networks exhibited strong predictive performance in another study based on Moodle log data (Altaf et al., 2019).

The extensive literature review on Educational Data Mining by Alyahyan & Dustegor (2020) concluded that the four most frequently used algorithms for classification in educational projects are Decision Trees (J48, C4.5, Random tree, REPTree), Bayesian algorithms, Neural Networks and Rule learner's algorithms. These algorithms have consistently demonstrated effective performance in previous studies concerning the prediction of student dropout in academic contexts.

Articles	Prediction level	Sample size
Ahmad et al. (2015)	Curricular year	399
Singh & Kaur (2016)	Curricular year	260
Mesaric & Šebalj (2016)	Curricular year	665
Khalaf et al. (2018)	Degree	161
Al-Barrak & Al-Razgan (2016)	Degree	236
Asif et al. (2014)	Degree	347
Aluko et al. (2018)	Degree	101
Adekitan & Salau (2019)	Degree	1841
Asif et al. (2017)	Degree	210
Mueen et al. (2016)	Course	60
Mohamed & Waguih (2017)	Course	8080
Sivasakthi (2017)	Course	300
Garg (2018)	Course	400
Yassein et al. (2017)	Course	150
Almarabeh (2017)	Course	255

Table 2 – Sample sizes of relevant research on academic performance in higher education

Source: Adapted from Alyahyan & Dustegor (2020).

2.5. Metrics used to assess model quality

Model evaluation is a critical aspect of machine learning applications. Performance measures are key in guiding and assessing classifier learning (Haixiang et al., 2017). Examples of metrics include accuracy, precision, recall, specificity, and Matthews Correlation Coefficient. If the dataset is imbalanced (one of the classes having significantly more examples than the other), then metrics such as balanced accuracy, Kappa, Receiver Operating Characteristics, Area Under the Curve, G-Mean, and the F1-Score are used. Among these, The Kappa statistic (Cohen, 1960) is employed to measure the agreement between predictions and the true class labels. On the other hand, the F1-Score (Chinchor, 1992) combines precision and recall into a single value. Adjusted F-measure and probabilistic thresholding methods (Espíndola & Ebecken, 2005) are also used to account for class imbalance and balance the trade-off between precision and recall.

The choice of the metric used for evaluation significantly affects the perceived effectiveness of a model. Different studies have used a variety of metrics to evaluate the predictive performance of the models. Strecht et al. (2015) used the F1-Score for classification to evaluate the performance of the models with imbalanced datasets, while Altaf et al. (2019) used accuracy and recall for a dataset with different characteristics.

3. METHODOLOGY

This study followed a systematic approach based on the CRoss Industry Standard Platform for Data Mining (CRISP-DM) framework, involving business understanding, data collection, and iterative data preparation phases. Unique models were built using separate datasets, employing popular algorithms for modeling. Performance metrics were collected for evaluation and comparison, offering valuable insights into model effectiveness and suitability.

3.1. Business Understanding

The main goal of this task was to predict student dropout from a higher education program using classification methods. Performance expectations were established by reviewing comparable studies in the literature, with similar goals (predicting dropout as a classification problem) using analogous predictor factors. The key metric of performance reported in these studies was the F1-Score, with its corresponding values presented in Table 3. Subsequently, a comprehensive project plan was formulated to achieve performance levels comparable to previous studies available in the literature.

Article	F1-Score
Kovačić & Nz (2010)	0.62
Aluko et al. (2018)	0.36
Plagge (2013)	0.52
Khalaf et al. (2018)	0.63

Table 3 – F1-Score of studies with similar goals and variables

3.2. Data Understanding

A total of 64 independent variables were extracted from the University of Porto's information system using an Extract, Transform, Load (ETL) process to create the models. These variables cover student-related factors such as personal information, academic background, socioeconomic status, and educational environment. E-learning activity and psychological attributes were not included in the current study but are planned for future work to capture student engagement in courses.

The dataset comprises approximately 100000 observations, spanning 10 academic years of student enrolment data (from 2012 to 2022). Determining the number of academic years for historical data is crucial as, according to the literature review (Svolba, 2022), increasing the dataset size may not necessarily improve prediction quality. Moreover, a larger dataset needs more computational resources and longer training times. Data quality issues were also identified, including an imbalance in the target variable (80/20), high data sparsity, and high-cardinality categorical variables, demanding extensive data preparation efforts.

3.3. Data Preparation

Data pre-selection involved isolating usable independent variables from problematic variables, particularly those with high sparsity or cardinality, such as a high number of missing values or levels. Specific actions, listed in Table 4, were applied to the initial dataset, which resulted in a final dataset including the 25 variables described in Appendix A.

Data cleaning procedures encompassed eliminating inconsistencies and handling missing values, given their potential impact on machine learning algorithm performance and applicability. Techniques such as mode, median, k-Nearest Neighbors, and Random Forest were employed for missing value treatment, selected through model iterations and evaluation metrics. The treatment of outliers was centered on removing significant inconsistencies and major outliers at a significance level of 0.5%.

Cardinality reduction was also addressed, specifying levels in categorical variables to avoid overfitting. Some variables were modified, using the European Credit Transfer and Accumulation System (ECTS) credits instead of years for scaling, and creating simplified categorical variables for application ranking. Other examples are the scientific area of programs, which is categorized according to the national classification of education and training areas (CNAEF) areas, the student status, and the marital status. Z-Score standardization was chosen over min-max due to the presence of minor outliers and the importance of maintaining their impact on highly imbalanced target variables.

Figure 1 shows the results of a variable importance analysis, depicting the different contributions of the 25 predictor variables that have been preserved for the model's predictions, measured by the Mean Decrease Gini metric (measured in thousands). This metric reveals the extent to which each variable shapes the coherence of nodes and leaves within the XGBoost algorithm. Notably, significant factors like age or the priority of students' application ranking hold considerably more importance than variables with a lower Mean Decrease Gini value. The analysis also indicated that a student's financial indebted status might hold significance in predicting dropout (Yuan et al., 2021). These findings provide valuable insights into the most influencing factors of student dropout, paving the way for more effective interventions and support strategies in higher education programs.

Action	Variables (Description in Appendix A)	Rationale
No change	Dropout_following_year, Scholarship_granted, Programme_type, Foreigner, First_enrollment, Dedication_regime, Sex, Displaced, Indebt, Special_needs, Average_prv_year, Weighted_average, Application_ranking, Average_1st_year, Average_12_grade	
Derive	Marital_status → Single (replace) Application_preference → First option (replace) Status → Worker (keep both)	Considerable concentration in mode. Converted to binary.
	Age → Over30 (keep both)	Threshold over 30 years of age (Y/N) AND/OR logarithmic normalization.
Change	Highest_grade, Lowest_grade	Create quartiles for a more accurate representation of distribution.
	Credits_approved, Credits_enrolled_year	Normalize with reference to the expected number of credits.
Reduce cardinality	Programme_scientific_area	Categorized according to the national classification of education and training areas (CNAEF) guidelines.
	Educational_level_parent1, Educational_level_parent2	Categorized according to the national qualifications framework.
	Nationality	Only Portugal and CPLP members are representative (% of students: Portugal – 95.5%; CPLP – 3.9%; Other – 0.6%).
	Admission_regime_name	Categories specified as "general", "reentering", "programme sucessor" and "programme change".
	Occupation_parent1, Occupation_parent2	Relevant categories are "dependent worker" and "freelancer".
	Status	Relevant categories are "ordinary", "worker", "athlete" and "association leader".
Removed from dataset in the pre-selection phase	Curricular_year, Academic_year, Academic_year_admission, Cod_student, Cod_person, Programme_name, Programme_initials, Programme_scientific_area, Extraction_date, Reference_date, Faculty, Last_year, Programme_credits_conclusion	Auxiliary.
	Average_1st_year_1_sem, Credits_enrolled_year_1_sem, Credits_enrolled_year_2_sem	Considered non-relevant due to division in semesters.
	Educational_level_student, Educational_level_parent1, Educational_level_parent2, Profession_student, Average_11_grade, Profession_parent1, Profession_parent2	High percentage of missing values.
	Scholarship_value	High percentage of outliers.
	Programme_degree, Occupation_student, Scholarship_requested, Admission_regime_initials, Birth_country, Courses_approved, Courses_enrolled_year, Courses_enrolled_year_1_sem, Courses_enrolled_year_2_sem, Overdue_courses, Credits_approved, Highest_grade, Lowest_grade	High correlation with another variable.
	Type_student	Low cardinality.
	Behind_years, Average	Variable related to years instead of credits.

Table 4 – Variable pre-selection rationale

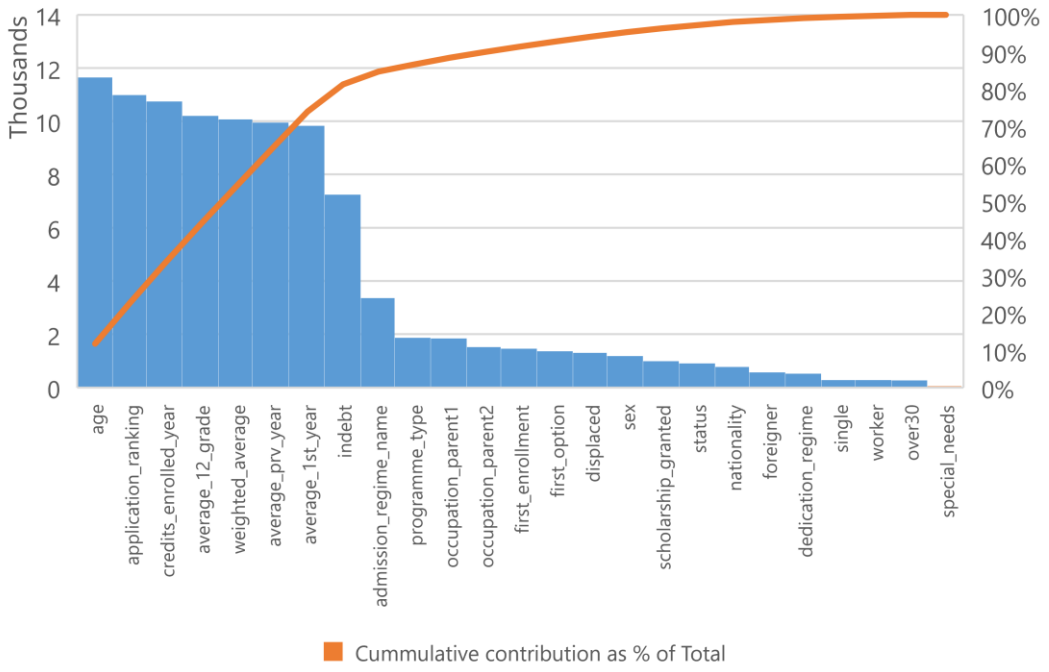

Figure 1 – Pareto Chart of the Variable Importance Analysis (Mean Decrease Gini)

3.4. Dataset separation

In the literature, two main approaches for modeling student data in different academic programs are observed. The first approach involves using separate algorithms for each program set, but this is challenging due to limited data and significant separation between datasets. The second approach includes using a single model with all data or using a model separated by logical separators, such as program and academic year, including first-year students.

Training a single model has multiple benefits: accessing more data, increasing generality, and understanding variable impact across the entire dataset. It also reduces overfitting, bias, and complexity. However, drawbacks include a lack of specificity, potentially overlooking nuances in different aggregations, and challenges with heterogeneous datasets affecting prediction accuracy. To address these drawbacks, the Wide and Deep Model (LeCun et al., 2015) is suggested, capable of accommodating diverse behaviors within the dataset while capturing unique characteristics of each level of aggregation.

Separating models by program acknowledges student diversity and vocational interests. Separating first-year students is justified by their unique data and similarity to high school students, while segregating by academic year allows a comprehensive understanding of each year's distinctive aspects. Despite the advantages, challenges arise when some programs have limited observations (less than 100 enrollments) or imbalanced classes (less than 10% observations from the positive class), impacting model performance and feasibility. Therefore, dividing models by programs may not be fully comprehensive in addressing these limitations.

3.5. Modeling

In the first iteration, a selection of algorithms was tested, encompassing Decision Trees (specifically C5.0), k-Nearest Neighbors, Support Vector Machines, Naïve Bayes, Multilayer Perceptrons (MLPs), and Logistic Regression. Additionally, ensemble methods, including Random Forest, Adaptive Boosting (AdaBoost), Gradient Boosting, Extreme Gradient Boosting (XGBoost), and Stacking (incorporates both XGBoost and MLPs), were also included.

It is advisable not to disregard any algorithms solely based on interpretability, as their performance has not been fully verified. Any potential trade-offs in interpretability might be outweighed by their performance benefits. These algorithms can be broadly classified into white box models like k-Nearest Neighbors, Decision Trees, and Naïve Bayes, which are interpretable and offer insights into predictions, and black box models such as Random Forest, Adaptive Boosting, Support Vector Machines, and Multilayer Perceptrons, which are less interpretable and provide limited reasoning behind their predictions.

3.6. Evaluation

The F1-Score and Kappa are currently the leading metrics used for assessing model predictive performance. However, to conduct a more comprehensive analysis, precision and recall should also be included. Correctly predicting the positive class (a student drops out) is paramount, even with the cost of having more false positives. Nevertheless, a model with high recall alone may not be optimal, as it could predict most cases as positive, resulting in poor overall performance. Hence, it is important to strike a balance between precision and recall while prioritizing recall. In this study, the main metrics used were the F1-Score and Kappa, supplemented by precision and recall for a more detailed view. Additionally, the F2-Score, a variant of the common F1-Score, was also considered. The F2-Score considers both precision and recall but places greater emphasis on recall. It was included due to its adaptability in addressing the specific class imbalance present in the data.

4. RESULTS

The main goal of this study is to evaluate and compare the predictive performance of multiple algorithms for the task at hand. Furthermore, the effectiveness of using a single model versus employing separate models based on different program and year criteria was assessed, considering three scenarios: a separation by program, by the year in the study plan, or separating first-year students from others. To establish a baseline for comparison, a single model was constructed, using eight academic years as the training data and employing sampling techniques such as SMOTE on the training set. Subsequently, the performance of the different algorithms was analyzed and summarized in Table 5.

The algorithms were tested with default hyperparameters. This should be followed by hyperparameter tuning, which may impact the models' performance. Although Support Vector Machines is considered a promising algorithm, its poor scalability led to longer runtimes, preventing an assessment of its performance. XGBoost outperforms all other algorithms, exhibiting higher predictive ability across the F1-Score, Kappa, and F2-Score metrics. Less complex algorithms like Naïve Bayes performed poorly, with considerably lower performance values. Gradient Boosting was excluded from further comparison due to its similarities with XGBoost.

Algorithm	Threshold	F1-Score	Kappa	Precision	Recall	F2-Score
XGBoost	0.130	**0.445**	**0.382**	0.340	0.758	**0.608**
Gradient Boosting	0.200	0.440	0.373	**0.374**	0.559	0.509
MLPs	0.600	0.426	0.356	0.312	0.700	0.551
C5.0	0.200	0.424	0.349	0.371	0.493	0.463
Stacking	0.870	0.411	0.331	0.350	0.500	0.459
Random Forest	0.200	0.398	0.311	0.317	0.533	0.469
AdaBoost	0.500	0.392	0.284	0.263	**0.770**	0.565
Logistic Regression	0.575	0.391	0.284	0.266	0.736	0.544
k-Nearest Neighbors	0.205	0.370	0.254	0.241	0.790	0.543
Naïve Bayes	0.300	0.351	0.254	0.271	0.495	0.425

Table 5 – Algorithm's predictive performance comparison metrics in the single model.

The threshold is the value at which an observation is classified as a positive, typically set to 0.5 in most algorithms. A dynamic threshold is used to enhance model performance. For instance, a threshold of 0.2 classifies observations with a probability of positive class above 0.2 as positive and the remaining ones as negative. The threshold choice is important, as a high threshold identifies critical cases with high dropout probability, while a low threshold is more conservative, capturing potential dropout cases with minimal chances. In XGBoost, a threshold of 0.13 detects observations with a dropout probability above 13% and classifies them accordingly.

Table 6 presents the performance comparison between a single model for all university programs and using separate models for each program. This comparison encompasses students across all academic years or categorizes them based on their position within the study plan.

The single model, based on an aggregated dataset, outperforms the separate models for both first-year students and other years, as well as when considering different curricular years. The model separated by program exhibits superior performance than the single model, resulting in a noteworthy 0.04 improvement in the F1-Score. However, statistical analysis indicates that the difference is not

significant at a 95% confidence level for the F1-Score (p-value of 33.93% > 5.00%), but it is significant for the Kappa measure.

Model	Algorithm	F1-Score	Kappa	F2-Score
Single Model	XGBoost	0.445	0.382	0.608
	MLPs	0.426	0.356	0.551
Models separated by program	XGBoost	0.490 ± 0.108	0.449 ± 0.040	0.638 ± 0.076
	MLPs	0.461 ± 0.110	0.404 ± 0.042	0.620 ± 0.085
Models separated by first-year students	XGBoost	0.419 ± 0.006	0.340 ± 0.004	0.575 ± 0.009
	MLPs	0.423 ± 0.016	0.346 ± 0.007	0.590 ± 0.005
Models separated by the year in the study plan	XGBoost	0.389 ± 0.054	0.294 ± 0.020	0.539 ± 0.073
	MLPs	0.376 ± 0.057	0.274 ± 0.023	0.534 ± 0.078

Table 6 – Comparison between the single and separate models

Performance comparisons of the single model with datasets of varying lengths for training the models are presented in Table 7. Significant differences are observed when analyzing the impact of the number of academic years in the training set. Reducing the training period proves beneficial despite the larger dataset available in the single model. Using two years as the training period yields the best results, with a F1-Score of 0.511 for MLPs compared to 0.407 for eight years. The single model does not present standard deviation unless multiple experiments are conducted. Considering the volatility observed in the models separated by program, with a Kappa standard deviation of 0.04, the 2-year model shows statistical superiority over the 8-year model (p-value of 0.03% < 5.00%).

Number of academic years	Algorithm	F1-Score	Kappa	F2-Score
8 academic years	XGBoost	0.394	0.302	0.580
	MLPs	0.407	0.321	0.562
6 academic years	XGBoost	0.423	0.346	0.579
	MLPs	0.400	0.310	0.557
4 academic years	XGBoost	0.458	0.399	0.608
	MLPs	0.413	0.330	0.569
2 academic years	XGBoost	0.495	0.456	0.648
	MLPs	0.511	0.481	0.643

Table 7 – Comparison between the number of academic years in the training dataset used in the single model

Table 8 presents the performance of the single model applied to all university programs with different sampling techniques, including no sampling, hybrid sampling (SMOTEENN), and oversampling (SMOTE).

Considering data imbalance, assessing different balancing techniques is crucial. Surprisingly, oversampling performs worse compared to not using any balancing technique in this specific case, likely due to the dataset's already large size and complexity. The effectiveness of SMOTEEN varies depending on the case, making it challenging to draw definitive conclusions regarding the superiority of any specific balancing technique, especially since a moving threshold is used. The stability of metrics in the single model outweighs the performance variability seen in different models, which can have a standard deviation of up to 30% from the mean. However, the models separated by program shows high performance, which can be advantageous for certain programs. Interestingly, reducing the number of academic years in the training set positively impacts model performance by reducing complexity. This finding contradicts the approach of separating models by academic year, which requires a larger number of observations to function effectively.

Sampling technique	Algorithm	F1-Score	Kappa	F2-Score
No sampling technique	XGBoost	0.411	0.328	0.572
	MLPs	0.414	0.332	0.586
Hybrid (SMOTEENN)	XGBoost	0.420	0.341	0.587
	MLPs	0.383	0.284	0.559
Oversampling (SMOTE)	XGBoost	0.395	0.303	0.577
	MLPs	0.388	0.292	0.558

Table 8 – Comparison between sampling techniques in the single model

It is important to acknowledge that a single model may not always be the best choice. When there is a substantial number of observations available, using separate models that perform better for specific programs can be beneficial. One potential improvement for the current study is to automatically select models based on their performance and stability for each program. This approach would involve choosing either a single model with fewer years or a model specifically tailored for a particular program, depending on their respective performances. Additionally, combining both approaches can address the issue of programs that do not meet minimum requirements for the separate models.

5. CONCLUSIONS AND FURTHER RESEARCH OPPORTUNITIES

Different models have strengths and weaknesses, and their performance varies in different scenarios. In this case, the model using only the two most recent years for training and using models separated by program performed better than the baseline. Combining these approaches through an ensemble

method could enhance overall predictive performance. However, no definitive conclusions were reached regarding data sampling techniques.

One limitation of this study is that the algorithm hyperparameters were not thoroughly tuned using methods like GridSearch or Bayesian Optimization. The initial goal was to identify the top-performing models, leaving fine-tuning for a subsequent stage to improve performance metrics. However, it is worth noting that this process might potentially change the initially best-performing algorithm.

Conducting robustness tests would also strengthen the findings. The dataset showed high sparsity, particularly in critical variables like prior academic performance, impacting the models' overall performance. With a more comprehensive dataset, especially in these essential variables, the results would be more robust and reliable. There is plenty of room for improvement, such as incorporating e-learning activity or psychological attributes from new data sources or deriving new variables like pandemic year indicators. Reducing data sparsity during data collection would be crucial to mitigate its impact on the study results.

In conclusion, the implemented models will be available in a portal at the University of Porto, providing access to a diverse range of stakeholders. Dropout prediction models will be complemented by academic success/failure prediction models at the course level. Leveraging these resources will enable proactive measures to prevent dropout and enhance student performance, fostering a supportive and successful academic environment.

REFERENCES

Adekitan, A. I., & Salau, O. (2019). The impact of engineering students' performance in the first three years on their graduation result using educational data mining. *Heliyon*, *5*(2), e01250.

Ahmad, F., Ismail, N. H., & Aziz, A. B. A. (2015). The Prediction of Students' Academic Performance Using Classification Data Mining Techniques. *Applied mathematical sciences*, *9*, 6415-6426.

Al-Barrak, M., & Al-Razgan, M. (2016). Predicting Students Final GPA Using Decision Trees: A Case Study. *International Journal of Information and Education Technology*, *6*, 528-533.

Almarabeh, H. (2017). Analysis of Students' Performance by Using Different Data Mining Classifiers. *International Journal of Modern Education and Computer Science*, *9*, 9-15.

Altaf, S., Soomro, W., & Rawi, M. I. M. (2019). *Student Performance Prediction using Multi-Layers Artificial Neural Networks: A Case Study on Educational Data Mining* Proceedings of the 2019 3rd International Conference on Information System and Data Mining, Houston, TX, USA. Doi: 10.1145/3325917.3325919

Aluko, O., Daniel, E., Oshodi, O., Aigbavboa, C., & Abisuga, O. (2018). Towards reliable prediction of academic performance of architecture students using data mining techniques. *Journal of Engineering Design and Technology*, *16(3), 385-397*. Doi: 10.1108/JEDT-08-2017-0081

Alyahyan, E., & Dustegor, D. (2020). Predicting Academic Success in Higher Education Literature Review and Best Practices. *International Journal of Educational Technology in Higher Education*, *17, 1-21*. Doi: 10.1186/s41239-020-0177-7

Anuradha, C., & Velmurugan, T. (2015). A Comparative Analysis on the Evaluation of Classification Algorithms in the Prediction of Students Performance. *Indian journal of science and technology*, *8*(15), 1-12.

Asif, R., Merceron, A., Abbas, D.-S., & Haider, N. (2017). Analyzing undergraduate students' performance using educational data mining. *Computers & Education*, *113*, 177-194.

Asif, R., Merceron, A., & Pathan, M. (2014). Predicting Student Academic Performance at Degree Level: A Case Study. *International Journal of Intelligent Systems and Applications*, *7*, 49-61. Doi: 10.5815/ijisa.2015.01.05

Breiman, L. (2001). Random Forests. *Machine Learning*, *45*(1), 5-32. Doi: 10.1023/A:1010933404324

Breiman, L., Friedman, J., Stone, C. J., & Olshen, R. A. (1984). *Classification and Regression Trees*. Taylor & Francis.

Chinchor, N. (1992). *MUC-4 evaluation metrics* Proceedings of the 4th conference on Message understanding, McLean, Virginia. Doi: 10.3115/1072064.1072067

Cohen, J. (1960). A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, *20*, 37 - 46.

Dietterich, T. G. (1997). Machine-Learning Research. *AI Magazine*, *18*(4), 97. Doi: 10.1609/aimag.v18i4.1324

Espíndola, R., & Ebecken, N. (2005). On extending f-measure and g-mean metrics to multi-class problems. *Sixth international conference on data mining, text mining and their business applications*, *35*, 25-34.

Garg, R. (2018). Predicting student performance of different regions of Punjab using classification techniques. *International Journal of Advanced Research in Computer Science*, *9*, 236-240. Doi: 10.26483/ijarcs.v9i1.5234

Haixiang, G., Yijing, L., Shang, J., Mingyun, G., Yuanyue, H., & Bing, G. (2017). Learning from class-imbalanced data: Review of methods and applications. *Expert Systems with Applications*, *73*, 220-239. Doi: 10.1016/j.eswa.2016.12.035

Huang, S., & Fang, N. (2013). Predicting student academic performance in an engineering dynamics course: A comparison of four types of predictive mathematical models. *Computers & Education*, *61*, 133-145. Doi: 10.1016/j.compedu.2012.08.015

Khalaf, A., Hashim, A., & Akeel, W. (2018). Predicting Student Performance in Higher Education Institutions Using Decision Tree Analysis. *International Journal of Interactive Multimedia and Artificial Intelligence*, *5*, 26-31. Doi: 10.9781/ijimai.2018.02.004

Kovačić, Z., & Nz. (2010). Early Prediction of Student Success: Mining Students Enrolment Data.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436-444. Doi: 10.1038/nature14539

Lewis, D., Info, C., Studies, L., & Ringuette, M. (1996). A Comparison of Two Learning Algorithms for Text Categorization. *Third Annual Symposium on Document Analysis and Information Retrieval*.

Mesaric, J., & Šebalj, D. (2016). Decision trees for predicting the academic success of students. *Croatian Operational Research Review*, *7*, 367-388. Doi: 10.17535/crorr.2016.0025

Mohamed, M. H., & Waguih, H. (2017). Early Prediction of Student Success Using a Data Mining Classification Technique. *International Journal of Science and Research, 6(10), 126-131*.

Mueen, A., Zafar, B., & Manzoor, U. (2016). Modeling and Predicting Students' Academic Performance Using Data Mining Techniques. *International Journal of Modern Education and Computer Science*, *11*, 36-42. Doi: 10.5815/ijmecs.2016.11.05

Plagge, M. (2013). *Using artificial neural networks to predict first-year traditional students second year retention rates* Proceedings of the 51st ACM Southeast Conference, Savannah, Georgia. Doi: 10.1145/2498328.2500061

Putpuek, N., Rojanaprasert, N., Atchariyachanvanich, K., & Thamrongthanyawong, T. (2018). Comparative Study of Prediction Models for Final GPA Score: A Case Study of Rajabhat Rajanagarindra University. *2018 IEEE/ACIS 17th International Conference on Computer and Information Science (ICIS)*, 92-97.

Salzberg, S. L. (1994). C4.5: Programs for Machine Learning by J. Ross Quinlan. Morgan Kaufmann Publishers, Inc., 1993. *Machine Learning*, *16*(3), 235-240. Doi: 10.1007/BF00993309

Silverman, B. W., & Jones, M. C. (1989). E. Fix and J.L. Hodges (1951): An Important Contribution to Nonparametric Discriminant Analysis and Density Estimation: Commentary on Fix and Hodges (1951). *International Statistical Review / Revue Internationale de Statistique*, *57*(3), 233-238. Doi: 10.2307/1403796

Singh, W., & Kaur, P. (2016). Comparative Analysis of Classification Techniques for Predicting Computer Engineering Students' Academic Performance. *International Journal of Advanced Research in Computer Science*, *7*(6), 31-36.

Sivasakthi, M. (2017). *Classification and prediction based data mining algorithms to predict students introductory programming performance*. In 2017 International Conference on Inventive Computing and Informatics (ICICI) (pp. 346-350). IEEE, Coimbatore, India.

Strecht, P., Cruz, L., Soares, C., Mendes Moreira, J., & Maranhão, R. (2015). *A Comparative Study of Regression and Classification Algorithms for Modelling Students Academic Performance* Proceedings of the 8th International Conference on Educational Data Mining, EDM 2015, Madrid, Spain, June 26-29, 2015,

Svolba, G. (2022). *Determining the best length of the history of your timeseries data for timeseries forecasting*. MLearning.ai. https://medium.com/mlearning-ai/determining-the-best-length-of-the-history-of-your-timeseries-data-for-timeseries-forecasting-f8600a3c086_(19 de maio de 2023)

Vapnik, V. (2013). *The Nature of Statistical Learning Theory*. Springer New York.

Yassein, N., Helali, R., & Mohomad, S. (2017). Predicting Student Academic Performance in KSA using Data Mining Techniques. *Journal of Information Technology & Software Engineering*, *07*(5), 1-5. Doi: 10.4172/2165-7866.1000213

Yuan, Y., Wu, L., & Zhang, X. (2021). Gini-Impurity Index Analysis. *IEEE Transactions on Information Forensics and Security*, *16*, 3154-3169.

Zimmermann, J., Brodersen, K. H., Pellet, J.-P., August, E., & Buhmann, J. (2011). Predicting Graduate-level Performance from Undergraduate Achievements. In 4th International Conference on Educational Data Mining (EDM) (pp. 357-358).

APPENDIX A. VARIABLE DESCRIPTION

Name	Description
Sex	Gender of the student
Single	Is the student single?
Age	Age of the student
Over30	Is the student above 30 years of age?
Nationality	Nationality of the student
Foreigner	Is the student a foreigner?
Displaced	Is the student displaced?
Special_needs	Does the student have special educational needs?
Occupation_parent1	Parent 1 main occupation
Occupation_parent2	Parent 2 main occupation
Status	Student status in the academic year
Worker	Is the student a worker?
Programme_type	Type of the programme in which the student is enrolled
Admission_regime_name	Name of the admission regime in the programme
First option	Was the student admitted in their first-option program?
Application_ranking	Application ranking within admission in the programme
Average_12_grade	Average of the 12th year of secondary education
Dedication_regime	Student dedication regime in the academic year
Scholarship_granted	Was the scholarship application granted during the academic year?
Indebt	Is the student indebted to the institution for the academic year?
First_enrollment	Is it the student's first enrollment in the course?
Weighted_average	Average credit-weighted grade in approvals from previous academic years
Average_prv_year	Average grade in approvals from the previous academic year
Average_1st_year	Average grade in approvals from the academic year of admission
Credits_enrolled_year	Total credits the student is enrolled in the academic year
Dropout_following_year	Did the student dropout of the programme the following academic year?

Table A.1 – Description of variables after pre-selection