applied sciences

MDPI

Article

Addressing Class Imbalance in Predicting Student Performance Using SMOTE and GAN Techniques

Fatema Mohammad Alnassar [1,*], Tim Blackwell [1], Elaheh Homayounvala [2] and Matthew Yee-king [1]

[1] Department of Computing, Goldsmiths, University of London, London SE14 6NW, UK
[2] Department of Computing and Digital Media, London Metropolitan University, London N7 8DB, UK
* Correspondence: falna001@gold.ac.uk

Abstract

Virtual Learning Environments (VLEs) have become increasingly popular in education, particularly with the rise of remote learning during the COVID-19 pandemic. Assessing student performance in VLEs is challenging, and the accurate prediction of final results is of great interest to educational institutions. Machine learning classification models have been shown to be effective in predicting student performance, but the accuracy of these models depends on the dataset's size, diversity, quality, and feature type. Class imbalance is a common issue in educational datasets, but there is a lack of research on addressing this problem in predicting student performance. In this paper, we present an experimental design that addresses class imbalance in predicting student performance by using the Synthetic Minority Over-sampling Technique (SMOTE) and Generative Adversarial Network (GAN) technique. We compared the classification performance of seven machine learning models (i.e., Multi-Layer Perceptron (MLP), Decision Trees (DT), Random Forests (RF), Extreme Gradient Boosting (XGBoost), Categorical Boosting (CATBoost), K-Nearest Neighbors (KNN), and Support Vector Classifier (SVC)) using different dataset combinations, and our results show that SMOTE techniques can improve model performance, and GAN models can generate useful simulated data for classification tasks. Among the SMOTE resampling methods, SMOTE NN produced the strongest performance for the RF model, achieving a Region of Convergence (ROC) Area Under the Curve (AUC) of 0.96 and a Type II error rate of 8%. For the generative data experiments, the XGBoost model demonstrated the best performance when trained on the GAN-generated dataset balanced using SMOTE NN, attaining a ROC AUC of 0.97 and a reduced Type II error rate of 3%. These results indicate that the combined use of class balancing techniques and generative synthetic data augmentation can enhance student outcome prediction performance.

Keywords: virtual learning environment; student grade prediction; machine learning; SMOTE; generative adversarial networks (GANs); synthetic data

Academic Editor: Keun Ho Ryu

Received: 15 December 2025
Revised: 10 February 2026
Accepted: 25 February 2026
Published: 28 March 2026

1. Introduction

Over the last fifty years, technology has had a significant impact on the educational sector, with the introduction of various electronic gadgets, software systems, and internet-based tools [1]. Virtual learning through online platforms, such as Virtual Learning Environments (VLEs), has emerged as a potential means of education due to technological advancements and easy access to the internet [2]. The recent COVID-19 pandemic has further highlighted the importance of virtual education, as most educational institutes have shifted from on-campus to online teaching [3]. However, the reliable and effective

assessment of students' performance in VLEs is challenging due to the vulnerability of online assessments to cheating and difficulty in grading [4]. Educational Data Mining (EDM) has emerged as an effective approach to extract useful information and patterns from the huge educational database obtained from students' use of VLEs [5].

The adoption of VLEs has made it necessary to evaluate students' results in such environments, as they may differ from regular learning styles, mainly due to location and time differences [6]. Therefore, predicting students' performance in various courses and programs is essential for improving educational outcomes and finding patterns of VLE usage [7]. Artificial Intelligence learning-based approaches, such as classification and regression, have been found effective in predicting student performance based on data collected through VLEs. However, the effectiveness and accuracy of prediction depend on the data type of features being used, the dataset size, and diversity in the dataset.

The performance prediction of students using machine learning models is often hindered by class imbalance in the educational datasets. The problem of class imbalance in machine learning arises due to various reasons, such as limitations in data collection, the use of datasets from diverse demographics, and the participation of human subjects. The majority of machine learning models are designed to work with datasets that have balanced classes, which can lead to biased training if the dataset is imbalanced. While class imbalance is a common issue in education-related datasets, there is a lack of research on this topic, particularly in predicting students' performance. Although solutions to class imbalance have been proposed in other contexts, there is a gap in the literature on addressing this problem in the context of predicting students' performance. Moreover, the use of VLEs with machine learning models introduces the challenge of developing trustworthy classifiers in the face of class imbalance. While existing methods address the problem of relative imbalance, they do not address the issue of an unbalanced distribution among classes when data for model training is scarce. This presents an opportunity to explore and compare various class imbalance solutions in the context of student performance prediction.

This paper proposes an experimental design to address class imbalance in predicting student performance by utilizing the Synthetic Minority Over-sampling Technique (SMOTE) and Generative Adversarial Network (GAN) technique. The proposed research makes significant contributions to the field of education and machine learning. First, it addresses the common problem of class imbalance in educational datasets and provides an experimental design that effectively addresses this issue. Second, it compares the classification performance of seven machine learning models (i.e., Multi-Layer Perceptron (MLP), Decision Trees (DT), Random Forests (RF), Extreme Gradient Boosting (XGBoost), Categorical Boosting (CATBoost), K-Nearest Neighbors (KNN), and Support Vector Classifier (SVC)), which can guide educational institutions in selecting the best model for their needs. Third, the study demonstrates the usefulness of SMOTE and GAN techniques in improving model performance and generating new data for classification tasks. Finally, the study provides insights into the importance of dataset quality and diversity in achieving accurate predictions of student performance. Overall, the proposed research has the potential to inform and guide educational institutions seeking to improve their student performance prediction accuracy using machine learning models.

2. Related Work

2.1. Prediction of Student Performance

In recent years, there has been a growing interest in using machine learning and artificial intelligence techniques for predicting student performance. The focus of these studies has been on developing models that can accurately predict student performance based on various factors, such as past academic performance, interaction with learning manage-

ment systems, and course-related activities. This section presents the review of literature where AI and ML are used for students' performance prediction. Literature is presented in chronological order to highlight the shift in trends over the years in this domain.

Elbadrawy et al. [8] proposed collaborative linear multi-regression models to predict continuous student performance scores in course activities using historical grades, Learning Management System (LMS) interactions, and course-related features. Evaluated on a large-scale dataset comprising 11,556 student records across 832 courses, their approach achieved an RMSE of 0.147, outperforming single regression models and demonstrating the benefit of modeling inter-course relationships. Focusing on collaborative learning environments, Yee-King et al. [9] employed a KNN model to predict categorical student grades represented as grade bands. Their results showed classification accuracies of 88%, 77%, and 31% for 2-, 3-, and 10-band grade categorizations, respectively. Despite promising results for coarse-grained prediction, the study lacked comparative evaluation against existing approaches in the literature.

Several works investigated classical classifiers for predicting exam outcomes. Al-Shehri et al. [10] applied KNN and Support Vector Machines (SVM) to predict binary student final exam outcomes (pass/fail) using data from the University of Minho, Portugal. Their experiments showed that SVM slightly outperformed KNN, achieving accuracies of 96% and 95%, respectively. Similarly, Iqbal et al. [11] framed grade prediction as a continuous regression task, comparing Collaborative Filtering (CF), Matrix Factorization (MF), and Restricted Boltzmann Machines (RBM). Using student data from the International Technical University (ITU), Pakistan, they reported that RBM achieved the best performance with an RMSE of 0.3. The role of student engagement in predicting academic outcomes has also received significant attention. Hussain et al. [12] analyzed engagement indicators using DT, CART, JRIP, Gradient Boosting Trees (GBT), and Naive Bayes Classifiers (NBC) to predict categorical assessment outcomes on Open University (OU) data. Their results indicated that the J48 decision tree achieved the highest accuracy (88.52%) and recall (93.4%). In a related effort, Heuer and Breiter [13] focused on identifying at-risk students as a binary classification problem using SVM, Naive Bayes, Random Forest, XGBoost, and Logistic Regression on the OULAD dataset, where SVM achieved the best accuracy of 87.98%.

With the increasing availability of temporal and large-scale educational data, more advanced learning models have been explored. Sekeroglu et al. [7] examined both continuous student performance prediction and categorical performance classification using a variety of models, including Long Short-Term Memory (LSTM), Backpropagation (BP), Support Vector Regression (SVR), SVM, and Gradient Boosting Classifiers. Their findings showed that SVR performed best for regression tasks, while BP achieved superior results for classification. Deep-learning-based approaches have further improved prediction accuracy in recent studies. El Fouki et al. [14] proposed a multidimensional framework combining deep neural networks with Principal Component Analysis (PCA) to classify categorical student performance levels, achieving an accuracy of 92.54% on a custom dataset. Hussain et al. [15] developed a deep learning model for predicting categorical internal assessment outcomes, outperforming Artificial Immune Recognition System (AIRS) v2.0 and AdaBoost with an accuracy of 95.34% on 10,140 student records.

Ensemble and comparative modeling approaches have also been widely studied. Ajibade et al. [16] evaluated DT, KNN, and SVM for categorical student performance classification, reporting that Decision Trees performed best, while an ensemble model further improved accuracy to 91.5%. Tomasevic et al. [17] conducted a comprehensive comparison of supervised learning techniques, including KNN, SVM, Artificial Neural Networks (ANN), Decision Trees, Bayesian Linear Regression (BLR), and Regularized Linear Regression (RLR), for categorical exam performance prediction. Their results showed

that ANN achieved the highest F1-score (96.62%) using engagement and performance features, while SVM with an RBF kernel reached 96.04% when demographic data were incorporated. More recent studies have explored behavioral and deep learning perspectives. Hooshyar et al. [18] introduced a procrastination-aware model for predicting categorical student performance outcomes, with Linear SVM achieving an accuracy of 95%. Similarly, Waheed et al. [19] applied Deep Neural Networks (DNN) to the OULAD dataset to predict categorical academic performance, demonstrating superior performance over traditional regression and SVM models with an accuracy of 93%.

Educational datasets with features of student access patterns, availability of course design, different teaching styles, and student activities have been used to predict the performance of students. Some standard datasets being used by researchers for performance prediction include OU, OULAD, SPD and SAPD. Machine learning and AI have emerged as a key role in exploiting the educational datasets in comparison to conventional statistical approaches. Machine learning approaches used by researchers include *k*-NN, DT, CART, JRIB, GBT, NBC, LSTM, BP, SVR, SVM, GBC, DNN, MLP, PCA, BayesNet, ANN, BLR, RLR, CF, MP, RBM, RF, XGBoost, LR, AIRS, AdaBoost, L-SVM, R-SVM, GP, NB, and Ensemble. A shift has been observed in the literature from conventional machine learning approaches (i.e., SVR, DT, GBT, PCA, BLR, RLR, NB, CF, MP, XGBoost, LR) towards deep learning approaches (i.e., DNN, MLP, ANN, LSTM). However, the availability of training datasets for deep learning approaches has been one of major shortcomings to date. Given the exponential rise in the use of VLE due to COVID-19, it is expected to have huge datasets available soon that are suitable for deep learning approaches in the future. F1-Score, accuracy, recall score and RMSE are reported to be commonly used evaluation measures for trained machine learning models.

2.2. Class-Imbalance Problem in Educational Settings

In educational data mining and predictive analytics, the class-imbalance problem arises when the distribution of target classes is highly skewed, such that minority classes are underrepresented relative to majority classes. This imbalance often leads to biased predictive models that maximize overall accuracy at the expense of minority-class detection, which is typically the critical outcome of interest in educational interventions. Consequently, addressing class imbalance has become a focal research challenge within the educational machine learning community.

Early efforts specifically recognizing imbalance in educational datasets demonstrated that standard classifiers perform poorly when applied to skewed academic outcomes. For example, Barros et al. [20] investigated school dropout prediction, showing that predictive performance significantly improved when data balancing techniques such as downsampling, SMOTE, and ADASYN were applied in combination with classifiers like DTs, NNs, and Balanced Bagging. Their results indicated that balanced training sets yielded more reliable geometric mean and UAR metrics compared to imbalanced models that over-emphasized the majority class. Subsequent work confirmed that oversampling and resampling strategies can substantially enhance prediction quality for imbalanced educational targets. Mduma [21] examined a range of balancing techniques including Random Oversampling, Random Undersampling, SMOTE, SMOTE with Edited Nearest Neighbors, and SMOTE with Tomek Links applied to dropout prediction using LR, FR, and MLP models. Results indicated that hybrid resampling methods, particularly SMOTE with Edited Nearest Neighbors, achieved superior classification performance for the minority class. Similarly, Wongvorachan et al. [22] compared random oversampling, random undersampling, and hybrid SMOTE-NC with undersampling on a large longitudinal educational

dataset, concluding that random oversampling was effective under moderate imbalance while hybrid strategies performed better under extreme imbalance ratios.

In addition to task-specific empirical studies, broader surveys of imbalance learning have influenced educational applications by categorizing approaches and highlighting their suitability for imbalanced prediction problems. Chen et al. [23] provided a comprehensive survey of state-of-the-art imbalanced learning techniques, categorizing methods into data-level resampling (e.g., ROS, RUS, SMOTE, and variants), algorithm-level strategies (e.g., cost-sensitive learning), and hybrid methods that combine resampling with ensemble frameworks such as RUSBoost and EasyEnsemble. Altalhan et al. [24] extended this review by emphasizing recent developments in hybrid and deep imbalance techniques, identifying oversampling variants and ensemble schemes as particularly promising for complex real-world datasets that include educational contexts alongside other domains.

Several studies in educational prediction have explicitly examined the effects of class imbalance on model performance. For instance, a study by El-Deeb et al. [25] on academic performance prediction empirically evaluated the impact of SMOTE on a range of classifiers including RF, KNN, NB, and SVM, showing that SMOTE often enhanced performance metrics such as recall and area under the ROC curve compared to imbalanced baselines. Complementing this, research by Alija et al. [26] on student performance prediction that integrates feature selection with resampling demonstrated that SMOTE, combined with optimization-based wrapper methods, improved recall and ROC for Random Forest and other supervised learners.

More recent applied research has expanded imbalance handling to include advanced oversampling techniques and ensemble frameworks tailored for educational outcomes. Wang and Yao [27]'s ensemble integration of resampling methods in general machine learning has broader implications for education, where ensemble-SMOTE variants and bagging strategies have been shown to reduce bias toward the majority class while preserving classifier diversity. Moreover, the state-of-the-art educational dropout prediction system proposed by Jain et al. [28] has applied adaptive oversampling, such as ADASYN and SMOTE in combination with tree-based models like RF and XGBoost, to demonstrate significant gains in F1 and precision metrics for minority classes.

These studies underscore that class imbalance is a pervasive challenge in educational prediction tasks and that statistical resampling, cost-sensitive learning, and ensemble methods are effective countermeasures.

3. Coursera Dataset

The Coursera dataset is a benchmark dataset that was collected from online courses taught at the computer science department of Stanford University. Coursera is an online platform developed by Andrew Ng and Daphne Koller at Stanford, in collaboration with over 160 universities worldwide [29]. The platform offers a vast number of courses (i.e., 5100) and has registered over 77 million students. The Coursera data used in this study were obtained from the ethics committee of the University, and included data from eight courses offered by the Computer Science department, spanning from 2018 to 2019. The dataset comprises nine key groups of information, including course information, course contents, course progress, assessments, course grades, discussions, feedback, learners, and demographics. The course information section provides basic information on the courses offered, such as their names and the sessions they are taught in, among others. The dataset covers eight courses, including Algorithm and Data Structure, Computational Mathematics, Discrete Mathematics, Fundamentals of Computer Science, How Computers Work, Introduction to Programming I, Introduction to Programming II, and Web Development. The course progress section describes the interactions between the learners and the course

content, while the assessments section provides detailed information about the learners' interactions with the assessments. The course grades section details the learners' grades and their passing states within the courses. The course contents section refers to the materials used in the courses, including modules, lessons, items, and mapping to specializations. The discussions section contains forums, forum posts, and vote information, while the feedback section contains information about the user ratings of course content and courses. The learner section describes the learners' information, such as when and where they joined Coursera, and the demographics section contains demographic data based on user surveys.

The baseline Coursera dataset used in this study consists of 7,521 samples with selected nine input features, namely *hits_count*, *partic_count*, *video_duration*, *assessment_type_id_6*, *assessment_type_id_7*, *video_count*, *quiz_count*, and *total_quiz_grade*, with *Course Passed* serving as the binary target variable. Among the total samples, 5089 instances belong to the Fail (0) class, while 2432 instances correspond to the Pass (1) class, as illustrated in Figure 1. To ensure robust and unbiased performance evaluation, a fixed training–validation split was not adopted. Instead, all experiments were conducted using a five-fold cross-validation strategy, allowing each sample to contribute to both training and validation across different folds while reducing the risk of overfitting. Dataset features are described in Table 1.

To examine the relationship between the input features and the target variable (*Course Passed*), a Pearson correlation analysis was performed, and the resulting correlation heatmap is presented in Figure 2. The analysis reveals that *hits_count*, *total_quiz_grade*, and *quiz_count* exhibit the strongest positive correlations with course completion outcomes, with correlation coefficients of 0.54, 0.47, and 0.45, respectively. These findings suggest that learner engagement and assessment-related performance are strongly associated with successful course completion. Features reflecting active interaction with course content demonstrate higher relevance for predictive modeling compared to passive or categorical attributes.

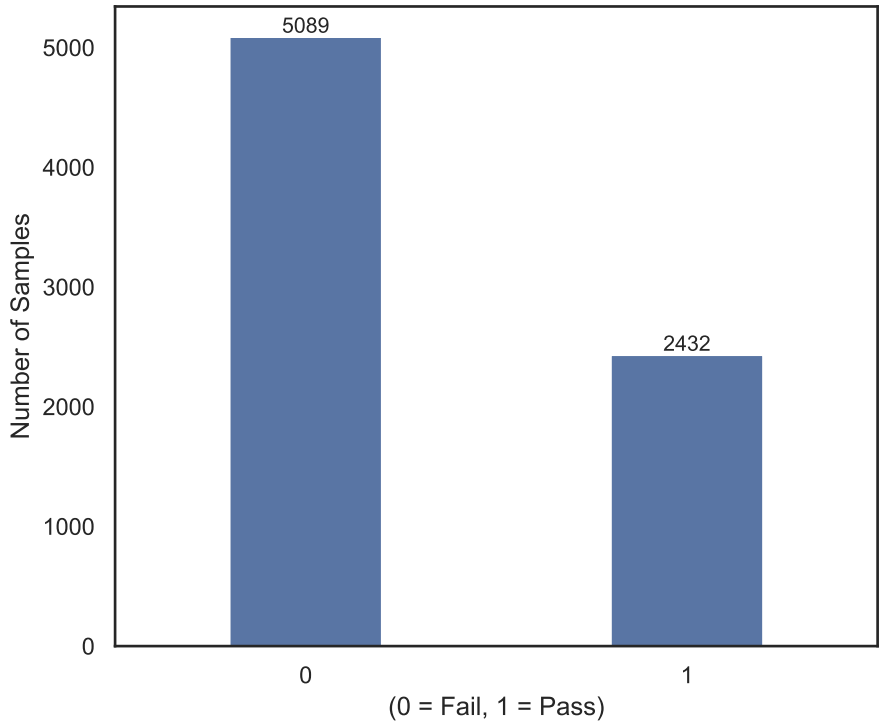


Figure 1. Pass/fail Class-Wise distribution of baseline Coursera dataset.

Table 1. Description of dataset features.

Feature	Description
hits_count	Total number of interactions or clicks performed by a learner
partic_count	Number of participatory actions (e.g., forum or discussion activity)
video_duration	Total duration of video content available or consumed
assessment_type_id_6	Binary indicator for participation in assessment type 6
assessment_type_id_7	Binary indicator for participation in assessment type 7
video_count	Total number of course videos accessed by the learner
quiz_count	Number of quizzes attempted by the learner
total_quiz_grade	Aggregate score obtained across all quizzes
Course Passed	Binary target variable (0 = Fail, 1 = Pass)

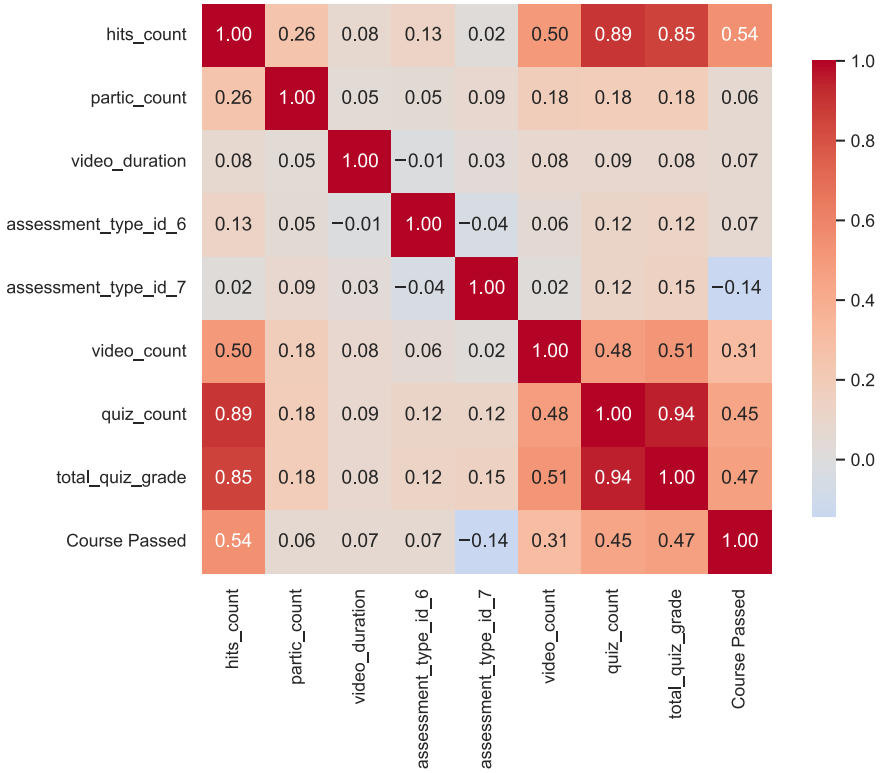


Figure 2. Feature correlation map for the baseline Coursera dataset.

4. Experimental Design and Evaluation Measures

The development of machine learning solutions for predicting students' performance has been divided into three stages. In the first stage, the raw data have been processed by cleaning and selecting relevant features, and missing values were filled. Annotations were added to the data where necessary. The class imbalance problem was addressed by generating simulated data using multiple techniques. In the second stage, machine learning and deep learning models were selected based on the literature, data type, and ease of implementation. All the models were subjected to Grid Search to obtain optimized features for training. Table 2 presents the search space for each model, while Table 3 shows the resulting optimized hyperparameters for each model. The Google Colab platform with Python v3.10 and SciKit packages was used for training.

In the third stage, the trained models were evaluated on unseen test data using various standard evaluation measures such as accuracy, precision, recall, F1 macro, class-wise F1, Sensitivity, Specificity and Matthews Correlation Coefficient (MCC) [30]. A detailed mathematical representation of the evaluation measures is presented under Appendix B.

Table 2. Search space of classification models for GridSearch.

Classifier	Hyperparameter	Search Space
MLP	activation	identity, logistic, tanh, relu
	solver	lbfgs, sgd, adam
	learning_rate_init	0.1, 0.01, 0.001
Decision Tree	splitter	best, random
	criterion	gini, entropy
	min_samples_split	2, 10, 30, 50, 100
	min_samples_leaf	1, 10, 50, 100
kNN	n_neighbors	5, 10, 50, 100
	weights	uniform, distance
	algorithm	auto, ball_tree, kd_tree, brute
	leaf_size	10, 30, 50, 100
Random Forest	n_estimators	10, 30, 50, 100
	criterion	gini, entropy
	min_samples_split	2, 10, 30, 50, 100
	min_samples_leaf	1, 10, 50, 100
XGBoost	booster	gbtree, gblinear, dart
	max_depth	3, 10, 50, 100
	learning_rate	0.1, 0.01, 0.001
	n_estimators	2, 10, 100
CatBoost	iterations	150
	learning_rate	0.1, 0.01, 0.001
	depth	4, 5
	l2_leaf_reg	0.5, 1
SVM	degree	1, 3, 5
	gamma	scale, auto

Table 3. Hyperparameters for training machine learning models.

Model	Hyperparameters
MLP	activation: logistic, learning_rate_init: 0.001, solver: sgd, iter = 500
DT	criterion: entropy, min_samples_leaf: 100, min_samples_split: 2, splitter: random
KNN	algorithm: auto, leaf_size: 10, n_neighbors: 100, weights: uniform
RF	criterion: gini, min_samples_leaf: 100, min_samples_split: 10, n_estimators: 100
XGBoost	booster: gbtree, learning_rate: 0.1, max_depth: 3, n_estimators: 10
CATBoost	depth: 4, iterations: 150, l2_leaf_reg: 1, learning_rate: 0.001
SVC	degree: 1, gamma: scale

The issue of class imbalance was thoroughly investigated in the experimental design through the application of SMOTE (see Appendix A.1 for more details) and GAN (see Appendix A.2 for more details) methods. To address class imbalance in the classification problem, SMOTE techniques were utilized, while GAN models were employed to generate simulated datasets. We note that SMOTE and GAN-based resampling were applied prior to the five-fold cross-validation split, which may introduce minor test-set contamination. While this could slightly inflate absolute performance metrics, the comparative evaluation of resampling techniques and their relative effectiveness remains valid. The Coursera dataset was utilized in these experiments, which comprised multiple investigations, including:

- Experiment A (Balancing using SMOTE): Various SMOTE techniques were employed to balance the Coursera dataset, and machine learning models were implemented to compare their performances against the baseline. The Coursera dataset was balanced

using the SMOTE techniques for this investigation with the same number of input features and the same target variable as in the original dataset.

- Experiment B (Synthetic Data using GANs): Generative models were utilized to augment the Coursera dataset with simulated samples, and the impact of simulated data was evaluated by implementing machine learning models.

5. Results

5.1. Experiment A—Class Balancing Using SMOTE Techniques

In this experiment, multiple SMOTE-based class balancing techniques have been compared for machine learning student performance classification. For each balancing technique, a dataset correlation heatmap is presented to demonstrate the impact on feature correlations in comparison to the original dataset. Furthermore, results are presented in detail in Table 4 for machine learning model performance on each SMOTE balanced dataset. From the tabular results, confusion matrices are presented for the best performing machine learning model for each case. Furthermore, ROC and PR curves are also presented for machine learning models on each SMOTE approach. The results are compared with the Coursera baseline dataset.

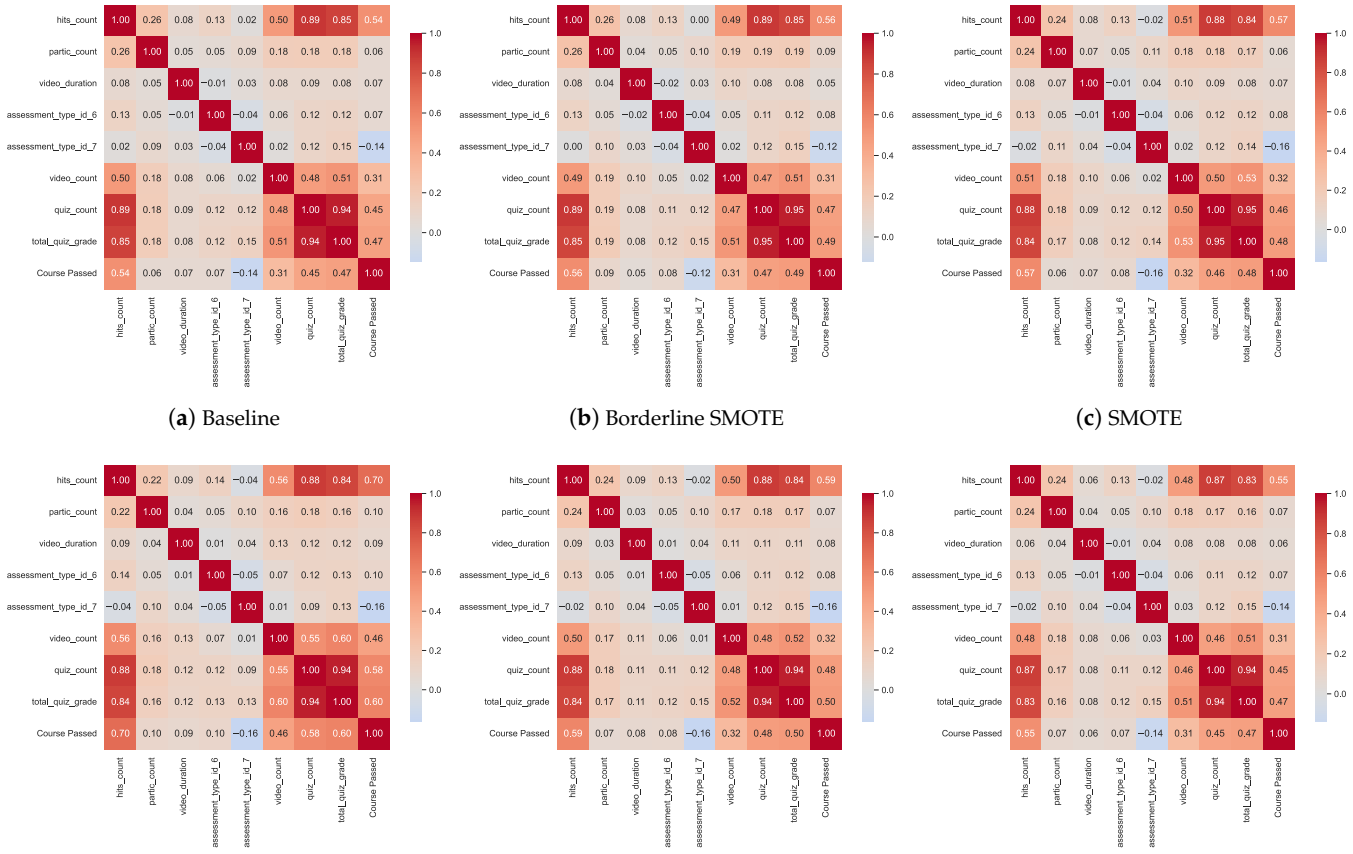
Figure 3 shows the correlation heatmap between the input features and target variable for each SMOTE technique and the baseline. It can be observed that trend remains the same in comparison to the baseline dataset where hit_counts, total_quiz_grade and quiz_count are the top three most correlated features with the target variable. However, it is important to mention that the correlation values increased a bit. In addition, to demonstrate the effectiveness of the applied SMOTE-based oversampling techniques in addressing class imbalance, Figure 4 illustrates the resulting class distributions obtained using each approach. As shown in the figure, all considered techniques successfully produced balanced class distributions between the Pass and Fail categories. However, the total number of samples varies across the different SMOTE variants. This variation arises from the distinct sampling strategies and neighborhood definitions employed by each method. While standard SMOTE generates synthetic samples uniformly until exact class balance is achieved, variants such as SMOTE NN focus on oversampling specific regions of the feature space, particularly near class boundaries or in areas of higher classification difficulty. Consequently, these methods may generate a different number of synthetic instances depending on the local data density and the distribution of minority samples, leading to variations in the final dataset size across techniques.

Table 4 presents the comprehensive quantitative results for the implemented machine learning models to each of the SMOTE-sampled datasets and compared with the baseline. From the tabular results, it can be observed that apart from SMOTE NN, all other SMOTE approaches resulted in similar performance to the baseline. The CATBoost model performed the best among all models, while SVC was among the lowest performers. In terms of specific performance, CATBoost with SMOTE NN resulted in the best performance, with an accuracy of 89% and MCC of 0.60.

Figure 5 shows the confusion matrices for the best performing model to evaluate the performance in terms of Type I and Type II errors. From the results, the best performance was observed for the RF with the SMOTE NN case where a Type I error of 15% and Type II error of 8% was observed. Figures 6 and 7 show the ROC and PR curves for each of the SMOTE techniques compared with the baseline. CATBoost along with XGBoost can be observed as the best performing model among the most cases, except the SMOTE NN, where the RF model emerged as the best. The AUC for the ROC and PR curves was best observed for RF on the SMOTE NN as 0.96.

Table 4. Quantitative test results for machine learning classification of student performance using Coursera dataset balanced using multiple SMOTE techniques.

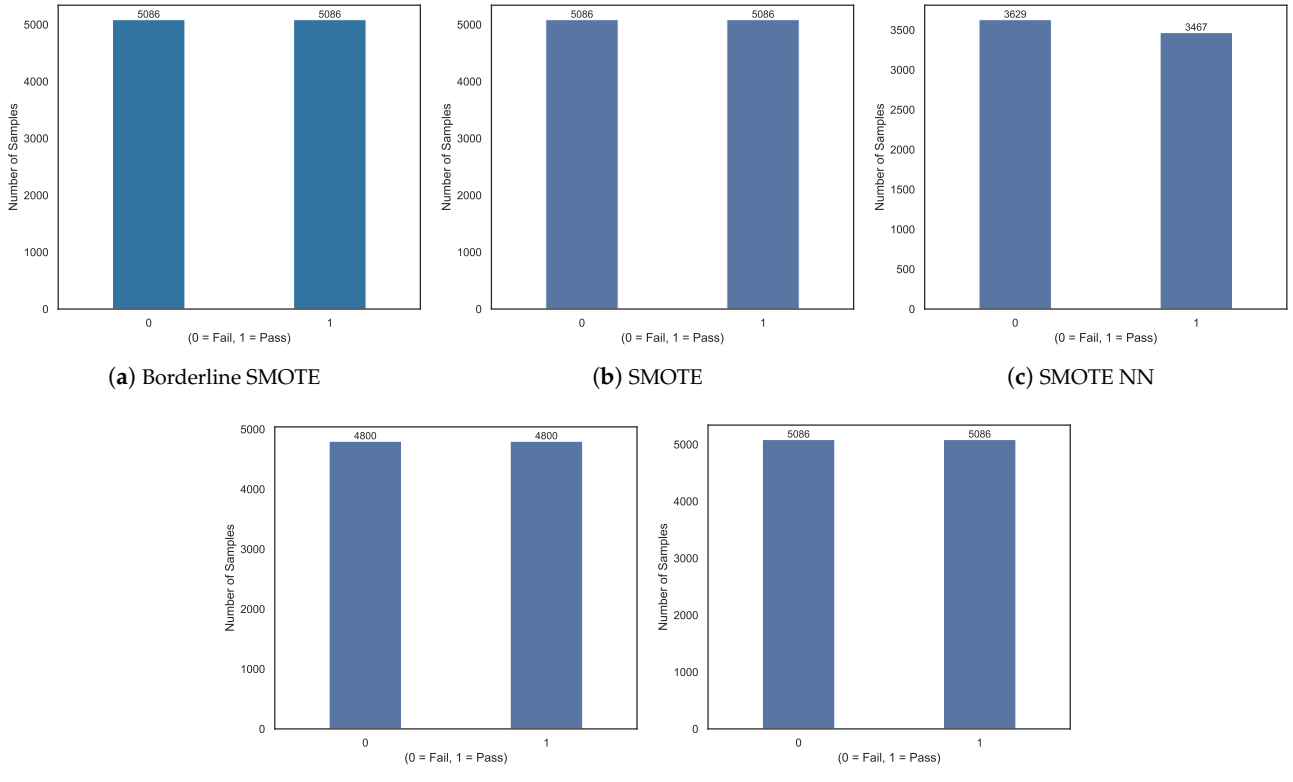
Models	Accuracy	F1 Macro	F1 Positive	F1 Negative	Precision	Recall	Sensitivity	Specificity	MCC
Baseline									
MLP	0.81	0.76	0.66	0.86	0.80	0.77	0.87	0.66	0.55
DT	0.77	0.72	0.62	0.83	0.74	0.72	0.87	0.50	0.47
KNN	0.79	0.74	0.63	0.85	0.77	0.74	0.87	0.62	0.51
RF	0.78	0.73	0.62	0.85	0.79	0.74	0.88	0.61	0.51
XGBoost	0.80	0.74	0.61	0.86	0.79	0.74	0.90	0.57	0.52
CATBoost	0.79	0.74	0.64	0.85	0.79	0.75	0.86	0.64	0.53
SVC	0.71	0.65	0.54	0.76	0.74	0.69	0.75	0.62	0.40
Borderline SMOTE									
MLP	0.78	0.78	0.78	0.78	0.78	0.78	0.80	0.76	0.56
DT	0.79	0.78	0.79	0.78	0.80	0.79	0.70	0.81	0.57
KNN	0.76	0.75	0.77	0.74	0.77	0.76	0.68	0.83	0.52
RF	0.78	0.78	0.80	0.78	0.79	0.78	0.73	0.84	0.58
XGBoost	0.79	0.79	0.80	0.78	0.80	0.79	0.73	0.85	0.59
CATBoost	0.80	0.79	0.80	0.78	0.80	0.80	0.74	0.85	0.60
SVC	0.74	0.74	0.75	0.72	0.75	0.74	0.67	0.81	0.49
SMOTE									
MLP	0.79	0.79	0.79	0.80	0.79	0.79	0.81	0.77	0.58
DT	0.76	0.76	0.78	0.76	0.77	0.76	0.68	0.82	0.55
KNN	0.77	0.77	0.79	0.75	0.78	0.77	0.69	0.85	0.55
RF	0.79	0.78	0.80	0.79	0.79	0.79	0.73	0.86	0.60
XGBoost	0.80	0.80	0.79	0.81	0.80	0.80	0.85	0.75	0.60
CATBoost	0.80	0.80	0.81	0.79	0.80	0.80	0.75	0.85	0.60
SVC	0.74	0.74	0.75	0.72	0.75	0.74	0.89	0.80	0.49
SMOTE NN									
MLP	0.88	0.87	0.88	0.86	0.88	0.88	0.83	0.91	0.75
DT	0.88	0.88	0.88	0.88	0.88	0.88	0.86	0.89	0.76
KNN	0.88	0.88	0.87	0.88	0.88	0.88	0.88	0.87	0.76
RF	0.89	0.89	0.89	0.89	0.89	0.89	0.85	0.92	0.78
XGBoost	0.88	0.88	0.88	0.89	0.88	0.88	0.89	0.87	0.78
CATBoost	0.89	0.89	0.89	0.90	0.89	0.89	0.91	0.87	0.78
SVC	0.86	0.86	0.86	0.86	0.87	0.86	0.85	0.87	0.73
SMOTE Tomek									
MLP	0.80	0.80	0.80	0.80	0.80	0.80	0.82	0.79	0.61
DT	0.79	0.79	0.81	0.78	0.81	0.80	0.72	0.87	0.60
KNN	0.79	0.79	0.80	0.77	0.80	0.79	0.73	0.85	0.58
RF	0.81	0.81	0.81	0.80	0.82	0.81	0.76	0.85	0.62
XGBoost	0.81	0.81	0.82	0.80	0.82	0.81	0.76	0.86	0.63
CATBoost	0.82	0.82	0.82	0.81	0.82	0.82	0.78	0.86	0.64
SVC	0.76	0.76	0.77	0.74	0.77	0.76	0.70	0.81	0.53
SVM SMOTE									
MLP	0.79	0.79	0.77	0.79	0.79	0.79	0.81	0.76	0.57
DT	0.78	0.77	0.79	0.76	0.78	0.78	0.70	0.85	0.56
KNN	0.77	0.77	0.78	0.76	0.78	0.77	0.72	0.83	0.55
RF	0.78	0.78	0.81	0.79	0.79	0.78	0.76	0.83	0.60
XGBoost	0.79	0.79	0.80	0.79	0.80	0.79	0.76	0.83	0.59
CATBoost	0.79	0.79	0.80	0.79	0.80	0.79	0.75	0.84	0.60
SVC	0.75	0.74	0.76	0.73	0.76	0.75	0.69	0.81	0.50



(**a**) Baseline (**b**) Borderline SMOTE (**c**) SMOTE

(**d**) SMOTE NN (**e**) SMOTE Tomek (**f**) SVM SMOTE

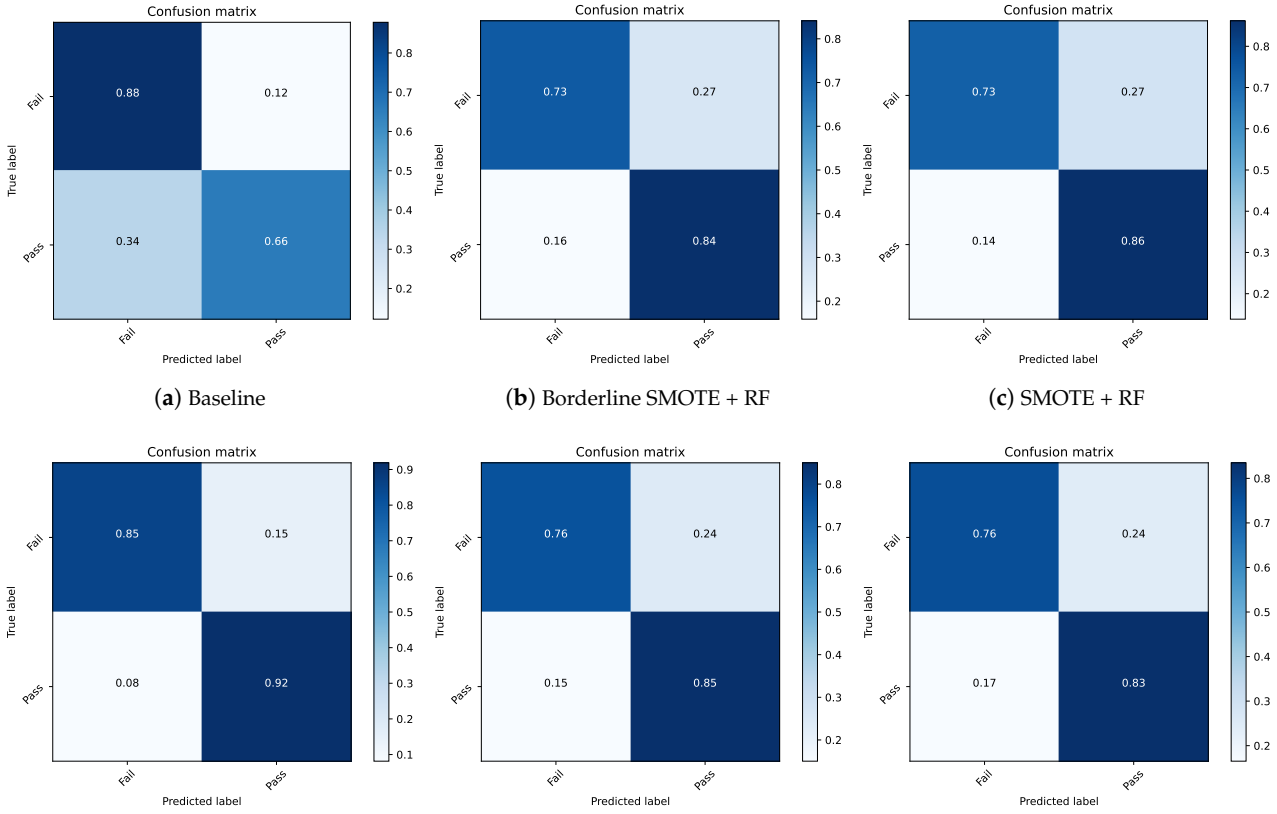
Figure 3. Correlation heatmaps for different SMOTE techniques used for dataset balancing.



(**a**) Borderline SMOTE (**b**) SMOTE (**c**) SMOTE NN

(**d**) SMOTE Tomek (**e**) SVM SMOTE

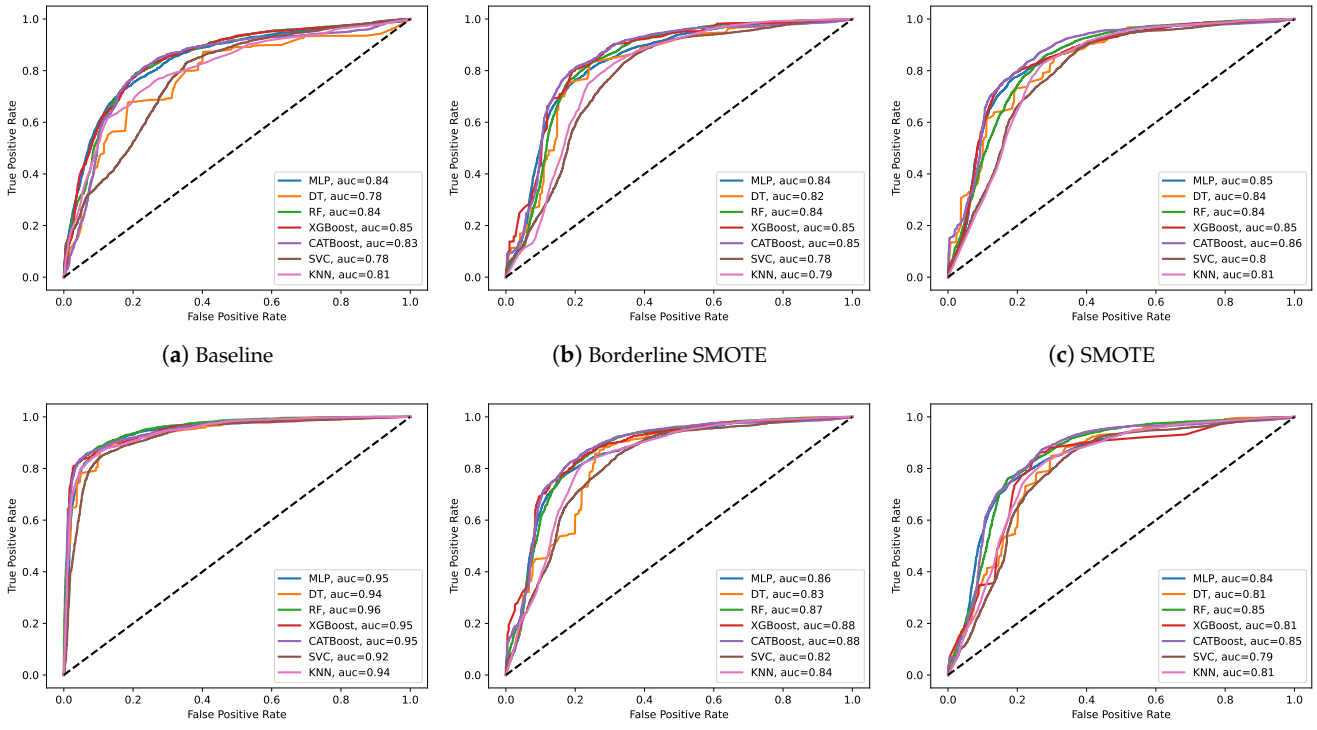
Figure 4. Data Class-Wise distribution for multiple SMOTE techniques.



(**a**) Baseline (**b**) Borderline SMOTE + RF (**c**) SMOTE + RF

(**d**) SMOTE NN + RF (**e**) SMOTE Tomek + RF (**f**) SVM SMOTE + RF

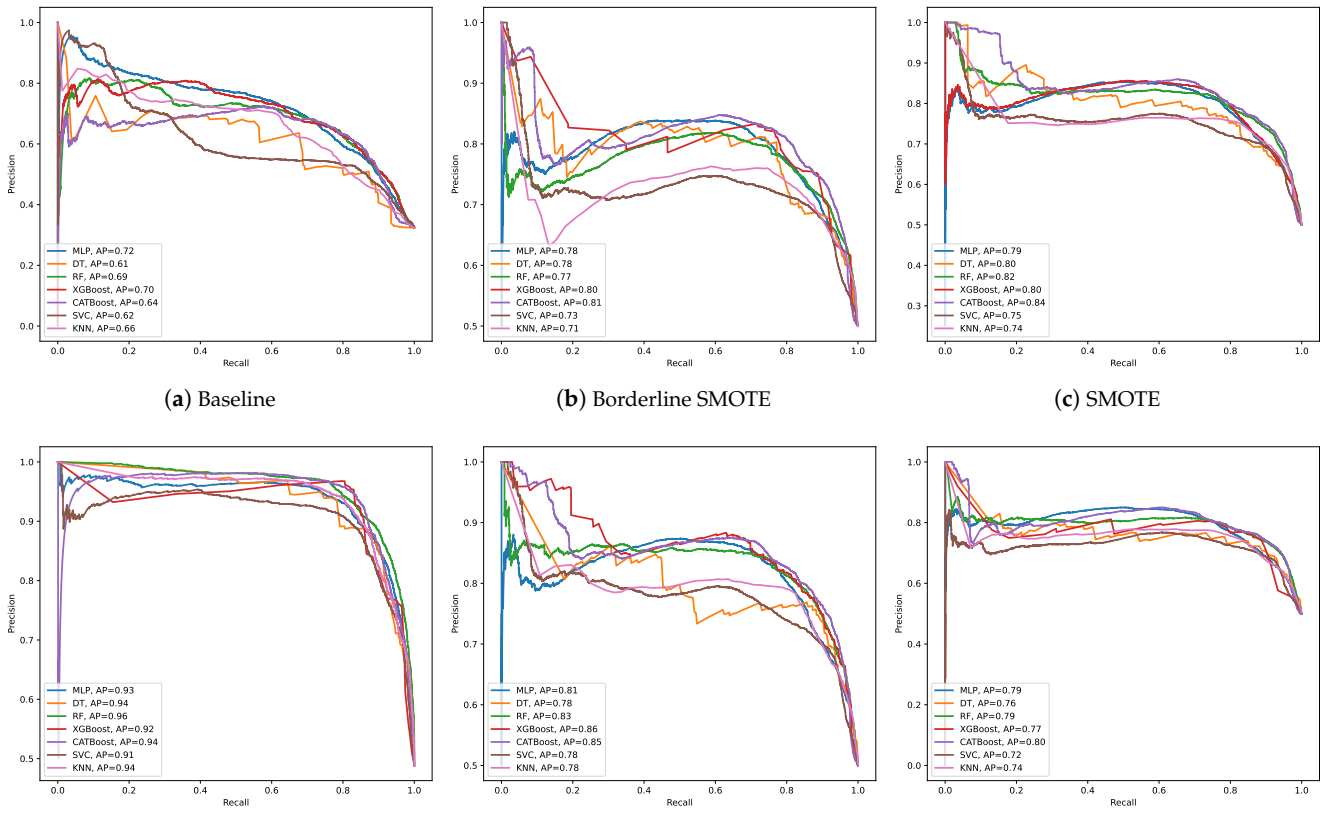
Figure 5. Confusion matrices of best performing models for each SMOTE technique.



(**a**) Baseline (**b**) Borderline SMOTE (**c**) SMOTE

(**d**) SMOTE NN (**e**) SMOTE Tomek (**f**) SVM SMOTE

Figure 6. ROC curves for machine learning models trained using the dataset balanced using different SMOTE techniques.



(**a**) Baseline (**b**) Borderline SMOTE (**c**) SMOTE

(**d**) SMOTE NN (**e**) SMOTE Tomek (**f**) SVM SMOTE

Figure 7. Precision–Recall (PR) curves for machine learning models trained using the dataset balanced using different SMOTE techniques.

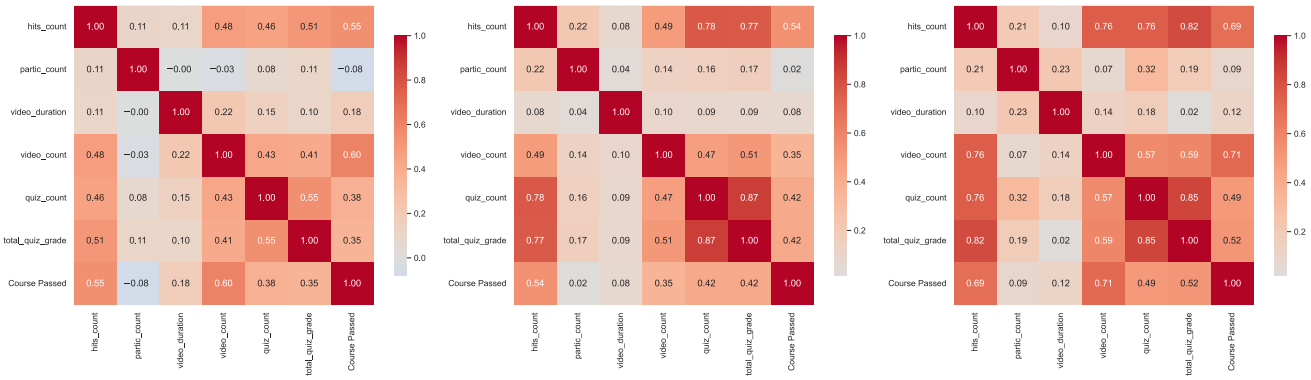
5.2. Experiment B—Synthetic Data Using GANs

Experiment B reported in this paper investigates the effectiveness of GANs for data augmentation in imbalanced educational datasets. Three distinct datasets were constructed and evaluated. The first dataset, referred to as the *GAN-simulated Coursera dataset*, consists solely of synthetic samples generated by a GAN trained on the original Coursera data. This dataset was used to assess whether machine learning models trained exclusively on synthetic data can capture the underlying patterns of the real dataset. The second dataset, termed the *Coursera Original dataset with GAN-simulated data*, was constructed by combining the original Coursera data with additional GAN-generated synthetic samples. The objective of this dataset is to examine whether augmenting real data with high-fidelity synthetic samples can improve predictive performance while preserving the original data distribution. The third dataset, denoted as the *SMOTENN balanced GAN-simulated data*, was created by balancing the GAN-simulated dataset using the SMOTENN technique.

The hyperparameters used for the experiments were the same as reported in Table 3. Figure 8 shows the correlation maps for the simulated datasets, which demonstrate that the artificially generated dataset retained the correlation of the baseline dataset with similar features as the top and worst related with the target variable. Table 5 presents the quantitative results for each of the sub-experiments investigating the impact of synthetic data, while Figures 9–11 present the confusion matrices, ROC and PR curves for the machine learning models.

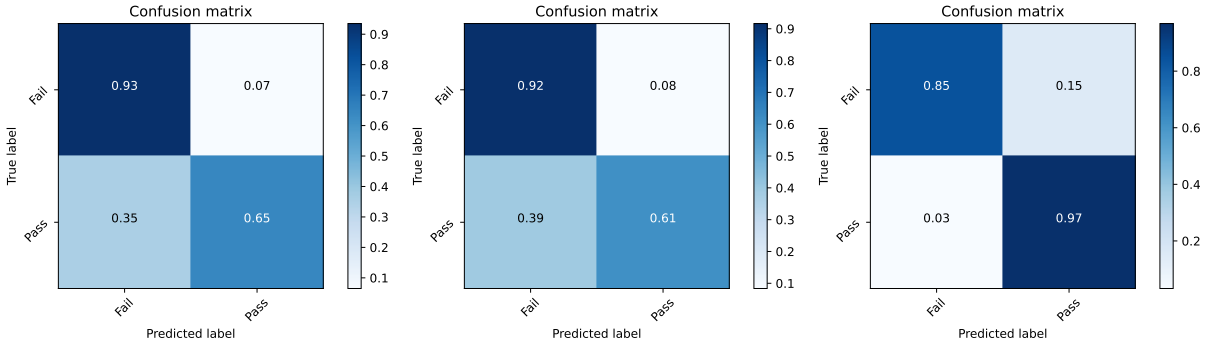
Table 5. Quantitative test results for machine learning classification of student performance using different combinations of synthetic data.

Models	Accuracy	F1 Macro	F1 Positive	F1 Negative	Precision	Recall	Sensitivity	Specificity	MCC
GAN-Simulated Coursera Dataset									
MLP	0.87	0.80	0.68	0.91	0.84	0.79	0.93	0.65	0.62
DT	0.82	0.74	0.63	0.90	0.77	0.74	0.89	0.59	0.55
KNN	0.85	0.77	0.64	0.90	0.83	0.77	0.92	0.62	0.58
RF	0.84	0.76	0.62	0.90	0.83	0.76	0.92	0.60	0.56
XGBoost	0.86	0.78	0.68	0.92	0.83	0.79	0.93	0.65	0.62
CATBoost	0.86	0.77	0.64	0.91	0.84	0.76	0.94	0.59	0.59
SVC	0.86	0.78	0.65	0.91	0.84	0.79	0.93	0.66	0.61
Original + GAN-Simulated Coursera Dataset									
MLP	0.80	0.72	0.59	0.88	0.79	0.71	0.93	0.51	0.52
DT	0.82	0.75	0.62	0.88	0.78	0.74	0.90	0.58	0.52
KNN	0.80	0.71	0.56	0.87	0.77	0.71	0.91	0.51	0.47
RF	0.82	0.74	0.60	0.88	0.79	0.74	0.91	0.55	0.52
XGBoost	0.83	0.77	0.56	0.88	0.82	0.76	0.92	0.60	0.49
CATBoost	0.81	0.74	0.6	0.88	0.78	0.73	0.92	0.55	0.51
SVC	0.80	0.71	0.55	0.87	0.77	0.71	0.92	0.50	0.46
SMOTENN-GAN Coursera Dataset									
MLP	0.92	0.92	0.91	0.89	0.92	0.92	0.89	0.95	0.82
DT	0.88	0.88	0.90	0.87	0.89	0.88	0.83	0.92	0.78
KNN	0.92	0.92	0.93	0.91	0.93	0.92	0.87	0.97	0.85
RF	0.92	0.91	0.92	0.90	0.93	0.91	0.86	0.96	0.83
XGBoost	0.91	0.91	0.90	0.87	0.92	0.91	0.85	0.97	0.80
CATBoost	0.91	0.90	0.91	0.90	0.91	0.91	0.90	0.91	0.81
SVC	0.90	0.90	0.91	0.89	0.91	0.90	0.88	0.93	0.81



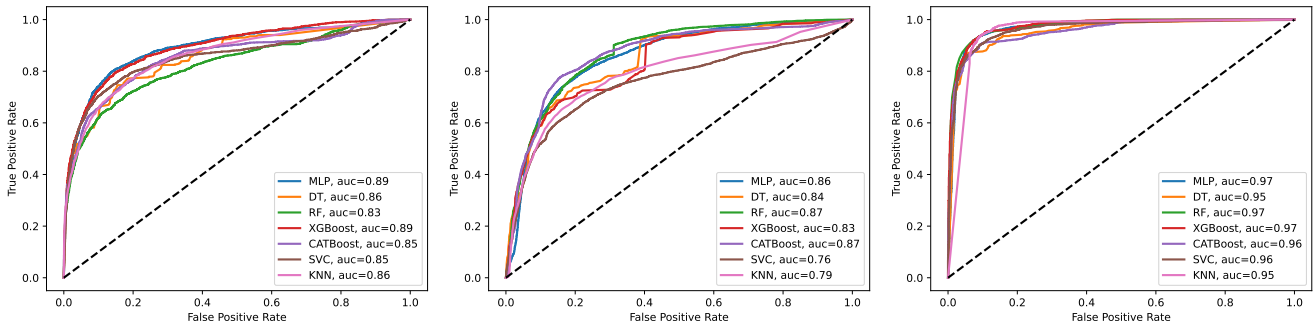
(**a**) GAN-Simulated (**b**) Original + GAN (**c**) SMOTENN-GAN

Figure 8. Correlation heatmaps for synthetic datasets generated using SMOTE and GAN techniques.



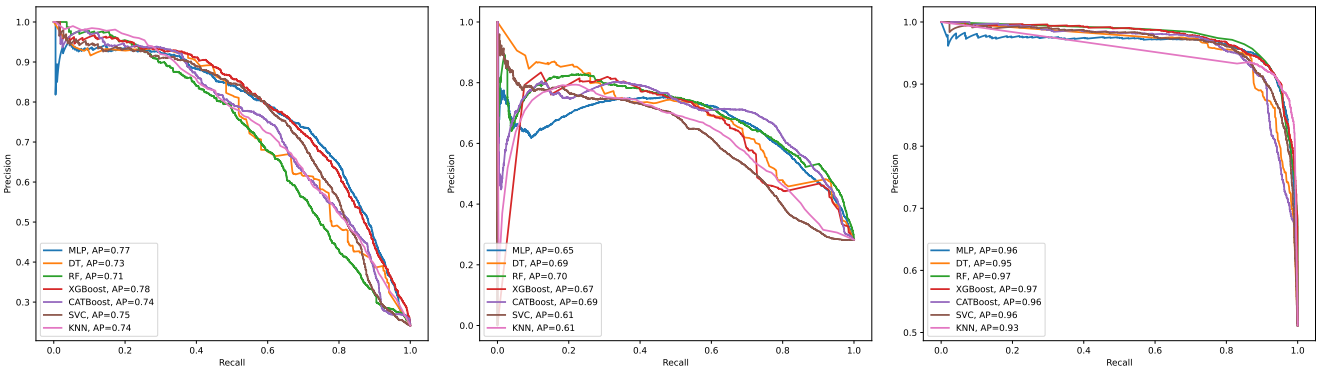
(**a**) GAN-Simulated (XGBoost) (**b**) Original + GAN-Simulated (XGBoost) (**c**) SMOTENN-GAN (XGBoost)

Figure 9. Confusion matrices of implemented machine learning models for classification of students' performance using different combinations of synthetic dataset.



(**a**) GAN-Simulated (**b**) Original + GAN-Simulated (**c**) SMOTENN-GAN

Figure 10. ROC curves of implemented machine learning models for classification of students' performance using different combinations of synthetic dataset.



(**a**) GAN-Simulated (**b**) Original + GAN-Simulated (**c**) SMOTENN-GAN

Figure 11. Precision–Recall (PR) curves of implemented machine learning models for classification of students' performance using different combinations of synthetic dataset.

For the GAN-simulated unbalanced Coursera dataset, it was observed that the MLP model achieved the best accuracy of 0.87, while DT had the lowest accuracy of 0.82. The simulated data was able to improve the overall accuracy of the models. In terms of Type I and Type II errors, the SVC model achieved the best Type II error of 34%, while CATBoost achieved the best Type I error of 6%. The XGBoost model had the most distributed performance among classes, as demonstrated by the ROC and PR curves with an AUC of 0.89 and 0.78, respectively.

For the Coursera Original dataset with GAN-simulated unbalanced data, it was observed that adding the simulated data in unbalanced form did not significantly improve the performance of the models. The XGBoost model achieved the best accuracy of 0.83, while SVC and KNN had the lowest accuracy of 0.80. The XGBoost model also achieved the best Type II error of 39%, while MLP achieved the best Type I error of 7%. However, the ROC and PR curves showed that the RF model had the highest AUC of 0.87 and 0.70, respectively. Adding the simulated data to the original data in unbalanced form did not help much in improving the performance. This suggested that class balancing is of more importance in comparison to increased data size for this specific case.

Finally, for the balanced Coursera dataset generated using SMOTE NN with GAN-simulated data, it was observed that the XGBoost model achieved the best accuracy of 0.85, while SVC had the lowest accuracy of 0.76. The XGBoost model also achieved the best Type II error of 3%, while MLP achieved the best Type I error of 10%. The ROC and PR curves showed that the XGBoost model had the highest AUC of 0.97. A significant improvement

in results was improved for the case when the original data was first balanced using the SMOTE NN technique prior to generating GAN-simulated data.

Overall, it can be concluded that the GAN-simulated data was able to improve the performance of the machine learning models in some cases, particularly when used with a SMOTE NN balanced dataset. The XGBoost model was the best performing model across all three sub-experiments.

6. Discussions, Limitations and Future Opportunities

The results of this study highlight the importance of class imbalance handling and synthetic data integration for student performance prediction in VLEs. Among the evaluated techniques, SMOTE NN consistently yielded the strongest improvements across accuracy, MCC, and ROC/PR metrics, indicating that neighborhood-based oversampling and data cleaning are particularly effective for educational datasets characterized by overlapping class distributions and noisy behavioral signals. Ensemble learning methods, especially XGBoost, CATBoost, and Random Forest, demonstrated robust and stable performance across both imbalance-handling and synthetic data scenarios, reflecting their ability to capture nonlinear interactions among VLE-derived features. The GAN-based experiments further revealed that synthetic data generation alone provides limited benefits when class imbalance persists; however, when applied after SMOTE NN balancing, GAN-augmented datasets produced substantial performance gains, including reduced misclassification errors and near-optimal ROC-AUC values. These findings collectively suggest that data-centric preprocessing strategies are more influential than model complexity alone for this task, and that synthetic data generation is most effective when embedded within a carefully designed balancing framework.

Nevertheless, the generalizability of the findings is constrained by several factors. All experiments were conducted using a single Coursera dataset derived exclusively from computer science courses, which may limit the applicability of the conclusions to other academic disciplines, learner populations, or educational platforms. Learning behaviors, assessment structures, and engagement dynamics can vary significantly across domains such as humanities, social sciences, or professional training, as well as across different learning management systems and institutional contexts. Consequently, the observed effectiveness of SMOTE NN, GAN-based augmentation, and ensemble models may not directly transfer to these settings without further validation. In addition, the feature space was restricted to VLE interaction and assessment-related variables, excluding demographic, socio-economic, and cognitive factors that may play a critical role in student performance and risk identification. The use of GAN-generated data also introduces potential risks related to hidden distributional bias, despite preserved correlation structures, and the strongest-performing models remain largely black-box, limiting interpretability for educational stakeholders.

Future research should therefore prioritize cross-discipline and cross-platform validation by applying the proposed framework to datasets from multiple academic domains and diverse learning environments to assess robustness and transferability. Expanding the feature set to include contextual and longitudinal learner information may further enhance both predictive performance and interpretability. Additionally, adaptive imbalance-handling pipelines that dynamically select oversampling and synthetic data strategies based on dataset characteristics warrant investigation. Emphasis should also be placed on integrating explainable AI techniques to improve transparency and trust, as well as on evaluating the real-world impact of these predictive models within early-warning and intervention systems. Finally, ethical considerations related to fairness, bias mitigation, and responsible deployment of synthetic data should be systematically addressed to ensure the equitable and trustworthy use of predictive analytics in education.

7. EduPredictor: Revolutionizing VLE Through Predictive Analytics

The rise of VLEs has created a critical need for technology-driven tools to enhance learning and predictive analytics in education. Building on the predictive analysis in this manuscript, we propose the idea of EduPredictor, an application that utilizes trained machine learning models on educational datasets for practical use. EduPredictor is proposed to be a state-of-the-art student performance monitoring and intervention system designed to address education-related challenges.

EduPredictor will use the prediction power of a machine learning model (e.g., XGBoost) trained on comprehensive VLE datasets. This model will ensure high accuracy in predicting student outcomes based on a myriad of factors, including engagement metrics, assignment scores, and interactive session data. The system will provide educators with an intuitive dashboard, offering a comprehensive view of student performance, using color-coded indicators and predictive analytics to quickly identify at-risk students. In addition, the tool will suggest tailored interventions, ranging from additional study materials to one-on-one tutoring sessions, ensuring that interventions are both timely and effective. EduPredictor will be designed for seamless integration with popular LMS, ensuring educators can access its features without disrupting workflows. EduPredictor will prioritize data security with robust encryption and adherence to international privacy standards, ensuring educators can use the tool confidently without compromising student privacy. Early identification of at-risk students will enable institutions to implement targeted support programs, helping reverse academic decline and boost student confidence. EduPredictor will optimize resource allocation, ensuring interventions are focused where most needed, thereby maximizing institutional impact. Beyond interventions, EduPredictor's data insights will inform curriculum development, teaching methods, and instructional design, promoting ongoing improvements in education.

8. Conclusions

In conclusion, this study addressed class imbalance in predicting student performance using machine learning. We applied SMOTE and GAN techniques to balance the dataset and generate new samples, comparing seven machine learning models. Among the SMOTE resampling methods, SMOTE NN yielded the strongest performance for the RF model, achieving a ROC AUC of 0.96 and a Type II error rate of 8%. In the generative data experiments, XGBoost delivered the best results when trained on the GAN-generated dataset balanced with SMOTE NN, attaining a ROC AUC of 0.97 and reducing the Type II error rate to just 3%. This approach has significant implications for educational institutions, especially in remote learning contexts like the COVID-19 pandemic. Predicting student performance can help identify at-risk students and provide timely interventions. Our proposed idea of application, EduPredictor, recommends these models to monitor and intervene in student performance. However, limitations remain, including reliance on a single Coursera dataset and the exclusion of contextual variables such as socio-economic status or learning disabilities. Future research should validate these approaches across diverse disciplines and platforms, expand the feature set, and incorporate explainable AI techniques to improve transparency, fairness, and actionable insights for educational stakeholders.

Author Contributions: Conceptualization, F.M.A., T.B., E.H. and M.Y.-k.; methodology, F.M.A., T.B., E.H. and M.Y.-k.; software, F.M.A., T.B., E.H. and M.Y.-k.; validation, F.M.A., T.B., E.H. and M.Y.-k.; formal analysis, F.M.A.; investigation, F.M.A.; resources, T.B., E.H. and M.Y.-k.; data curation, F.M.A.; writing—original draft preparation, F.M.A.; writing—review and editing, F.M.A., T.B., E.H. and M.Y.-k.; visualization, F.M.A., T.B., E.H. and M.Y.-k.; supervision, T.B., E.H. and M.Y.-k. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Acknowledgments: The work presented in this manuscript is derived from the PhD thesis of Fatema Mohammad Alnassar titled "Predicting Student Performance on Virtual Learning Environment" submitted to Goldsmiths University of London, London, in 2023.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Background to Sampling Techniques and Generative Models

Appendix A.1. Synthetic Minority Over-Sampling Technique (SMOTE)

The SMOTE is a widely used data-level method for addressing class imbalance in supervised learning problems. Class imbalance arises when one or more classes are underrepresented relative to others, which can bias learning algorithms toward the majority class and degrade predictive performance on minority outcomes. This issue is prevalent in domains such as healthcare, finance, and security, where minority classes often correspond to rare but critical events. Unlike naive oversampling, which duplicates existing minority samples and increases the risk of overfitting, SMOTE generates new synthetic observations to enrich the minority class distribution.

The fundamental principle of SMOTE is to create synthetic minority samples through interpolation in the feature space. For each minority-class instance, the algorithm identifies its k nearest minority neighbors and generates new samples along the line segments connecting the instance to one of its neighbors. Formally, a synthetic sample is generated by adding a random proportion of the difference between the feature vector of the original instance and that of a selected neighbor. This approach expands the minority class decision region and encourages classifiers to learn smoother and less biased decision boundaries. The primary advantage of SMOTE lies in its simplicity and effectiveness in reducing class imbalance without merely replicating existing data. However, SMOTE does not consider the local class distribution and may generate ambiguous or noisy samples in regions where minority and majority classes overlap.

To address this limitation, several extensions of SMOTE have been proposed. Borderline-SMOTE focuses on minority samples that lie near the decision boundary between classes. It identifies minority instances whose nearest neighbors include a significant proportion of majority-class samples and preferentially generates synthetic data around these borderline points. The underlying principle is that such regions are more informative for classification, as they represent areas of higher misclassification risk. Borderline-SMOTE often improves classifier sensitivity on difficult cases, but it may also amplify noise if borderline samples are themselves unreliable or mislabeled.

Another extension, Adaptive Synthetic Sampling (ADASYN), further refines the generation process by adapting the number of synthetic samples produced for each minority instance based on local learning difficulty. Minority samples surrounded by many majority-class neighbors receive higher weights and generate more synthetic data, while well-separated samples generate fewer or none. This adaptive mechanism shifts the classifier's focus toward harder-to-learn regions of the feature space. The advantage of ADASYN is its ability to concentrate synthetic data generation where it is most needed; however, this same property can increase sensitivity to noise and outliers, potentially leading to unstable decision boundaries if the data quality is poor [31,32].

SMOTE and its variants provide effective tools for mitigating class imbalance by augmenting minority-class data in a principled manner. Standard SMOTE offers simplicity and broad applicability, Borderline-SMOTE enhances learning near class boundaries, and ADASYN adaptively targets difficult regions. Nevertheless, all variants require careful parameter tuning and should be applied with consideration of class overlap, noise, and the underlying data distribution.

Appendix A.2. Generative Adversarial Networks (GANs)

GANs are a class of machine learning models that were first introduced by Ian Goodfellow and his colleagues in 2014. GANs are unique in that they use a two-part system, consisting of a generator and a discriminator, to generate new data that are similar to a training dataset. The generator creates new candidate samples, while the discriminator evaluates their authenticity by comparing them to the original training data. The generator then attempts to create better samples by learning from its mistakes, and the discriminator aims to better differentiate between the real and fake samples [33].

The idea behind GANs is that the generator network learns to create new data by attempting to trick the discriminator network into thinking that the new samples are real data. This means that the generator is not explicitly told what kind of data to produce, but instead learns to create data that are similar to the training data by constantly improving its ability to deceive the discriminator. The discriminator is trained initially on a well-known dataset to accurately identify real data. The generator network is then seeded with random input, typically sampled from a latent space, and generates candidate samples. The discriminator evaluates the candidate samples and provides feedback to the generator on how to improve its samples. Both networks undergo independent backpropagation techniques, resulting in the generator producing better samples and the discriminator improving its ability to identify fake samples [34].

GANs have been used for various applications, such as generating realistic images, music, and text. They are capable of unsupervised learning, semi-supervised learning, fully supervised learning, and even reinforcement learning. They have also been used in transfer learning scenarios where pre-trained GANs are fine-tuned for a specific task. In image production, GANs use convolutional neural networks (CNNs) as discriminators and deconvolutional neural networks (DCNNs) as generators. The discriminator network typically consists of several convolutional layers followed by fully connected layers, while the generator network consists of several deconvolutional layers followed by fully connected layers.

GAN Implementation Details

We generated synthetic tabular data using TabGAN, a generative adversarial framework specifically designed for structured datasets. Unlike image-based GANs, TabGAN abstracts the internal neural generator architecture and instead emphasizes post-generation adversarial filtering to improve stability and fidelity. The generator produces candidate synthetic samples from the joint distribution of training features and targets, while a separate adversarial model based on gradient-boosted decision trees is trained to distinguish real from synthetic observations. This discriminator uses mean squared error as its optimization metric and is configured with a maximum tree depth of 4200 estimators, a learning rate of 0.05, subsampling and feature subsampling rates of 0.8, and a fixed random seed for reproducibility. Synthetic samples that are easily distinguishable from real data are removed during post-processing, improving realism and reducing overfitting.

The synthetic-to-real data ratio was controlled by setting the generation multiplier to five, with a pre-generation fraction of two to encourage diversity without overwhelming the

empirical distribution. Additional stability safeguards included post-generation quantile filtering, where values below the 1st percentile and above the 99th percentile were removed to eliminate extreme outliers and reduce the risk of degenerate samples. Batch size, number of epochs, and internal neural optimization details are managed internally by the TabGAN library and are not directly user-configurable.

Synthetic data fidelity and utility were assessed using both distributional and task-based validation. Marginal feature distributions, summary statistics, and pairwise correlations of synthetic data were compared against those of the real training data and showed close alignment without exact replication. To mitigate privacy risks, no direct identifiers were included in the training data, and no row-level conditioning or record seeding was applied. We additionally verified that no synthetic samples were exact duplicates of real observations. While formal differential privacy guarantees are not claimed, these safeguards substantially reduce membership inference and reconstruction risks in line with prior work on tabular GANs.

Appendix B. Evaluation Metrics

Let TP, TN, FP, and FN denote the number of true positives, true negatives, false positives, and false negatives, respectively.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \text{Sensitivity} = \frac{TP}{TP + FN}$$

$$\text{Specificity} = \frac{TN}{TN + FP}$$

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

$$\text{F1}_{\text{macro}} = \frac{1}{C} \sum_{c=1}^{C} \text{F1}_c$$

$$\text{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

References

1. Szymkowiak, A.; Melović, B.; Dabić, M.; Jeganathan, K.; Kundi, G.S. Information technology and Gen Z: The role of teachers, the internet, and technology in the education of young people. *Technol. Soc.* **2021**, *65*, 101565. [CrossRef]
2. Gavilanes-Sagnay, F.; Loza-Aguirre, E.; Riofrío-Luzcando, D.; Segura-Morales, M. A systematic literature review of indicators for the understanding of interactions in Virtual Learning Environments. In Proceedings of the 2018 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, 12–14 December 2018; pp. 596–600.
3. Torres Martín, C.; Acal, C.; El Honrani, M.; Mingorance Estrada, Á.C. Impact on the virtual learning environment due to COVID-19. *Sustainability* **2021**, *13*, 582. [CrossRef]

4. Darusman, A.H.; Omar, Y. Enhancing Student Engagement in VLE Platform: Student Perceptions Towards Programming Course Learning Resources. *Psychol. Educ.* **2021**, *58*, 5607–5612.
5. Shahiri, A.M.; Husain, W.; Rashid, N.A. A review on predicting student's performance using data mining techniques. *Procedia Comput. Sci.* **2015**, *72*, 414–422. [CrossRef]
6. Hellas, A.; Ihantola, P.; Petersen, A.; Ajanovski, V.V.; Gutica, M.; Hynninen, T.; Knutas, A.; Leinonen, J.; Messom, C.; Liao, S.N. Predicting academic performance: A systematic literature review. In Proceedings of the Proceedings Companion of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education, Larnaca, Cyprus, 2–4 July 2018; pp. 175–199.
7. Sekeroglu, B.; Dimililer, K.; Tuncal, K. Student performance prediction and classification using machine learning algorithms. In Proceedings of the 2019 8th International Conference on Educational and Information Technology, Cambridge, UK, 2–4 March 2019; pp. 7–11.
8. Elbadrawy, A.; Studham, R.S.; Karypis, G. Collaborative multi-regression models for predicting students' performance in course activities. In Proceedings of the Fifth International Conference on Learning Analytics and Knowledge, Poughkeepsie, NY, USA, 16–20 March 2015; pp. 103–107.
9. Yee-King, M.; Grimalt-Reynes, A.; d'Inverno, M. Predicting student grades from online, collaborative social learning metrics using K-NN. In Proceedings of the EDM, Raleigh, NC, USA, 29 June–2 July 2016; pp. 654–655.
10. Al-Shehri, H.; Al-Qarni, A.; Al-Saati, L.; Batoaq, A.; Badukhen, H.; Alrashed, S.; Alhiyafi, J.; Olatunji, S.O. Student performance prediction using support vector machine and k-nearest neighbor. In Proceedings of the 2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE), Windsor, ON, Canada, 30 April–3 May 2017; pp. 1–4.
11. Iqbal, Z.; Qadir, J.; Mian, A.N.; Kamiran, F. Machine learning based student grade prediction: A case study. *arXiv* **2017**, arXiv:1708.08744. [CrossRef]
12. Hussain, M.; Zhu, W.; Zhang, W.; Abidi, S.M.R. Student engagement predictions in an e-learning system and their impact on student course assessment scores. *Comput. Intell. Neurosci.* **2018**, *2018*, 6347186. [CrossRef] [PubMed]
13. Heuer, H.; Breiter, A. Student success prediction and the trade-off between big data and data minimization. In *DeLFI 2018-Die 16. E-Learning Fachtagung Informatik*; Gesellschaft für Informatik e.V.: Bonn, Germany, 2018; pp. 219–230.
14. El Fouki, M.; Aknin, N.; El Kadiri, K.E. Multidimensional Approach Based on Deep Learning to Improve the Prediction Performance of DNN Models. *Int. J. Emerg. Technol. Learn.* **2019**, *14*, 30. [CrossRef]
15. Hussain, S.; Muhsin, Z.; Salal, Y.; Theodorou, P.; Kurtoğlu, F.; Hazarika, G. Prediction model on student performance based on internal assessment using deep learning. *Int. J. Emerg. Technol. Learn.* **2019**, *14*, 4–22. [CrossRef]
16. Ajibade, S.S.M.; Ahmad, N.B.B.; Shamsuddin, S.M. Educational data mining: Enhancement of student performance model using ensemble methods. *Iop Conf. Ser. Mater. Sci. Eng.* **2019**, *551*, 012061. [CrossRef]
17. Tomasevic, N.; Gvozdenovic, N.; Vranes, S. An overview and comparison of supervised data mining techniques for student exam performance prediction. *Comput. Educ.* **2020**, *143*, 103676. [CrossRef]
18. Hooshyar, D.; Pedaste, M.; Yang, Y.; Malva, L.; Hwang, G.J.; Wang, M.; Lim, H.; Delev, D. From gaming to computational thinking: An adaptive educational computer game-based learning approach. *J. Educ. Comput. Res.* **2021**, *59*, 383–409. [CrossRef]
19. Waheed, H.; Hassan, S.U.; Aljohani, N.R.; Hardman, J.; Alelyani, S.; Nawaz, R. Predicting academic performance of students from VLE big data using deep learning models. *Comput. Hum. Behav.* **2020**, *104*, 106189. [CrossRef]
20. Barros, T.M.; Neto, P.A.S.; Silva, I.; Guedes, L.A. Predictive Models for Imbalanced Data: A School Dropout Perspective. *Educ. Sci.* **2019**, *9*, 275. [CrossRef]
21. Mduma, N. Data Balancing Techniques for Predicting Student Dropout Using Machine Learning. *Data* **2023**, *8*, 49. [CrossRef]
22. Wongvorachan, T.; He, S.; Bulut, O. A Comparison of Undersampling, Oversampling, and SMOTE Methods for Dealing with Imbalanced Classification in Educational Data Mining. *Information* **2023**, *14*, 54. [CrossRef]
23. Chen, W.; Yang, K.; Yu, Z.; Shi, Y.; Chen, C.P. A survey on imbalanced learning: Latest research, applications and future directions. *Artif. Intell. Rev.* **2024**, *57*, 137. [CrossRef]
24. Altalhan, M.; Algarni, A.; Alouane, M.T.H. Imbalanced data problem in machine learning: A review. *IEEE Access* **2025**, *13*, 13686–13699. [CrossRef]
25. El-Deeb, O.M.; Elbadawy, W.; Elzanfaly, D.S. The effect of imbalanced classes on students' academic performance prediction. *Int. J. e-Collab.* **2022**, *18*, 1–20. [CrossRef]
26. Alija, S.; Beqiri, E.; Gaafar, A.S.; Hamoud, A.K. Predicting students performance using supervised machine learning based on imbalanced dataset and wrapper feature selection. *Informatica* **2023**, *47*, 11–19. [CrossRef]
27. Fachrie, M.; Musdholifah, A.; Pulungan, R. Effectiveness of data resampling and ensemble learning in multiclass imbalance learning. *Appl. Soft Comput.* **2024**, *146*, 110596. [CrossRef]
28. Jain, A.; Dubey, A.K.; Khan, S.; Panwar, A.; Alkhatib, M.; Alshahrani, A.M. A PSO weighted ensemble framework with SMOTE balancing for student dropout prediction in smart education systems. *Sci. Rep.* **2025**, *15*, 97506. [CrossRef]

29. Koller, D.; Ng, A. The online revolution: Education for everyone. In Proceedings of the 22nd ACM International Conference on Conference on Information & Knowledge Management, San Francisco, CA, USA, 27 October–1 November 2013.
30. Sokolova, M.; Lapalme, G. A systematic analysis of performance measures for classification tasks. *Inf. Process. Manag.* **2009**, *45*, 427–437. [CrossRef]
31. Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. SMOTE: Synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **2002**, *16*, 321–357. [CrossRef]
32. Mansourifar, H.; Shi, W. Deep synthetic minority over-sampling technique. *arXiv* **2020**, arXiv:2003.09788. [CrossRef]
33. Saxena, D.; Cao, J. Generative adversarial networks (GANs) challenges, solutions, and future directions. *ACM Comput. Surv. (CSUR)* **2021**, *54*, 1–42. [CrossRef]
34. Creswell, A.; White, T.; Dumoulin, V.; Arulkumaran, K.; Sengupta, B.; Bharath, A.A. Generative adversarial networks: An overview. *IEEE Signal Process. Mag.* **2018**, *35*, 53–65. [CrossRef]