https://dx.doi.org/10.4314/dujopas.v10i4b.24
ISSN (Print): 2476-8316
ISSN (Online): 2635-3490
Dutse Journal of Pure and Applied Sciences (DUJOPAS), *Vol. 10 No. 4b December 2024*

Evaluating the Significance of Data Engineering Techniques in Multi-Class Prediction: Multi-Factor Educational Data Mining Experiments

A.Z. Umar[1*], H.S. Tuge[2], Y.G. Ibrahim[1]
[1]Department of Software Engineering and Cybersecurity,
Al-Qalam University
Katsina

[2]Department of Software Engineering,
Cosmopolitan University
Abuja

Email: azumar@auk.edu.ng

Abstract

Artificial Intelligence, particularly predictive modelling, is increasingly influencing education. For instance, a specific algorithm predicted with 74% accuracy the students that would fail within three weeks of the course. These results could lead to interventions that promote inclusivity and personalized learning, supporting the UN's goals of quality education and reducing inequalities. While predictive analytics holds great promise for education, datasets often suffer from small sample sizes and class imbalances which can result in inaccurate predictions and biased machine learning models. In this study, we evaluate the significance of various data engineering techniques in the context of educational data mining using a multi-factor supervised learning experiment. We applied data augmentation and balancing techniques to assess their impact on model performance. Additionally, data discretization for continuous features and feature selection, to identify the most relevant features for model training, were implemented and evaluated. The experimental design followed a 2 X 2 X 3 X3 factorial structure, incorporating different combinations of these techniques. We employed three models: Random Forest, Decision Tree, and Feed Forward Neural Network. The performance was measured using accuracy and F1 score metrics. The results also show that the data augmentation and balancing techniques seem to improve testing accuracy and F1 scores slightly, particularly for simpler models like Decision Trees. Feedforward Neural Networks perform more consistently across different datasets, while Decision Trees and Random Forests are more prone to overfitting, particularly without proper data balancing or augmentation.

Keywords: Data Engineering, Feature Selection, Data Augmentation, Educational Mining

INTRODUCTION

There has been growing evidence of the impact of Artificial Intelligence in education especially in predictive modelling(Bates et al., 2020; Gkontzis et al., 2022). For instance, (Akçapınar et al., 2019) conducted a study with 76 second-year university students registered in a computing course at two Peruvian universities from 2020-2022. They estimated that 74% of the students who were unsuccessful at the end of term could be accurately predicted through the use of a specific algorithm in as short as 3 weeks from the beginning of the course. Similarly, (Tsai et al., 2020)investigated the use of big data and artificial intelligence to predict

Author for Correspondence

dropout probabilities. Using data from 3,552 students at a Taiwanese university, statistical and deep learning methods were employed. The results showed accuracies of 68% and 77% for the statistical and deep learning methods, respectively. These findings could inform interventions that could improve inclusiveness and adaptation to individual learners' differences thereby contributing to achieving SDGs Goal No.4 (Quality Education) and Goal No. 10(Reduced Inequalities).

Despite the enormous potential of predictive analytics in educational mining and the increasing availability of educational data, educational datasets often present significant challenges, including small sample sizes and class imbalances, which can adversely affect the performance of machine learning models (Ghosh et al., 2024; Rekha et al., 2021; Selim & Rezk, 2023). These issues can lead to inaccurate predictions and biased learning models(Thabtah et al., 2020; Thölke et al., 2023). In the context of educational data mining, a machine learning model that fails to accurately predict student performance can have serious consequences. For example, it might mistakenly identify struggling students as high achievers or vice versa. This can lead to students not receiving the appropriate support they need. Similarly, the same model can misclassify a successful student, potentially limiting their educational opportunities.

Many studies proposed different data engineering techniques to remedy the problem. For the challenge of a small dataset, data engineering techniques exist for generating synthetic datasets(Ghaleb et al., 2023; Majeed & Hwang, 2023a). These techniques can increase the dataset size, balance the distribution of different student types, and ultimately enhance the model's accuracy. For the imbalance datasets, studies proposed different data engineering techniques and data pre-processing pipelines to address it(Ashraf et al., 2020; Bujang et al., 2021; López-García et al., 2023; Walid et al., 2022). For instance, (Ashraf et al., 2020)employed a multi-classifier ensemble approach to enhance accuracy in students' performance prediction using a pedagogical dataset that was compiled by the University of Kashmir. Similarly, feature selection and data balancing in combination were proposed in(Ghaleb et al., 2023). In the study, after pre-processing the data, Multiple Criteria Decision Making (MCDM) methods were utilized for feature extraction. In addition, Adaptive Synthetic Sampling (ADASYN)(López-García et al., 2023) was employed to balance the dataset through oversampling. At the end of the pipeline, Extreme Gradient Boosting (XGBoos) was then used to build an ensemble of decision trees. However, the results in (Ashraf et al., 2020; López-García et al., 2023) could have been influenced by the interaction with the ensemble technique and there is the need to test the data engineering techniques, separate from the ensembling.

A more comprehensive study (Zhang et al., 2023) explored the combination of data re-sampling and feature selection and which technique should be applied before the other by applying feature selection before or after data re-sampling in a large number of experiments, with a total of 9225 tests, on 52 publicly available datasets. The study used different feature selection and data resampling methods on three classification algorithms. They found that neither method consistently outperformed the other, suggesting that both should be considered when working with imbalanced data. This implies that more studies are needed to either generalize the results obtained the previous study (Zhang et al., 2023) or refute them. Despite significant advances in leveraging machine learning for educational data mining, existing studies often focus on either ensemble methods or tuning model parameters without systematically examining the individual and combined effects of foundational data engineering techniques. Many works report improvements primarily in training accuracy, which may indicate overfitting rather than genuine model robustness. Additionally, the majority of prior studies address only one or two data challenges (e.g., class imbalance or small dataset size) without exploring their interplay or implications for multiple model types. Our study addresses these gaps by rigorously evaluating data discretization, feature selection, data balancing, and data augmentation both in isolation and in combination. Unlike previous efforts, this work adopts a factorial experimental design, and ensures unbiased model comparisons through hyperparameter optimization. The novelty of this study lies in its holistic approach to assessing data engineering impacts across both shallow models (Decision Tree, Random Forest) and deep models (Feedforward Neural Network), thus providing a comprehensive understanding of how these techniques enhance model reliability and generalization.

The motivation for this paper stems from the critical need to improve the performance and reliability of machine learning models used in educational data mining. Most existing studies that report high accuracy often reflect the training accuracy(Guabassi et al., 2021), which suggests that the model may have overfitted to the noise in the training data. This overfitting is particularly concerning in high-stakes educational settings where misclassifications can significantly impact students' learning opportunities and support systems. The ultimate goal is to assess the effectiveness of data augmentation, data balancing, and feature selection techniques in improving the performance of supervised learning models for multi-class prediction tasks, thereby contributing to better educational outcomes and data-driven decision-making in education.

Thus, the following are the specific objectives of the paper.

1. To compare the performance of Random Forest, Decision Tree, and Neural Network models in terms of accuracy and F1 score when applied to small and unbalanced datasets.
2. To investigate the extent to which data augmentation techniques can reduce overfitting in supervised learning models.
3. To determine the effectiveness of data balancing techniques in enhancing the F1 scores of multi-class prediction models.
4. To evaluate how feature selection contributes to model efficiency and accuracy in multi-class prediction tasks.

The structure of the remainder of the paper is as follows: The next section provides the theoretical foundation for the components utilized in the Methodology section. Section 3.0 (Methodology) elaborates on the methods employed in the study. Section 4 presents the results, followed by discussions. Finally, Section 5 concludes the paper.

THEORETICAL ANALYSIS

This section presents the theoretical basis of the components utilized in the Methodology section and the metrics used in evaluating the machine learning model.

Hyperparameter tuning using Grid Search cross-validation:

Hyperparameters are settings that determine the structure and behaviour of a machine learning model. Hyperparameters are set before training begins as they control the learning process itself. Thus, hyperparameter tuning is the process of finding the best settings for a model's hyperparameters(Feurer & Hutter, 2019).

Grid Search Cross-Validation is an automated approach that aims to find the optimal hyperparameters $\emptyset^*$ that minimize the loss function L. More specifically, given the parameter grid:

Ƿ:Ƿ $=\{(\emptyset_{1,1,}\emptyset_{1,2,}\dots,\emptyset_{1,n}),(\emptyset_{2,1,}\emptyset_{2,2,}\dots,\emptyset_{2,n}),\dots,(\emptyset_{m,1,}\emptyset_{m,2,}\dots,\emptyset_{m,n})\}$……………………..(1)
Where m, in (1) above is the number of combinations in the grid. Then for each $pi\ \epsilon$Ƿ: the model, M, is trained with hyperparameters $\emptyset_i$ on k fold cross validation using the dataset, D, and the average loss Li is calculated across all folds. The combination of p^* that minimizes loss:$p^* = argmin_{p\in P\ L_i}$ are then selected. The optimal hyperparameters $\emptyset^*$ are the hyperparameters corresponding to:
$p^* : \emptyset = (\emptyset_{p,1,}\emptyset_{p,2,}\dots,\emptyset_{p,n})$… (2)

Equal-width feature discretization
For equal-width discretization, the range of values for each feature is divided into equal-width intervals. The width of each bin or interval given by Equation (3)
$\Delta x = \frac{x_{max}-x_{min}}{n_{bins}}$…………………………………………………(3)

Where $x_{max}$the maximum value of a feature is is, x_{min} is the minimum value of a feature, n_{bins} is the number of bins (intervals)

Gini index for determining feature importance
Gini Index $G(F) = 1 - \sum_{i=1}^{K} p^i$ …………….. (4)

Where: p^i is the proportion of samples in a feature set F that belong to class $i\ and\ \ K$ is the total number of classes. Gini index is a measure of impurity or entropy and the feature importance is on how much it reduces the impurity across all the nodes of the trees in which it appears because it contributes more to the overall predictive power of the model.

Multiclass prediction
In machine learning, when the goal is to predict a categorical outcome (one of several possible choices) by training a model on labelled data, the task is called a classification problem. If there are only two possible outcomes, such as 0 and 1, true or false, or positive and negative, the problem is referred to as a binary classification problem. However, if the goal is to predict one of more than two possible outcomes, it is known as a multi-class classification problem. More specifically, let $X = (X_{1,}\ X_{2,}\dots,X_{p,})$ represent the vector of p predictor variables (features) and Y represent the response variable, where Y takes values in the set $\{1,2,\dots,K\}$ with K being the number of classes in the multi-class classification problem, the goal of training the machine learning model is to find a function $F(X)$ that maps the feature vector X to a predicted class label Ŷ.
Formally, the predicted outcome, Ŷ, is such that:
Ŷ= $arg_{k\in\{1,2,\dots K\}}max\ \ P(Y = k|X = x)$ ………………………………………………(5)

Equation (5) represents the class $k \in K$ that maximizes the conditional probability Y belonging to class k given the observed features $X = x$.

2.4.1 Metrics

$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ nymbe\ of\ predictions}$ X 100 ……………………………………….(6)

$F1 = 2\text{X}\frac{Precision\ X\ Recall}{Precicion+Recall}$…………………………………………………………..… (7)

The Precision and the Recall are given in Equations (7) and (8) respectively

Precision $= \frac{True\ \ Positive\ Predictions}{True\ Positive\ Predictions+False\ Positive\ Predictions}$………………………..(8)

$$Recall = \frac{True\ Positive\ Predictions}{True\ Positive\ Predictions + False\ Negative\ Predictions} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (9)$$

METHODOLOGY

Data collection, pre-processing, augmentation, and balancing

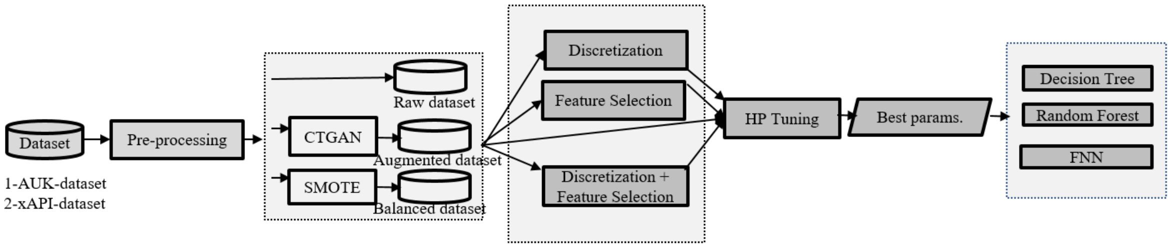

Figure 1: Summary of the methods used

Figure 1 represents the flowchart of the methods used. Two datasets were used in the study. The first dataset (AUK-dataset) was collected from the final year computing students of Al-Qalam University Katsina, Nigeria after getting clearance from the Research Ethics Committee of the University and as part of other research(Umar & Ado, 2021, 2022) to understand students' pain points. Relevant protocols of the Nigerian Data Protection Regulations were also strictly observed(Abubakar et al., 2022). As the sample drawn was only for the students who undertook software development projects, the size was found to be small, 96 records to be precise, which can be prone to overfitting the machine learning model. In addition, the AUK dataset was imbalanced as the grades were mostly As, followed by Bs and the small number of Cs. The second dataset (xAPI-dataset) was the academic performance dataset obtained from a public repository[1]. As shown in Figure 1, at the pre-processing step, the data was pre-processed to remove null values, especially in the AUK dataset as some students were found to have dropped one course or the other.

Algorithm 1: Data augmentation using CTGAN

```
Input: Original dataset D
Output: Augmented dataset D_aug
1)  Create metadata
2)  Train CTGAN on dataset D using the created metadata
3)  Generate synthetic samples using trained CTGAN
4)  Combine original dataset D and synthetic samples to form D_aug
5)  Return: D_aug
```

To augment the datasets, the Conditional Tabular Generative Adversarial Network (CTGAN)(Xu et al., 2019) (see Algorithm 1), which extends the conventional Generative Adversarial Network (GAN) framework(Goodfellow et al., 2020; Gui et al., 2023). CTGAN was used to ensure that the synthetic samples not only resemble the marginal distribution of the original data but also maintain conditional dependencies between attributes. As presented in Step 4, and Step 5 of Algorithm 1, the synthetically generated samples and the original dataset were combined to produce a new dataset that would be used for training the models.

[1] https://www.kaggle.com/datasets/aljarah/xAPI-Edu-Data

Similarly, the Synthetic Minority Over-sampling Technique (SMOTE) (Elreedy & Atiya, 2019) was used to balance the dataset (see Algorithm 2). SMOTE is a popular data balancing technique which works by creating synthetic data points for the minority class, effectively increasing its size and reducing the imbalance between the classes. Algorithm 2 represents the processes of balancing the dataset using SMOTE. As shown in Step 3 of Algorithm 2, minority samples were generated to balance the dataset.

Algorithm 2: Data balancing using SMOTE

```
Input: Dataset D with minority class imbalance
1)  Output: Balanced dataset D_bal
2)  Encode categorical features
3)  Initialize the SMOTE algorithm
4)  For each minority class sample:
        a)  generate synthetic samples by interpolating between it and its nearest
            neighbors
5)  Add synthetic samples to original dataset D
6)  Return: D_bal
```

Consequently, after the data augmentation and the data balancing, there are three versions of each of the two datasets: raw dataset, dataset augmented with CTGAN, and dataset balanced with SMOTE.

For the Discretization, equal-width discretization was implemented (see Equation (3) in section 2.3 and Algorithm 3) in such a way that it could be toggled on/off in the course of the experiments.

For the feature Selection, we utilized Recursive Feature Elimination (RFE)(Ramírez-Hernández & Fernandez, 2007) to determine the most important features for the use in Decision Tree and Random Forest(Feature Selection in Figure 1). RFE considers feature interactions and dependencies(Chen & Jeong, 2007) and is also a flexible technique compatible with various machine learning algorithms and metrics. In this study, we leveraged the Gini Index (see Equation (4) in section 2.3 above) for determining feature importance in RFE, as employed in the Decision Tree(Myles et al., 2004).

As shown in Figure 1, the combination of discretization and feature selection was also evaluated. It should also be noted that, for comparison, the models were initially trained and evaluated on the datasets before applying the discretization and the feature selection.

Algorithm 3: Data balancing using SMOTE

```
    Input: Feature vector X
1)  Output: Discretized feature vector X_disc
2)  Define the number of bins n_bins
3)  Calculate the bin width using Equation (3)
4)  For each feature value x in X:
        a.  Assign x to a bin based on its range
5)  Return: X_disc
```

Algorithm 4 Feature Selection using Recursive Feature Elimination Algorithm

```
1)  $X_0 = X$
2)  $M_0 = TrainInitialModel(X_{0,y})$
```

```
3) $I = calculateFeatureImportance(M_0)$
4) $X_0 = EliminateLeastImportantFeatures(X_0, I)$
5) $M_1 = TrainModel(X_1, y)$
6) Repeat:
    a) $X_{i+1} = EliminateLeastImportantFeatures(X_{i+1}, I)$
    b) $M_{i+1} = TrainModel(X_{1+1}, y)$
7) Until: $X=k$
8) $SelectedFeatures = FeaturesFrom(X_{final})$
```

Algorithm 4 was adopted from(Umar et al., 2023) for feature selection. In algorithm 4, X_0 is the initial feature matrix (a matrix where each row represents a sample and each column represents a feature). M_0 is the initial model; M_i is the model at ith iteration; X_i is the feature matrix at ith iteration. y is the target variable (a vector containing the labels or values you're trying to predict) and k is the number of features to select or retain.

Before training and evaluating each of the models, the best hyperparameters were obtained using the Grid Search cross-validation library (GridSearchCV[2]) (See Equation 1) and parameterized with 5 cross validation (see HP tuning in Figure 1). The optimal parameters represented in Equation 2 were used to train the respective models.

Machine learning models

As shown in Figure 1, three machine learning modelling techniques were considered for this study: Decision Trees(Vanneschi & Silva, 2023), Random Forest(Breiman, 2001), and Feedforward Neural Networks(Gabella, 2021). These modelling techniques are elaborated in the following subsections:

Decision Tree

Decision Tree provide a transparent and interpretable approach to modelling complex relationships between variables. A Decision Tree predicts the output by splitting the dataset based on feature values to minimize impurity. In this study, the Gini Index, as presented in Equation (4) was used as the measure of impurity to determine the optimal split point for a node. The Decision Tree recursively splits the feature space, choosing the feature and threshold that minimize the Gini Index at each node. The tree grows until a stopping criterion is met (e.g., max depth)(Vanneschi & Silva, 2023).

Random Forest

Random Forest is an ensemble method that leverages multiple decision trees to enhance predictive accuracy and robustness (Majeed & Hwang, 2023b). Each tree is trained on a random subset of features and samples from the dataset. Prediction in Random Forest is determined by majority voting(Breiman, 2001):

$\hat{y} = mode(T1(x), T2(x), \ldots, Tn(x))$ -- (10)

$T_{i(x)}$ in Equation (10) is the prediction of the i-th tree.

Feedforward Neural Network (FNN)

An FNN consists of multiple layers: an input layer, hidden layers, and an output layer. Each layer applies a linear transformation followed by a non-linear activation function(Gabella, 2021). In this study, the first hidden layer contains 32 neurons and utilizes the ReLU activation function. To enhance training stability and generalization, a batch normalization layer is placed after this layer. A second dropout layer with a 30% dropout rate is then applied to

[2] https://scikit-learn.org/stable/modules/generated/sklearn.mod el_selection.GridSearchCV.html

further regularize the model. The final hidden layer also consists of 32 neurons with ReLU activation. The output layer is a dense layer with the number of neurons corresponding to the number of classes in the dataset. A softmax activation function was applied to generate class probabilities. The model is compiled using categorical cross-entropy as the loss function, the Adam optimizer, and accuracy as the evaluation metric. To optimize training efficiency and prevent overfitting, early stopping, and learning rate reduction techniques are employed. Feature selection and discretization were not applied to the FNN as they are less relevant.
Consequent to the above, the experimental design was treated as a 2 × 2 × 3 × 3 factorial design with the following factors and levels:

1. Feature Selection: Two levels - with and without feature selection.
2. Discretization: Two levels - with and without discretization.
3. Data Augmentation/Balancing: Three levels - no augmentation, augmentation using GAN, and balancing using SMOTE (considered a form of augmentation).
4. Model Type: Three levels - Decision Tree, Random Forest, and Feedforward Neural Network.

Each combination was evaluated across multiple datasets to assess the impact on model performance, measured by training accuracy, testing accuracy, and F1 score.

RESULTS AND DISCUSSION

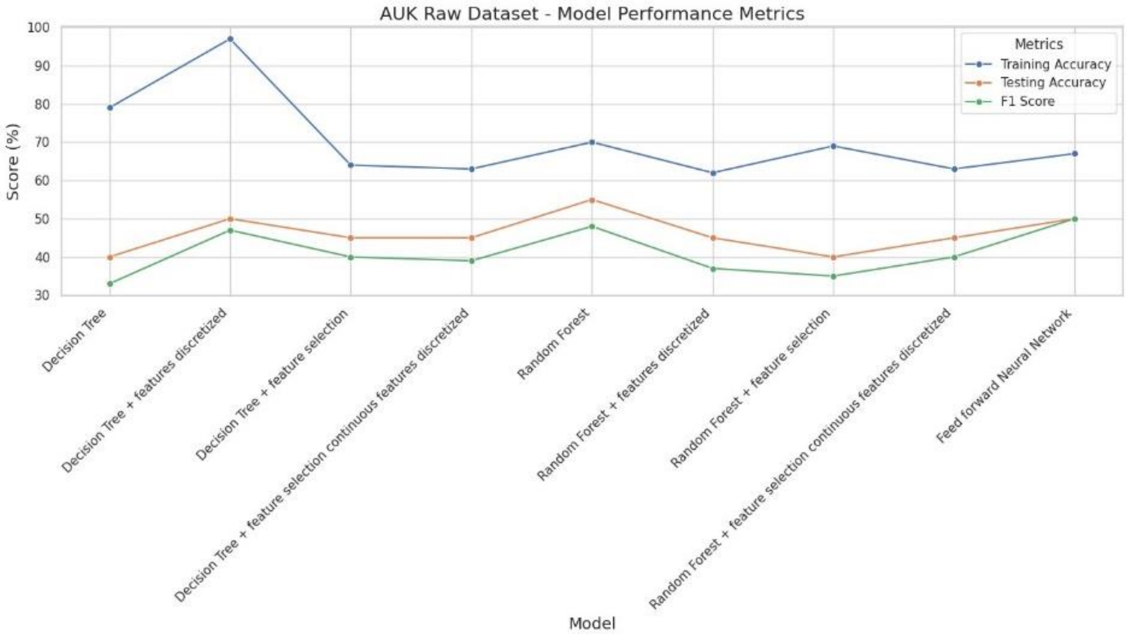

Figure 2: Models' performance on the Al-Qalam University raw dataset

Looking at Figure 2, for the Decision Tree, high training accuracy (97%) can be observed when features are discretized, but testing accuracy is moderate (50%). The F1 score is relatively low (47), indicating possible overfitting. The Random Forest shows better generalization than the Decision Tree, with a higher testing accuracy (55%) and F1 score (48%) without feature selection or discretization. For the Feedforward Neural Network, a balance between training and testing accuracy (67% and 50%, respectively) can be observed, with an F1 score of 50, indicating it performs consistently across training and testing.

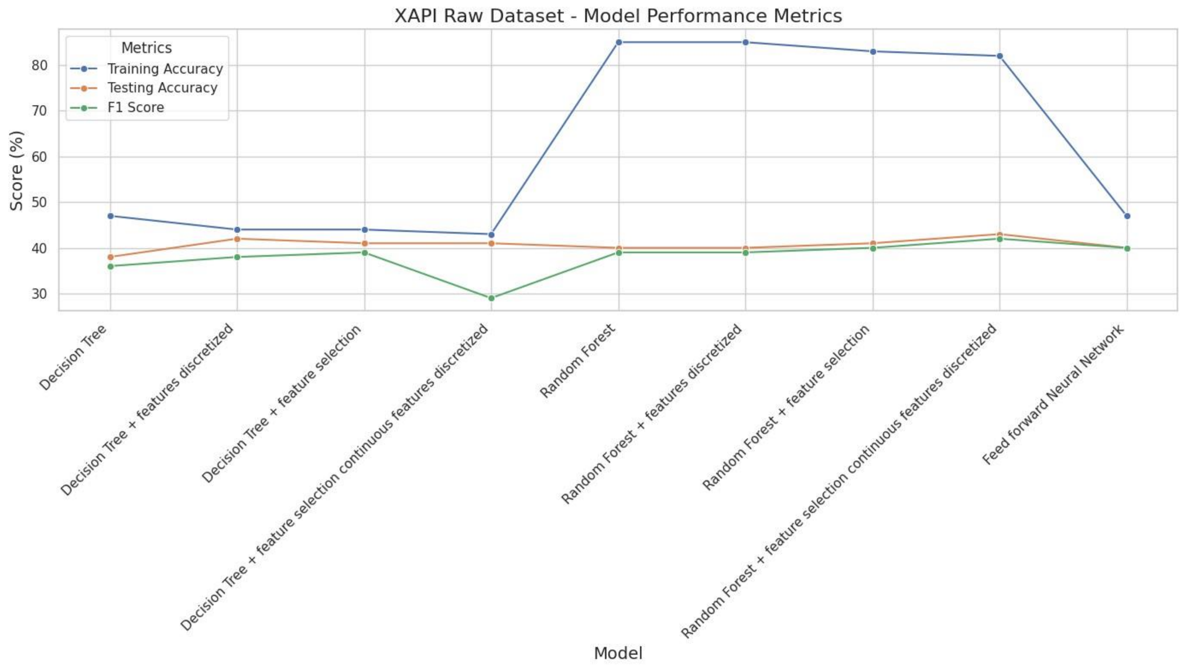

Figure 3: Models' performances on student's academic performance dataset

Looking at Figure 3, Decision Tree generally performs low across the board, with the highest testing accuracy being 42% and F1 score being 39. Discretization improves performance slightly. For the Random Forest, it achieves high training accuracy (up to 85%). However, the testing accuracy and F1 scores remain low (40%-43%), again indicating potential overfitting. The Feedforward Neural Network achieves a balance between training and testing with both accuracies around 47%-50%, and an F1 score of 40.

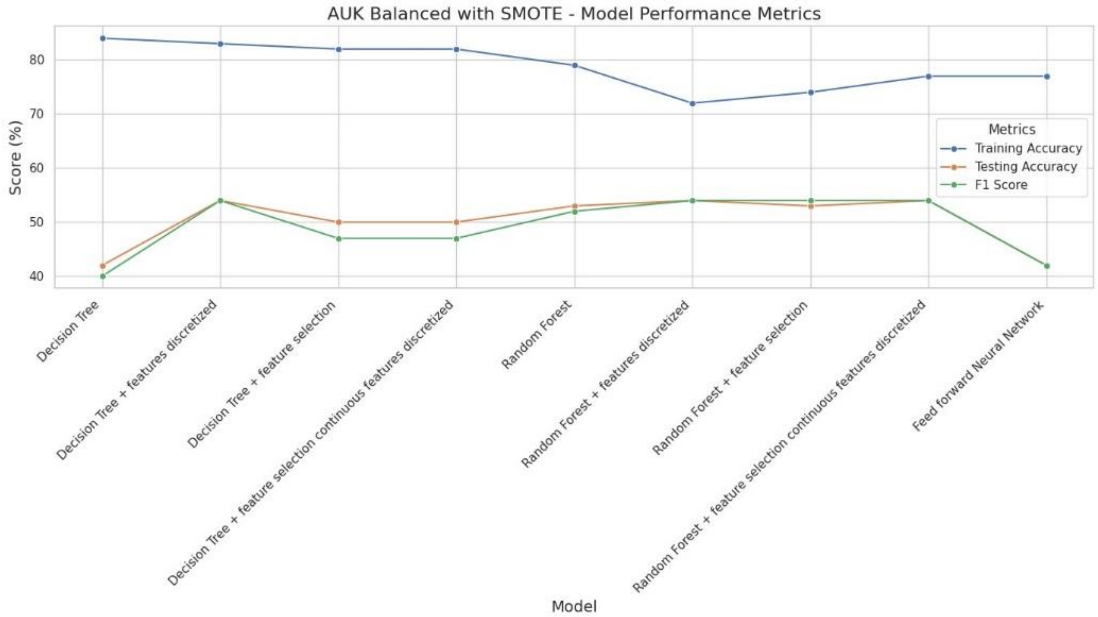

Figure 4: Models' performances on the Al-Qalam University dataset balanced with SMOTE

Looking at Figure 4, the Decision Tree exhibits improved testing accuracy (up to 54%) and F1 score (54) with discretization, showing that balancing the dataset helps. The Random Forest shows consistent performance with F1 scores up to 54% and testing accuracy also improving significantly (up to 54%). The Feedforward Neural Network maintains consistent training and testing accuracy of 77%-42%, but the F1 score is lower (42).

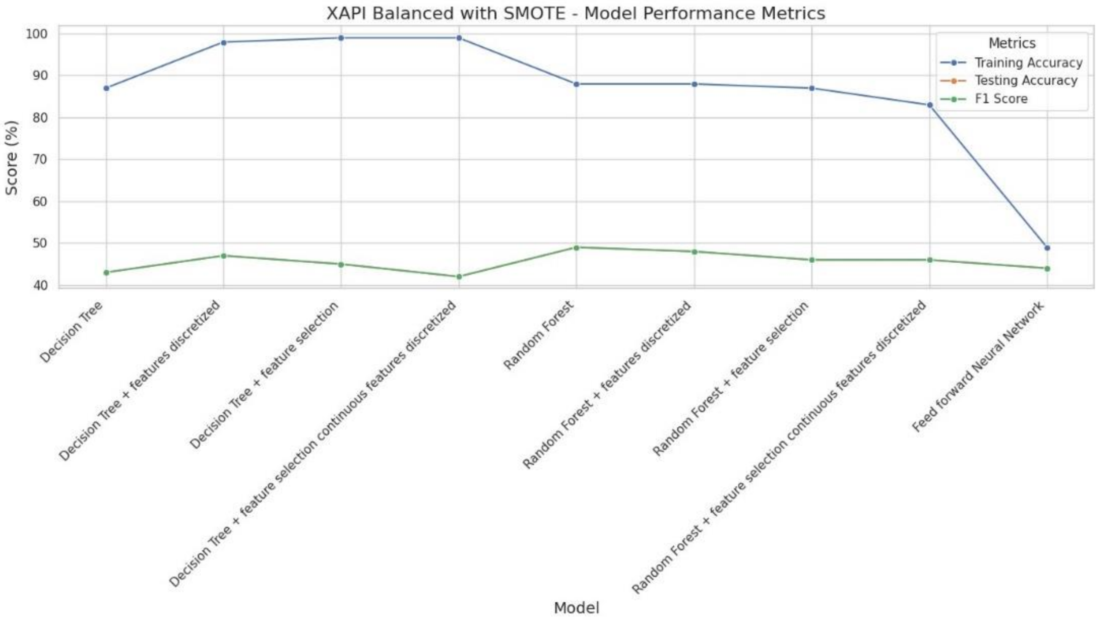

Figure 5: Models' performances on students' academic performance dataset

In Figure 5, the Decision Tree recorded training accuracy of up to 99% with discretization, but testing accuracy remains low (42%-47%), again highlighting overfitting. F1 scores are consistent with testing accuracy. For the Random Forest, better generalization can be observed with testing accuracy and F1 scores around 49%-49%. For the Feedforward Neural Network, low training accuracy (49%) was recorded, but relatively balanced testing accuracy (44%) and F1 score (44).

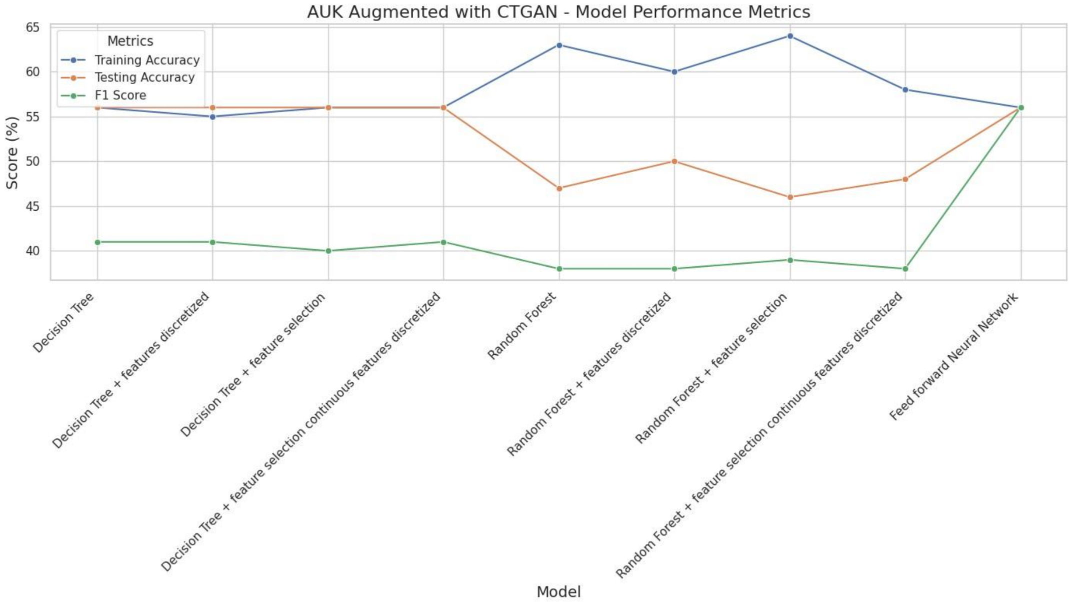

Figure 6: Models' performances on Al-Qalam University dataset that has been augmented with CTGAN

In Figure 6, the Decision Tree shows consistent training and testing accuracy at 56%, with F1 scores around 40-41%. The Random Forest's testing accuracy is slightly lower (47%-50%) and F1 scores are in the 38-39% range, indicating that CTGAN augmentation might not have provided significant improvement for this model in this case. The same Figure 6 shows Feedforward Neural Network achieves balanced performance with 56% in all metrics, which might suggest good generalization from training to testing.

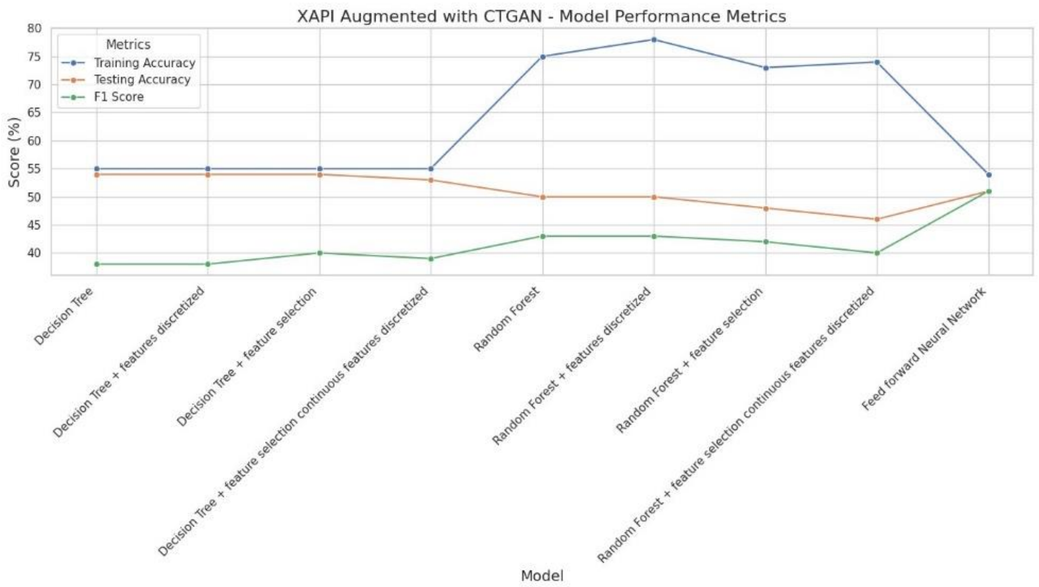

Figure 7: Models' performances of student academic performance dataset that has been augmented with CTGAN

Figure 7 shows that the Decision Tree's testing accuracy is consistent with training (54%-55%), but F1 scores are lower (38%-40%), showing only modest performance. The Random Forest shows some overfitting with higher training accuracy (75%-78%) but moderate testing accuracy (50%) and F1 scores (40%-43%). For the Feedforward Neural Network, the result shows a balance across metrics with 54%-51% and an F1 score of 51, indicating decent performance with the augmented data.

The results obtained underscore that while Decision Trees and Random Forests often achieve high training accuracy, their testing accuracy frequently falls short, a clear indicator of overfitting as observed in(Guabassi et al., 2021). This overfitting is particularly pronounced when these models are applied to raw, unbalanced datasets. In contrast, Feedforward Neural Networks consistently exhibit a more balanced performance across training and testing, suggesting a stronger generalization capability, which is crucial for real-world applications. Data augmentation techniques, particularly CTGAN, were evaluated for their ability to mitigate overfitting. The results show a modest reduction in overfitting, with slight improvements in testing accuracy and F1 scores for simpler models like Decision Trees. However, the improvement is not uniform across all models, indicating that while augmentation helps, it is not a panacea for overfitting in complex models like Random Forests. This is in line with the findings in(Zhang et al., 2023) The application of SMOTE for data balancing demonstrated significant enhancements in both testing accuracy and F1 scores. This improvement was particularly notable for Decision Trees, where the balanced datasets led to better generalization and reduced overfitting. The Random Forest also benefited from SMOTE, but to a lesser extent, indicating that while data balancing is effective, its impact varies depending on the model complexity and the inherent characteristics of the dataset(Zhang et al., 2023). Feature selection played a pivotal role in improving model efficiency and accuracy. For Decision Trees and Random Forests, the application of feature selection, particularly when combined with discretization, led to better performance metrics. This highlights the importance of carefully selecting and engineering features, especially in multi-class prediction tasks where irrelevant features can detract from model performance.

From the results, it can be summarized that many models, especially Decision Trees and Random Forests, show high training accuracy but lower testing accuracy, indicating overfitting. However, the Feedforward Neural Networks tend to have more balanced performance across training and testing, suggesting they may generalize better. The results also show that the data augmentation and balancing techniques like SMOTE and CTGAN augmentation seem to improve testing accuracy and F1 scores slightly, particularly for simpler models like Decision Trees. Overall, the choice of pre-processing (like feature discretization and selection), data balancing (SMOTE), and augmentation (CTGAN) impacts the model's ability to generalize to unseen data. Feedforward Neural Networks seem to perform more consistently across different datasets, while Decision Trees and Random Forests are more prone to overfitting, particularly without proper data balancing or augmentation.

CONCLUSION

In conclusion, this paper assessed the effectiveness of feature discretization, data augmentation, data balancing, and feature selection techniques in improving the performance of supervised learning models for multi-class prediction tasks. This study reaffirms the critical role of data engineering techniques, including discretization, feature selection, balancing, and augmentation, in enhancing the performance and generalization of machine learning models. While Feedforward Neural Networks demonstrate robust and consistent performance across different datasets, traditional models like Decision Trees and Random Forests require more careful handling to avoid overfitting. The findings suggest that a tailored approach, combining appropriate data engineering techniques with model selection, is essential for achieving optimal performance in predictive analytics, particularly when dealing with small and unbalanced datasets. This study provides a roadmap for practitioners aiming to enhance model reliability and accuracy in real-world applications.

REFERENCES

Abubakar, M. M., Armaya'u, Z. U., & Abubakar, M. (2022). Personal Data and Privacy Protection Regulations: State of compliance with Nigeria Data Protection Regulations (NDPR) in Ministries, Departments, and Agencies (MDAs). *2022 5th Information Technology for Education and Development (ITED)*, 1–6.

Akçapınar, G., Altun, A., & Aşkar, P. (2019). Using learning analytics to develop early-warning system for at-risk students. *International Journal of Educational Technology in Higher Education*, *16*(1). https://doi.org/10.1186/s41239-019-0172-z

Ashraf, M., Zaman, M., & Ahmed, M. (2020). An Intelligent Prediction System for Educational Data Mining Based on Ensemble and Filtering approaches. *Procedia Computer Science*, *167*, 1471–1483. https://doi.org/10.1016/j.procs.2020.03.358

Bates, T., Cobo, C., Mariño, O., & Wheeler, S. (2020). Can artificial intelligence transform higher education? In *International Journal of Educational Technology in Higher Education* (Vol. 17, Issue 1). https://doi.org/10.1186/s41239-020-00218-x

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Bujang, S. D. A., Selamat, A., Ibrahim, R., Krejcar, O., Herrera-Viedma, E., Fujita, H., & Ghani, N. A. M. (2021). Multiclass Prediction Model for Student Grade Prediction Using Machine Learning. *IEEE Access*, *9*, 95608–95621. https://doi.org/10.1109/ACCESS.2021.3093563

Chen, X. W., & Jeong, J. C. (2007). Enhanced recursive feature elimination. *Proceedings - 6th International Conference on Machine Learning and Applications, ICMLA 2007*. https://doi.org/10.1109/ICMLA.2007.44

Elreedy, D., & Atiya, A. F. (2019). A Comprehensive Analysis of Synthetic Minority Oversampling Technique (SMOTE) for handling class imbalance. *Information Sciences*, *505*, 32–64. https://doi.org/10.1016/j.ins.2019.07.070

Feurer, M., & Hutter, F. (2019). *Hyperparameter Optimization*. https://doi.org/10.1007/978-3-030-05318-5_1

Gabella, M. (2021). Topology of Learning in Feedforward Neural Networks. *IEEE Transactions on Neural Networks and Learning Systems*, *32*(8), 3588–3592. https://doi.org/10.1109/TNNLS.2020.3015790

Ghaleb, F. A., Saeed, F., Al-Sarem, M., Qasem, S. N., & Al-Hadhrami, T. (2023). Ensemble Synthesized Minority Oversampling-Based Generative Adversarial Networks and Random Forest Algorithm for Credit Card Fraud Detection. *IEEE Access*. https://doi.org/10.1109/ACCESS.2023.3306621

Ghosh, K., Bellinger, C., Corizzo, R., Branco, P., Krawczyk, B., & Japkowicz, N. (2024). The class imbalance problem in deep learning. *Machine Learning*, *113*(7), 4845–4901. https://doi.org/10.1007/s10994-022-06268-8

Gkontzis, A. F., Kotsiantis, S., Panagiotakopoulos, C. T., & Verykios, V. S. (2022). A predictive analytics framework as a countermeasure for attrition of students. *Interactive Learning Environments*, *30*(6), 1028–1043. https://doi.org/10.1080/10494820.2019.1709209

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2020). Generative adversarial networks. *Communications of the ACM*, *63*(11), 139–144. https://doi.org/10.1145/3422622

Guabassi, I. El, Bousalem, Z., Marah, R., & Qazdar, A. (2021). Comparative Analysis of Supervised Machine Learning Algorithms to Build a Predictive Model for Evaluating Students' Performance. *International Journal of Online and Biomedical Engineering*. https://doi.org/10.3991/ijoe.v17i02.20025

Gui, J., Sun, Z., Wen, Y., Tao, D., & Ye, J. (2023). A Review on Generative Adversarial Networks: Algorithms, Theory, and Applications. *IEEE Transactions on Knowledge and Data Engineering*, *35*(4), 3313–3332. https://doi.org/10.1109/TKDE.2021.3130191

López-García, A., Blasco-Blasco, O., Liern-García, M., & Parada-Rico, S. E. (2023). Early detection of students' failure using Machine Learning techniques. *Operations Research Perspectives*, *11*. https://doi.org/10.1016/j.orp.2023.100292

Majeed, A., & Hwang, S. O. (2023a). CTGAN-MOS: Conditional Generative Adversarial Network Based Minority-Class-Augmented Oversampling Scheme for Imbalanced Problems. *IEEE Access*. https://doi.org/10.1109/ACCESS.2023.3303509

Majeed, A., & Hwang, S. O. (2023b). Quantifying the Vulnerability of Attributes for Effective Privacy Preservation Using Machine Learning. *IEEE Access*, *11*, 4400–4411. https://doi.org/10.1109/ACCESS.2023.3235016

Myles, A. J., Feudale, R. N., Liu, Y., Woody, N. A., & Brown, S. D. (2004). An introduction to decision tree modeling. In *Journal of Chemometrics* (Vol. 18, Issue 6, pp. 275–285). https://doi.org/10.1002/cem.873

Ramírez-Hernández, J. A., & Fernandez, E. (2007). Control of a re-entrant line manufacturing model with a reinforcement learning approach. *Proceedings - 6th International Conference on Machine Learning and Applications, ICMLA 2007*, 330–335. https://doi.org/10.1109/ICMLA.2007.35

Rekha, G., Tyagi, A. K., Sreenath, N., & Mishra, S. (2021). Class Imbalanced Data: Open Issues and Future Research Directions. *2021 International Conference on Computer Communication and Informatics, ICCCI 2021*. https://doi.org/10.1109/ICCCI50826.2021.9402272

Selim, K. S., & Rezk, S. S. (2023). On predicting school dropouts in Egypt: A machine learning approach. *Education and Information Technologies*, *28*(7), 9235–9266. https://doi.org/10.1007/s10639-022-11571-x

Thabtah, F., Hammoud, S., Kamalov, F., & Gonsalves, A. (2020). Data imbalance in classification: Experimental evaluation. *Information Sciences*.

https://doi.org/10.1016/j.ins.2019.11.004

Thölke, P., Mantilla-Ramos, Y. J., Abdelhedi, H., Maschke, C., Dehgan, A., Harel, Y., Kemtur, A., Mekki Berrada, L., Sahraoui, M., Young, T., Bellemare Pépin, A., El Khantour, C., Landry, M., Pascarella, A., Hadid, V., Combrisson, E., O'Byrne, J., & Jerbi, K. (2023). Class imbalance should not throw you off balance: Choosing the right classifiers and performance metrics for brain decoding with imbalanced data. *NeuroImage*. https://doi.org/10.1016/j.neuroimage.2023.120253

Tsai, S. C., Chen, C. H., Shiao, Y. T., Ciou, J. S., & Wu, T. N. (2020). Precision education with statistical learning and deep learning: a case study in Taiwan. *International Journal of Educational Technology in Higher Education*, *17*(1). https://doi.org/10.1186/s41239-020-00186-2

Umar, A. Z., & Ado, S. G. (2021). Emergency Remote Learning During COVID-19 Lockdown: Al-Qalam University Katsina Students' Experience. In Prof. Afolayan A. Obiniyi, Prof. Rasheed Gbenga Jimoh, Dr. Uyinomen O. Ekong, Prof. Steve Adesina, & Prof. Folorunsho Olaiya (Eds.), *International Conference on Information Technology in Education and Development (ITED)* (pp. 167–174).

Umar, A. Z., & Ado, S. G. (2022). Emergency remote teaching during COVID-19 lockdown: Al-Qalam University Katsina lecturers' experience. *Bayero Journal of Pure and Applied Sciences*, *13*(1), 393–399.

Umar, A. Z., Galadima Ibrahim, Y., & Ndanusa, A. (2023). Detecting Anomalies In Network Traffic Using a Hybrid of Linear-based and Tree-based Feature Selection Approaches. *Researchgate.NetYG Ibrahim, A Ndanusaresearchgate.Net*, 21–23.

Vanneschi, L., & Silva, S. (2023). Decision Tree Learning. In *Natural Computing Series* (pp. 149–159). https://doi.org/10.1007/978-3-031-17922-8_6

Walid, M. A. A., Ahmed, S. M. M., Zeyad, M., Galib, S. M. S., & Nesa, M. (2022). Analysis of machine learning strategies for prediction of passing undergraduate admission test. *International Journal of Information Management Data Insights*, *2*(2). https://doi.org/10.1016/j.jjimei.2022.100111

Xu, L., Skoularidou, M., Cuesta-Infante, A., & Veeramachaneni, K. (2019). Modeling tabular data using conditional GAN. *Advances in Neural Information Processing Systems*, *32*.

Zhang, C., Soda, P., Bi, J., Fan, G., Almpanidis, G., García, S., & Ding, W. (2023). An empirical study on the joint impact of feature selection and data resampling on imbalance classification. *Applied Intelligence*, *53*(5), 5449–5461. https://doi.org/10.1007/s10489-022-03772-1