



Classification of Educational Vulnerability Levels Using Boosting Methods on Imbalance Socio-Economic Data

Andi Illa Erviani Nensi¹, Meavi Cintani^{1*}, Mahda Al Maida¹, A. Qeis Tenridapi¹, Bagus Sartono¹, Aulia Rizki Firdawanti¹, Budi Susetyo¹, and Gerry Alfa Dito¹

¹ *Statistics and Data Science Study Program, IPB University*

Abstract

Extreme class imbalance in education data causes classification models to tend to only recognize students who are still in school and fail to detect students who are not in school, even though this group is the one most in need of intervention. This study aims to develop a classification model capable of identifying educational vulnerability in school-age children with a data imbalance ratio of approximately 1:48. Four boosting methods were tested, namely AdaBoost-M2, SMOTEBoost, RBBoost, and RUSBoost using a Stratified 5-Fold Cross Validation scheme. Model performance was evaluated using metrics relevant to imbalanced data, including sensitivity, specificity, balanced accuracy, and Area Under Curve (AUC). The results show that SMOTEBoost L50 with a threshold of 0.50 is the best model with a sensitivity of 0.8824, balanced accuracy of 0.7474, and AUC of 0.7745, making it the most effective in detecting vulnerable students without an extreme drop in accuracy. Feature importance analysis revealed that education level or class was the most dominant predictor, followed by household size and frequency of internet use. Overall, this approach offers strong potential to be applied as an early detection system to support more targeted educational interventions for children at risk of dropping out of school.

Keywords: boosting; class imbalance; educational vulnerability; SMOTEBoost; student dropout detection.

Copyright © 2025 by Authors, Published by CAUCHY Group. This is an open access article under the CC BY-SA License (<https://creativecommons.org/licenses/by-sa/4.0>)

1 Introduction

Education is a fundamental aspect that determines the quality of human resources and serves as an essential foundation for employment opportunities and social mobility. The Indonesian government has implemented a 12-Year Compulsory Education program to ensure equal access to education for all children [1]. However, this policy has not fully addressed disparities in educational access across regions. Data from the Central Statistics Agency (BPS) indicate that some adolescents aged 13–18 years remain vulnerable to educational exclusion, either due to the risk of not attending school or discontinuing schooling altogether [2]. Previous studies suggest that educational vulnerability is strongly influenced by socio-economic factors, particularly among low-income households, families whose heads work in the informal sector, and communities residing in areas with limited educational infrastructure [3].

*Corresponding author. E-mail: meavi2501cintani@apps.ipb.ac.id

Educational vulnerability is not only associated with formal access to schooling but also encompasses households' capacity to bear indirect educational costs, such as transportation, learning materials, and access to technology that supports the learning process [4]. Adolescents who face barriers to education tend to have lower literacy levels and limited competencies, which reduces their opportunities to obtain employment in the formal sector and increases the risk of being trapped in low-income jobs [5]. This situation systematically reinforces the cycle of intergenerational poverty and poses a major challenge to achieving the Sustainable Development Goals (SDGs), particularly Goal 4 on inclusive and quality education [6].

The National Socioeconomic Survey (SUSENAS) provides comprehensive and nationally representative household-level data that are highly relevant for analyzing educational vulnerability in Indonesia. Key variables include ownership of the Indonesia Smart Card (KIP), participation in the Indonesia Smart Program (PIP), housing conditions, asset ownership, literacy skills, internet access, and school characteristics. These indicators collectively offer a multidimensional representation of the factors influencing adolescents' schooling status. SUSENAS-based analysis enables systematic and objective identification of vulnerable youth groups, thereby supporting evidence-based policy formulation and targeted educational interventions.

Modeling educational vulnerability presents a significant methodological challenge due to severe class imbalance [7]. Empirically, the proportion of adolescents who are not attending school is substantially smaller than that of those who remain enrolled, resulting in a minority class that is underrepresented in the data. This imbalance causes conventional classification algorithms to favor the majority class, leading to poor sensitivity or low recall in detecting minority-class observations [8]. Such prediction failures are particularly critical in public policy contexts, as misclassification of vulnerable individuals may result in ineffective targeting and exacerbate existing social inequalities [6].

Boosting methods have been developed to address imbalanced data in various domains, such as health, fraud detection, and security systems. However, their application in the context of educational vulnerability remains limited, particularly in studies utilizing SUSENAS data as a source of socio-economic information. This limitation indicates an important research gap that warrants further investigation. Accordingly, this study aims to fill this gap by developing an educational vulnerability classification model using several boosting algorithms specifically designed to address class imbalance. In addition, this study seeks to provide empirical insights into the socio-economic factors that most strongly influence the risk of adolescents not attending school. The findings are expected to support the formulation of more targeted and effective educational policies for adolescents who are most vulnerable to educational exclusion.

2 Methods

2.1 Data

This study utilizes data from the 2024 National Socioeconomic Survey (SUSENAS) conducted by the Central Statistics Agency (BPS). The dataset is employed to analyze the socio-economic profiles of households and individuals in relation to educational vulnerability, referred to in this study as the *Education Vulnerability Level*. The dataset contains information on educational status, demographic characteristics, and household socio-economic conditions that are closely associated with educational participation. The unit of analysis in this study is individuals aged 13–18 years were selected, representing the secondary education age group. After completing data cleaning, age filtering, and variable selection stages, a total of 2,781 observations were obtained. The dataset consists of 17 independent variables and one dependent variable, namely school participation status.

Table 1: Research variables

Variables	Information	Scale
Y	School participation status	Binary
X_1	Amount of PIP assistance	Ratio
X_2	Number of household members	Ratio
X_3	Sources of household financing	Nominal
X_4	Savings account ownership	Binary
X_5	Asset/home ownership	Binary
X_6	Ability to read Latin letters	Binary
X_7	Ability to read Arabic/Hijaiyah letters	Binary
X_8	Ability to read other letters	Binary
X_9	Internet usage at home	Binary
X_{10}	Schooling status in the previous academic year	Nominal
X_{11}	Highest level/class of education ever/currently attended	Ordinal
X_{12}	School status	Binary
X_{13}	Ownership of the Smart Indonesia Card (KIP)	Nominal
X_{14}	Experience using the internet	Binary
X_{15}	Frequency of internet use	Ratio
X_{16}	Type of residential area	Binary
X_{17}	Ownership of another house by head of household/spouse/child	Binary
X_{18}	Respondent's marital status	Nominal
X_{19}	Respondent's gender	Binary

2.2 Research Stages

2.2.1 Data Preparation

Data preparation involved normalization, removal of problematic features, and transformation of the target variable into binary classes. To ensure balanced class distribution during the modeling process, the dataset was divided using *Stratified 5-Fold Cross Validation*, which preserves the proportion of each class across all folds prior to model training [9].

2.2.2 Data Exploration

Initial data exploration was conducted to examine the imbalanced class structure and to identify variables that exhibit strong associations with the target variable. This step aims to understand the underlying data characteristics and to assess the severity of class imbalance, which necessitates the application of class imbalance handling techniques and feature selection prior to model development.

2.2.3 SMOTEBoost

SMOTEBoost combines the AdaBoost ensemble learning mechanism with synthetic minority oversampling using the Synthetic Minority Over-sampling Technique (SMOTE) at each boosting iteration. This integration produces a more balanced class distribution during the training process. The SMOTEBoost predictor generates class labels based on weighted log-likelihood weak learners at each boosting iteration [10]. The SMOTEBoost procedure consists of the following steps:

1. Identifying minority class samples.
2. Generating synthetic samples using SMOTE.
3. Combining real and synthetic data.
4. Each weak learner uses a decision tree of maximum depth two (depth = 2), with sample weights updated at each iteration.

5. Calculate the error and update the distribution weights with the formula:

$$\beta = \frac{\epsilon}{1 - \epsilon} \quad (1)$$

6. Updating weights by going through the equation

$$D_i \leftarrow D_i \cdot \beta^{(1-P(h(x_i)=y_i))} \quad (2)$$

2.3 AdaBoost-M2

AdaBoost-M2 is implemented using two main functions, namely *AdaBoostM2()* which runs the boosting process by adjusting the weights based on the pseudo-loss function, and *AdaBoostM2_predict* [11] which produces the final prediction using the aggregation of weak learners' weights.

In the boosting iteration, the following stages are carried out:

1. Initializing the initial sample weights uniformly.
2. Training a weak learner (decision tree with depth 2) using sample weights.
3. Calculating the pseudo-loss using the following formula:

$$\varepsilon_t = \frac{1}{2} \sum_{i=1}^N \sum_{y \neq y_i} D_t(i, y) (1 - h_t(x_i, y_i) + h_t(x_i, y)) \quad (1)$$

4. Calculating the model weight:

$$\alpha_t = \ln \left(\frac{1 - \varepsilon_t}{\varepsilon_t} \right) \quad (2)$$

5. Updating the distribution weights:

$$D_{t+1}(i, y) = \frac{D_t(i, y) \cdot \exp(\alpha_t (1 - h_t(x_i, y_i) + h_t(x_i, y)))}{Z_t} \quad (3)$$

The final weights of the weak learners are used to determine predictions through a weighted voting mechanism.

2.4 RBBoost (Random Balance Boosting)

RBBoost is implemented through two main functions, namely *RBBoost()* to train the boosting model with random undersampling and oversampling without fixed rules, and *RBBoost_predict()* to produce the final prediction using weighted majority voting [12].

At each iteration, the following steps are performed:

1. Calculating the imbalance class ratio in the dataset.
2. Determining a new random sampling proportion such that the class distribution lies within the range of 40–60%.
3. Forming a new dataset from a combination of random undersampling of the majority class and random oversampling of the minority class.
4. Training a weak learner (decision tree with depth 2) using the balanced sample results.
5. Calculating the model error using:

$$\varepsilon_t = \sum_{i=1}^N D_t(i) \mathbb{I}(h_t(x_i) \neq y_i) \quad (4)$$

6. Updating the weights, which are then normalized, and the process continues until the iteration is complete.

$$\beta_t = \frac{\varepsilon_t}{1 - \varepsilon_t} \quad (5)$$

$$D_{t+1}(i) = D_t(i) \cdot \beta_t^{1 - \mathbb{I}(h_t(x_i) = y_i)} \quad (6)$$

2.5 RUSBoost

The implementation of RUSBoost is carried out through two main functions, namely *RUSBoost()* for the training process and *RUSBoost_predict()* to produce the final prediction [13]. In general, the boosting process is carried out with a reweighting mechanism similar to AdaBoost, but at each iteration, Random UnderSampling (RUS) is performed to balance the class distribution.

The stages of the RUSBoost algorithm are explained as follows:

1. Sample Weight Initialization

In the first iteration, the distribution weights D_i for all samples are given uniform initial values:

$$D_i = \frac{1}{N} \quad (7)$$

where N is the total number of samples in the training dataset.

2. Random UnderSampling (RUS)

Random UnderSampling (RUS) is a data balancing technique that reduces the number of observations in the majority class through random sample selection until the proportions of the majority and minority classes are balanced. Unlike synthetic oversampling approaches such as SMOTE, which balance data by adding synthetic samples to the minority class, RUS does not generate new data but achieves class balance by removing some data from the majority class. Thus, RUS has the potential to reduce information in the majority class, but does not change the distribution structure of the minority class.

3. Weak Learner Training

Decision tree models (usually with low depth, for example depth = 2) are trained using the undersampled data by taking into account the distribution weights D_i .

4. Error Calculation

After the model is trained, the error value is calculated based on the sample weights:

$$\varepsilon_t = \sum_{i=1}^N D_i \cdot \mathbb{I}(h_t(x_i) \neq y_i) \quad (8)$$

where the value of the indicator function $\mathbb{I}(\cdot)$ is 1 if the prediction is incorrect and 0 if it is correct.

5. Beta Value Calculation

The learning weight for weak learners is calculated using the boosting formula:

$$\beta_t = \frac{\varepsilon_t}{1 - \varepsilon_t} \quad (9)$$

6. Sample Weight Update

The sample weights are updated to place greater emphasis on misclassified data:

$$D_i \leftarrow D_i \cdot \beta_t^{\mathbb{I}(h_t(x_i)=y_i)} \quad (10)$$

If the prediction is correct, the weight decreases, and if the prediction is incorrect, the weight increases. The weights are then normalized as follows:

$$D_i = \frac{D_i}{\sum_{j=1}^N D_j} \quad (11)$$

7. Iteration

Steps 2–6 are repeated until the number of iterations (L) is reached.

8. Final Prediction

The final prediction uses a log-weighted voting function in *RUSBoost_predict()* by combining the predictions of each weak learner with weighting based on:

$$F(x) = \sum_{t=1}^L \log\left(\frac{1}{\beta_t}\right) h_t(x) \quad (12)$$

7. Hyperparameter Tuning

This is done to find the best combination of parameters in each model [14].

Table 2: Hyperparameter Tuning

Parameter	Variations Tried
Number of <i>Weak Learners</i> (L)	10, 50, 100
Probability Threshold	0.30, 0.50, 0.70

8. Model evaluation

Model evaluation is performed to determine the best performance under imbalanced data conditions. The evaluation process uses several metrics, namely *accuracy*, *precision*, *recall* (*True Positive Rate* / *TPR* or *sensitivity*), *specificity* (*True Negative Rate* / *TNR*), F1-score, and *Balanced Accuracy*.

9. Feature importance analysis on the best model

The final stage of the research was to analyze the best model based on the evaluation results from the previous stage. The two highest-performing models were compared, and then a *feature importance analysis* was performed using the aggregation weights of the *weak learners* in boosting.

3 Results and Discussion

3.1 Data Exploration

3.1.1 Class Imbalance

The analysis focused on the class imbalance of the target variable, namely schooling status. The minority class is defined as children who are not attending school, while the majority class is children who are still attending school. Of the total 2,781 observations, there are 57 children ($\pm 2.05\%$) who are not attending school as the minority class and 2,724 children ($\pm 97.95\%$) who are still attending school as the majority class. This comparison produces an imbalance ratio of approximately 1:48, indicating that the size of the minority class is much smaller than the majority class. This condition indicates that the dataset is categorized as highly class imbalance

and has the potential to affect the performance of the classification model if not specifically addressed.

3.1.2 Initial Analysis of Predictor Variables

The characteristics of the predictor variables were analyzed descriptively through data distribution, differences in patterns based on schooling status, and relationships between features. The results of this exploration served as the basis for assessing the potential contribution of each feature and identifying variables that require specific attention during the data preparation stage.

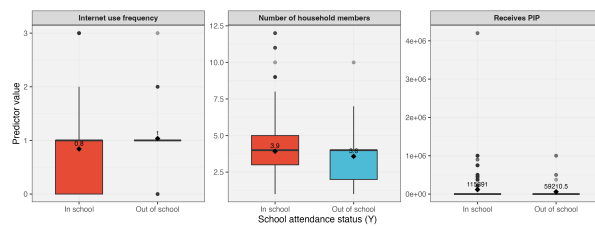


Figure 1: Boxplot of Numeric Variables Based on Schooling Status

As shown in Figure 1, the boxplot of numeric variables based on schooling status shows a fairly wide distribution and several *outliers* in the three main numeric variables: PIP assistance, number of household members, and frequency of internet use. The PIP assistance variable is dominated by the value 0, indicating that the majority of observations did not receive this assistance. The distribution patterns between children who are still in school and those who are not in school appear to overlap, so that when these three numeric variables are observed separately, it is not possible to clearly distinguish between the two groups of schooling status.

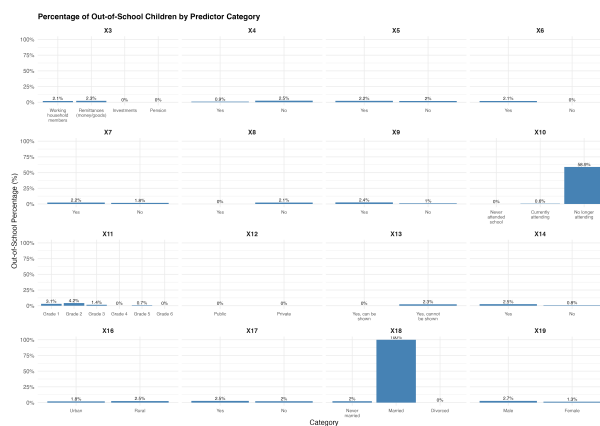


Figure 2: Percentage of Children Not Attending School by Predictor Variable Category

In almost all predictor variable categories, the percentage of children out of school is relatively small. Even in categories representing vulnerable conditions, such as children who have never or are no longer attending school (X10), low-education classes (X11), and rural areas (X16), the proportion of children out of school remains low. This pattern indicates a strong class imbalance, with the number of children still in school far outnumbering those who are not.

Cramér's V values indicate that most categorical predictor variables have a weak relationship with schooling status, with values generally below 0.10. However, two variables, namely X12 and X10, show Cramér's V values that are much higher than the other variables. The high level of association in these two variables indicates the existence of overlapping information with the response variable, which has the potential to cause *information leakage* if they are still used as predictors.

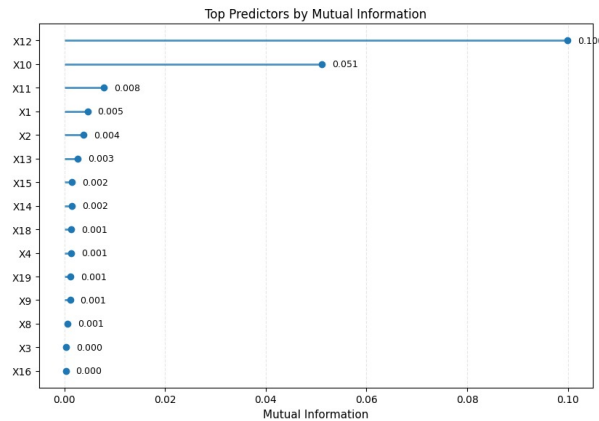


Figure 3: Association of categorical variables with schooling status.

3.2 Data Preparation

Based on the results of feature selection using Cramér’s V value in Subsection 3.1.2, the two features with the highest level of association with schooling status (X12 and X10) were removed from the predictor set, leaving 17 features from 2,781 observations. All numeric features were normalized using StandardScaler to be on a comparable scale, with a mean close to 0 and a standard deviation close to 1. After normalization, the data were split using a *stratified 5-fold cross validation scheme*, as summarized in Table 3.

Table 3: Distribution of classes in each fold

Fold	Train	Test	Minority Train	Minority Test
1	2224	557	45 (2%)	12 (2.2%)
2	2225	556	46 (2.1%)	14 (2%)
3	2225	556	46 (2.1%)	15 (2.2%)
4	2225	556	46 (2.1%)	10 (2%)
5	2225	556	45 (2%)	11 (2.2%)

As shown in Table 3, each fold contains approximately 2,224–2,225 observations for the training data and 556–557 observations for the test data. The number of minority classes (children not attending school) in the training data ranges from 45–46 cases (± 2.0 –2.1%), while in the test data it is 10–15 cases (± 2.0 –2.2%). This uniform proportion indicates that stratification successfully maintains the class distribution close to the initial distribution, with an imbalance ratio of approximately 1:48 consistent across all folds. This condition is important to ensure that the model performance evaluation is not biased towards a particular data partitioning scheme, even if the minority class size is very small.

3.3 Random Forest

Random Forest was used as a baseline model to provide an initial overview of the performance of the classification method without explicit class imbalance handling mechanisms. This method constructs multiple decision trees using bootstrap aggregation (bagging) techniques and randomly selects a subset of variables at each node split, then combines all tree predictions using a majority rule. This approach is generally effective in reducing overfitting and capturing nonlinear relationships between predictor variables. All Random Forest configurations were identified and evaluated using metrics across thresholds and number of estimators, so their results are not shown in Table 4.

The Random Forest performance evaluation results presented in Table 4 show that the model produces very high accuracy (0.9796) and a True Negative Rate (TNR) of 1.00. However, the True Positive Rate (TPR) and F1-score each reach 0.00, resulting in a Balanced Accuracy of

Table 4: SMOTEBoost performance evaluation with various thresholds

Model	Accuracy	TPR	TNR	F1-Score	Balanced Accuracy
Random Forest (Baseline)	0.9796	0.0000	1.0000	0.0000	0.5000

only 0.50. This pattern indicates that the model consistently predicts only the majority class, without successfully identifying a single observation from the minority class, namely children who do not attend school.

This performance was affected by the extreme class imbalance in the research data. Under these conditions, the Random Forest learning process was dominated by the majority class, resulting in misclassifications in the minority class contributing very little to the formation of the tree structure. As a result, while the overall accuracy appeared excellent, the model's ability to detect educational vulnerability was severely limited and inconsistent with the primary research objective.

Based on these results, Random Forest is considered less effective as a primary predictive model, but remains relevant as a baseline for comparison. These results confirm that high accuracy on imbalanced data does not necessarily reflect a model's ability to recognize minority classes, and highlight the importance of using methods specifically designed to address class imbalance, such as SMOTEBoost, in the context of identifying educational vulnerability.

3.4 SMOTEBoost

SMOTEBoost combines the AdaBoost framework with SMOTE-based synthetic sample generation to improve the model's ability to recognize minority classes. At each iteration, the model not only updates the weights of misclassified observations but also adds synthetic samples from the minority class to more balance the training data distribution. Table 4 presents the performance evaluation of SMOTEBoost based on variations in the number of estimators (L) and three probability threshold scenarios.

Table 5: Performance evaluation of SMOTEBoost with various thresholds

Model Configuration	Threshold	Accuracy	TPR	TNR	F1-Score	Balanced Accuracy
SMOTEBoost L10	0.30	0.7569	0.5294	0.7616	0.0814	0.6455
	0.50	0.9796	0.0000	1.0000	0.0000	0.5000
	0.30	0.6168	0.8824	0.6112	0.0857	0.7468
SMOTEBoost L50	0.50	0.6180	0.8824	0.6125	0.0860	0.7474
	0.70	0.6180	0.8824	0.6125	0.0860	0.7474
	0.30	0.5581	0.8824	0.5513	0.0752	0.7168
SMOTEBoost L100	0.50	0.6180	0.8824	0.6125	0.0860	0.7474
	0.70	0.6180	0.8824	0.6125	0.0860	0.7474

The SMOTEBoost model shows a clear performance difference between *weak learner* configurations. At L10, a threshold of 0.50–0.70 yields very high accuracy (≈ 0.98), but the TPR and F1-score for the minority class are 0, so the model only predicts the majority class and does not detect out-of-school children at all (*Balanced Accuracy* 0.50). This condition is influenced by the extremely class imbalance in the data, as the model's learning process is dominated by the majority class and prediction errors in the minority class contribute very little to the loss function, especially at relatively high thresholds. As a result, despite seemingly good accuracy, the model's ability to recognize the minority class is very low.

Lowering the threshold to 0.30 increases the TPR to 0.5294 and the Balanced Accuracy to 0.6455, although with a decrease in overall accuracy. Configurations with L50 and L100 show stable TPR values around 0.88 and Balanced Accuracy values in the range of 0.72–0.75 for all threshold values, so both configurations are able to detect minority classes than L10. Based on the combined accuracy and sensitivity, SMOTEBoost L50 with a threshold of 0.50 is chosen

as the main configuration because its performance is equivalent to L100 but with lower model complexity, and is considered more representative for detecting educational vulnerabilities.

3.5 AdaBoost-M2

AdaBoost-M2 is used as an error-weighted *boosting* approach without explicitly manipulating the class distribution. This mechanism gives misclassified observations greater weight in subsequent iterations, allowing the model to focus more on difficult-to-predict cases, including minority classes. Table 6 shows the performance evaluation of the AdaBoost-M2 model.

Table 6: Performance evaluation of AdaBoost-M2 with various thresholds

Model Configuration	Threshold	Accuracy	TPR	TNR	F1-Score	Balanced Accuracy
AdaBoost-M2 L10	0.30	0.8970	0.3529	0.9083	0.1224	0.6306
	0.50	0.9796	0.0000	1.0000	0.0000	0.5000
	0.70	0.9796	0.0000	1.0000	0.0000	0.5000
AdaBoost-M2 L50	0.30	0.7150	0.6471	0.7164	0.0846	0.6817
	0.50	0.9784	0.0000	0.9988	0.0000	0.4994
	0.70	0.9796	0.0000	1.0000	0.0000	0.5000
AdaBoost-M2 L100	0.30	0.6647	0.7059	0.6638	0.0789	0.6848
	0.50	0.9784	0.0000	0.9988	0.0000	0.4994
	0.70	0.9796	0.0000	1.0000	0.0000	0.5000

At thresholds of 0.50–0.70, the model’s accuracy appears very high (97–98%), but its sensitivity to minority classes is zero, resulting in the model completely failing to detect cases of educational vulnerability. This indicates an *accuracy paradox*, a situation where accuracy appears good while its actual classification ability is low due to extreme class imbalance. Balanced accuracy, which is only around 0.50, reinforces the evidence that at the default threshold, the model tends to make homogeneous predictions to the majority class and does not learn patterns that represent vulnerable students.

When the threshold is lowered to 0.30, there is a significant increase in sensitivity across all *weak learner* (L) configurations. For the L10 configuration, Balanced Accuracy reaches 0.6306, while for the L50 and L100 configurations, it increases to 0.6817–0.6848, albeit with a decrease in specificity due to increased false positives. This performance demonstrates that *threshold tuning* is more important for classification success than simply increasing the number of estimators. AdaBoost-M2 with a low threshold is able to compensate for bias towards the majority class by prioritizing detection of the minority class without sacrificing specificity excessively.

3.6 RBBoost

RBBoost (*Random Balance Boosting*) is a boosting approach designed to address class imbalance without removing observations from the majority class. This model increases the weight of minority class observations while giving additional weight to hard-to-classify majority class observations. This strategy makes RBBoost adaptive in learning minority patterns without losing majority class information. Table 7 presents the evaluation of RBBoost based on variations in the number of *weak learners* (L) and three probability threshold scenarios.

threshold of 0.50 provides the most stable performance (Accuracy = 92.57%; Balanced Accuracy = 0.5877) because it maintains sensitivity at a reasonable level without sacrificing specificity significantly.

3.7 RUSBoost

RUSBoost is a *boosting* method that combines *random undersampling* of the majority class and adaptive weighting, similar to conventional boosting algorithms. This approach is designed to reduce the dominance of the majority class in the learning process by trimming some of the

Table 7: RBBoost performance evaluation with various thresholds

Model Configuration	Threshold	Accuracy	TPR	TNR	F1-Score	Balanced Accuracy
RBBoost L10	0.30	0.7557	0.4120	0.7630	0.0642	0.5873
	0.50	0.9257	0.2350	0.9400	0.1143	0.5877
	0.70	0.9257	0.2350	0.9400	0.1143	0.5877
RBBoost L50	0.30	0.6802	0.2940	0.6880	0.0361	0.4912
	0.50	0.7305	0.2350	0.7410	0.0343	0.4881
	0.70	0.7916	0.1770	0.8040	0.0333	0.4904
RBBoost L100	0.30	0.2886	0.7060	0.2800	0.0388	0.4929
	0.50	0.4551	0.6470	0.4510	0.0461	0.5491
	0.70	0.6228	0.4710	0.6260	0.0483	0.5483

majority observations, thus increasing the exposure of the model to minority class patterns. Table 8 presents an evaluation of RUSBoost's performance based on the number of *weak learners* (L) and three probability threshold scenarios.

Table 8: RUSBoost performance evaluation with various thresholds

Model Configuration	Threshold	Accuracy	TPR	TNR	F1-Score	Balanced Accuracy
RUSBoost L10	0.30	0.7557	0.4120	0.7630	0.0642	0.5873
	0.50	0.9257	0.2350	0.9400	0.1143	0.5877
	0.70	0.9257	0.2350	0.9400	0.1143	0.5877
RUSBoost L50	0.30	0.6802	0.2940	0.6880	0.0361	0.4912
	0.50	0.7305	0.2350	0.7410	0.0343	0.4881
	0.70	0.7916	0.1770	0.8040	0.0333	0.4904
RUSBoost L100	0.30	0.2886	0.7060	0.2800	0.0388	0.4929
	0.50	0.4551	0.6470	0.4510	0.0461	0.5491
	0.70	0.6228	0.4710	0.6260	0.0483	0.5483

In general, RUSBoost exhibits a consistent performance pattern with RBBoost at a threshold of 0.50–0.70, maintaining sensitivity to the minority class ($\text{TPR} > 0$), even though accuracy is lower than AdaBoost-M2. This indicates that the undersampling mechanism is effective in reducing bias toward the majority class, but at the same time incurs *information loss* that affects overall performance. The performance of RBBoost and RUSBoost tends to be identical in the results of this study. This condition is caused by the mathematical consequences of the data structure with extreme class imbalance. Both algorithms undersample the majority class and retain all minority samples with almost the same ratio in each boosting iteration, so that the training subsets formed by both methods are very similar. In addition, the weight update mechanisms in RBBoost and RUSBoost have similar functional forms, so the learning process places overall emphasis on the minority class. As a result, the prediction patterns and evaluation metrics, especially sensitivity, specificity, Balanced Accuracy, and F1-Score, converged and produced almost the same performance across all configurations of the number of estimators and threshold variations.

3.8 Comparison and Selection of the Best Model

Figure 4 shows a comparison of the overall models to select the best method to use.

Based on the comparison of the best models from each boosting method in Figure 4, it can be seen that each model has different strengths and characteristics in handling class imbalance. AdaBoost-M2 L100 (threshold 0.30) shows high Balanced Accuracy (0.6848) with good sensitivity ($\text{TPR} = 0.7059$), but the overall accuracy is relatively low. RBBoost L10 and RUSBoost L10 (threshold 0.30) offer higher accuracy (0.7557) but lower sensitivity ($\text{TPR} = 0.412$), thus risking failure to recognize a large portion of vulnerable students. Among all models, SMOTEBoost L50 with a threshold of 0.50 is the best configuration because it produces the highest sensitivity ($\text{TPR} = 0.8824$) and the highest Balanced Accuracy (0.7474), making it most effective in identifying

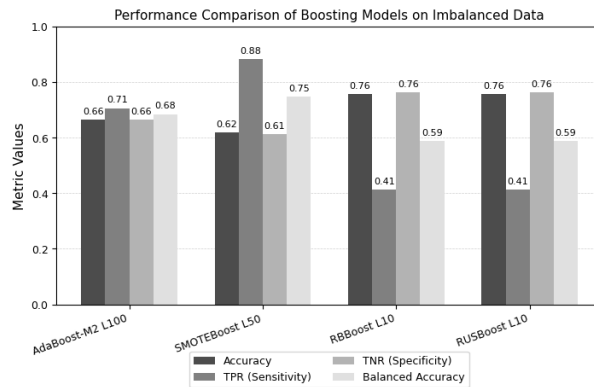


Figure 4: Comparison of boosting models based on imbalanced datasets

minority classes without significantly reducing specificity.

The results of this study are consistent with previous research [15], which shows that SMOTEBoost is able to significantly increase sensitivity and Balanced Accuracy in imbalanced data, surpassing AdaBoost and RUSBoost in most cases. In addition, AdaBoost's suboptimal performance in this research is in line with findings in [15], which confirm that the effectiveness of AdaBoost is highly dependent on parameter tuning and still tends to be biased toward the majority class in skewed data distributions. The low sensitivity results of RUSBoost in this study are also in line with findings in [8], which state that repeated undersampling mechanisms can eliminate majority information, causing the model to have difficulty recognizing minority patterns consistently. In addition, the trade-off pattern between increasing TPR and decreasing overall accuracy observed in this study is consistent with [15], which shows that boosting algorithms on imbalanced data tend to increase TPR at the cost of decreasing specificity. Thus, SMOTEBoost L50 (threshold 0.50) was chosen as the final model in this study because it provides the best performance for detecting educational vulnerability in imbalanced data, considering the balance between sensitivity, specificity, and overall classification robustness.

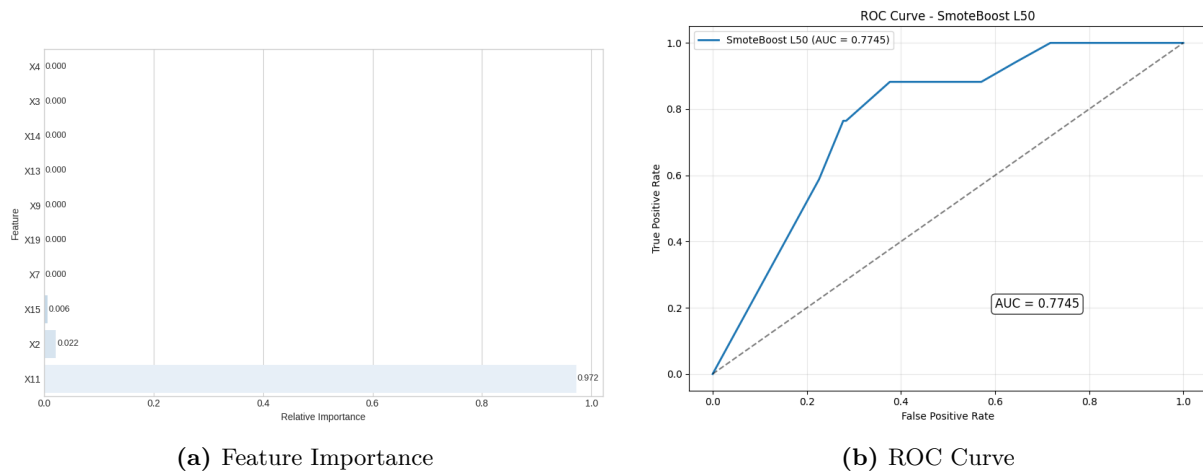


Figure 5: (a) Feature Importance and (b) ROC Curve SMOTEBoost L50

The feature importance analysis in Figure 5 shows that variable X11, namely the highest level/class of education ever or currently occupied, is the most dominant predictor in detecting educational vulnerability, contributing 97.2% to the model's inadequacy. This means that the respondent's grade level or level is the strongest indicator in distinguishing whether a student is at risk of not attending school/stopping continuing education. This makes empirical sense because students at higher levels of education have different risks than students at the elementary level, both in terms of academic burden, level transitions, and family economic pressure.

The next contributing variables, although relatively small, are X2 (number of household members/ART) and X15 (frequency of internet use). The X2 variable indicates that the larger the number of household members, the potential for economic burden and competing needs within the family increases, which can affect school sustainability. Meanwhile, X15 can describe high access to basic resources with better frequency of internet use and higher academic connectivity, thus being more protected from educational vulnerability. In contrast, other variables such as X3 (household financing source), X4 (account ownership), X7 (ability to read Latin/Arabic), X9 (home internet access), X13 (KIP ownership), and X19 (gender) do not contribute to the model's inadequacy (importance value 0), indicating that these variables do not provide additional information in separating vulnerable and non-vulnerable classes. This finding confirms that educational vulnerability in the context of this dataset is more determined by the student's position in the educational level and the structural economic conditions of the family, rather than by factors of financial access, gender, or non-Latin literacy.

The model's ability to distinguish between the two classes was further assessed using the ROC curve, as shown in Figure 5. The use of the ROC curve and AUC values in this study is more relevant than accuracy because the data used is highly imbalanced. Accuracy tends to be biased towards the majority class and can provide high scores even if the model fails to recognize the minority class [7]. In contrast, ROC evaluates the relationship between the True Positive Rate and False Positive Rate at various thresholds, thus independent of class proportions [16]. The AUC value represents the model's global discriminatory ability, namely the probability that the model scores observations in the positive class higher than those in the negative class. Therefore, ROC-AUC provides a fairer and more informative evaluation for assessing the model's performance in detecting vulnerable students in imbalanced educational data. The ROC curve shows that the model line is located well above the reference diagonal line, with an Area Under Curve (AUC) of 0.7745. This value illustrates the model's strong discriminatory capacity, where the model's probability of correctly classifying individuals between vulnerable and non-vulnerable classes is relatively high.

The classification performance reaches 77.45% in terms of AUC across various threshold values. The pattern of increasing True Positive Rate as False Positive Rate increases also demonstrates the model's consistent performance across various misclassification scenarios. Thus, the combination of high AUC values and excellent sensitivity reinforces the conclusion that SMOTEBoost L50 (threshold 0.50) is the most suitable model for use as the final model in this study to detect educational vulnerability in highly imbalanced data.

4 Conclusion

This study aims to develop a classification model capable of detecting educational vulnerability in data with very high class imbalance. Through a comparative evaluation of four *boosting* methods, it was found that handling class imbalance is crucial for modeling success, not only based on accuracy but especially on the ability to recognize minority classes. The results confirmed that SMOTEBoost L50 with a threshold of 0.50 is the most effective approach because it produces the best balance between sensitivity and specificity, and demonstrates strong discriminatory ability in distinguishing vulnerable and non-vulnerable students. The main contribution of this study lies in empirical evidence that synthetic oversampling-based *boosting techniques* are more adaptive to educational contexts with skewed data distributions than standard *boosting* methods and undersampling-based approaches.

These findings imply that the resulting model can be utilized as an early detection system to support the formulation of more targeted educational intervention policies. However, this study still has limitations in using cross-sectional data and does not consider school or environmental contextual variables. Further research can expand the scope with longitudinal data, integration of contextual features, and development of *cost-sensitive models* to improve predictive ability

and policy relevance.

CRediT Authorship Contribution Statement

Andi Illa Erviani Nensi: Conceptualization, Methodology, Software, Formal Analysis, Data Curation, Visualization, Writing–Original Draft Preparation.

Meavi Cintani: Conceptualization, Methodology, Validation, Writing–Review & Editing, Supervision, Project Administration.

Mahda Al Maida: Software, Validation, Visualization, Writing–Review & Editing.

A. Qeis Tenridapi: Data Curation, Investigation, Formal Analysis, Writing–Review & Editing.

Bagus Sartono: Conceptualization, Methodology, Validation, Writing–Review & Editing, Supervision.

Aulia Rizki Firdawanti: Investigation, Data Curation, Formal Analysis, Writing–Review & Editing.

Budi Susetyo: Validation, Writing–Review & Editing, Supervision.

Gerry Alfa Dito: Data Curation, Visualization, Software, Writing–Review & Editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding and Acknowledgments

This research was funded by personal funds and did not receive any specific grant from public, commercial, or not-for-profit funding agencies. The authors would like to express their sincere gratitude to **IPB University**, particularly the *Statistics and Data Science Study Program*, for academic support and a constructive research environment. The authors also acknowledge the Central Statistics Agency (BPS) for providing access to SUSENAS data used in this study.

Data and Code Availability

The SUSENAS microdata analyzed in this study are owned and regulated by the Central Statistics Agency (BPS) and are not publicly available. Access to the data can be requested through official BPS procedures. The code (pre-processing, modeling, and evaluation scripts) used to support the findings of this study is available from the corresponding author upon reasonable request. If required due to ethical/legal constraints, the authors may provide a reproducible workflow using synthetic/aggregated data that preserves confidentiality.

References

- [1] Republic of Indonesia, *Law of the Republic of Indonesia Number 20 of 2003 concerning the National Education System*, Jakarta, 2003.
- [2] Central Statistics Agency, *Indonesian Education Statistics 2023*, Jakarta, 2023.
- [3] Suharti, “Educational inequality and household socioeconomic status in indonesia,” *Journal of Indonesian Economy and Business*, vol. 36, no. 2, pp. 123–140, 2021.
- [4] A. Z. Azzahra, “Addressing distance learning barriers in indonesia amid the covid-19 pandemic,” *Humanities & Social Sciences Communications*, vol. 7, no. 1, pp. 1–8, 2020. DOI: [10.1057/s41599-020-00676-0](https://doi.org/10.1057/s41599-020-00676-0).

- [5] Ministry of National Development Planning/Bappenas, *Human Resource Development within the SDGs Framework*, Jakarta, 2023.
- [6] United Nations Development Programme, *Human Development Report 2019: Beyond Income, Beyond Averages, Beyond Today*, New York, 2019.
- [7] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1263–1284, 2009. DOI: [10.1109/TKDE.2008.239](https://doi.org/10.1109/TKDE.2008.239).
- [8] B. Krawczyk, "Learning from imbalanced data: Open challenges and future directions," *Progress in Artificial Intelligence*, vol. 5, no. 4, pp. 221–232, 2016. DOI: [10.1007/s13748-016-0094-0](https://doi.org/10.1007/s13748-016-0094-0).
- [9] R. e. a. Alizadehsani, "The stratified k-folds cross-validation and class-balancing methods with high-performance ensemble classifiers for breast cancer classification," *Scientific Reports*, vol. 11, no. 1, pp. 1–15, 2021. DOI: [10.1038/s41598-021-93042-6](https://doi.org/10.1038/s41598-021-93042-6).
- [10] N. V. Chawla, A. Lazarevic, L. O. Hall, and K. W. Bowyer, "Smoteboost: Improving prediction of the minority class in boosting," in *Proceedings of the 7th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD)*, Cavtat-Dubrovnik, Croatia, 2003, pp. 107–119.
- [11] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, 1997. DOI: [10.1006/jcss.1997.1504](https://doi.org/10.1006/jcss.1997.1504).
- [12] Y. Sun, M. S. Kamel, A. K. C. Wong, and Y. Wang, "Cost-sensitive boosting for classification of imbalanced data," *Pattern Recognition*, vol. 40, no. 12, pp. 3358–3378, 2007. DOI: [10.1016/j.patcog.2007.04.009](https://doi.org/10.1016/j.patcog.2007.04.009).
- [13] C. Seiffert, T. M. Khoshgoftaar, J. Van Hulse, and A. Napolitano, "Rusboost: A hybrid approach to alleviating class imbalance," *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans*, vol. 40, no. 1, pp. 185–197, 2010. DOI: [10.1109/TSMCA.2009.2029559](https://doi.org/10.1109/TSMCA.2009.2029559).
- [14] R. Gao and Z. Liu, "An improved adaboost algorithm for hyperparameter optimization," *IEEE Access*, vol. 8, pp. 124 283–124 296, 2020. DOI: [10.1109/ACCESS.2020.3006825](https://doi.org/10.1109/ACCESS.2020.3006825).
- [15] D. Reis, F. Melo, and J. P. Pedroso, "Boosting strategies for imbalanced data," *Expert Systems with Applications*, vol. 42, no. 7, pp. 3544–3557, 2015. DOI: [10.1016/j.eswa.2014.12.036](https://doi.org/10.1016/j.eswa.2014.12.036).
- [16] T. Fawcett, "An introduction to roc analysis," *Pattern Recognition Letters*, vol. 27, pp. 861–874, 2006. DOI: [10.1016/j.patrec.2005.10.010](https://doi.org/10.1016/j.patrec.2005.10.010).