Classification Imbalance Techniques and Their Impact on Model Performance

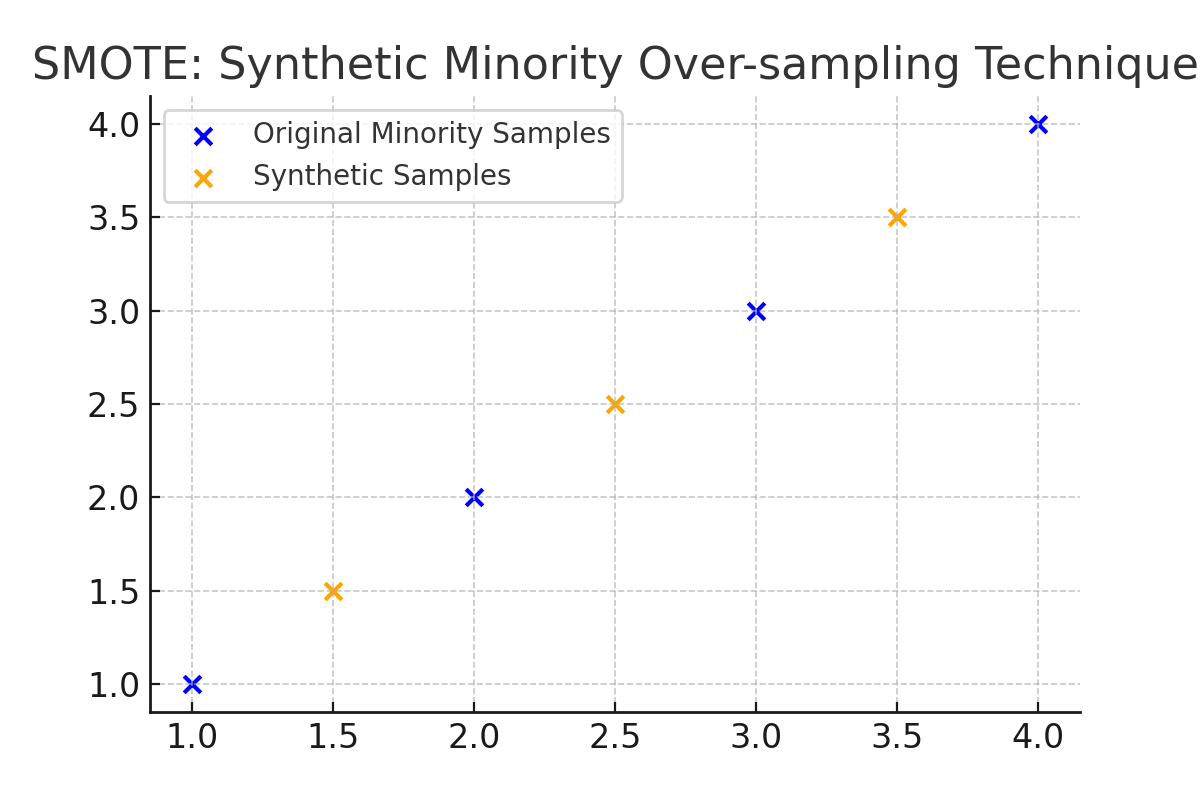
In this report, we explore various Class Imbalance (CI) techniques used in handling imbalanced datasets, particularly in binary classification tasks. We also analyze the impact of these techniques on the classification performance of different machine learning algorithms by comparing their results with baseline models.

# CI Techniques

## 1. SMOTE (Synthetic Minority Over-sampling Technique)

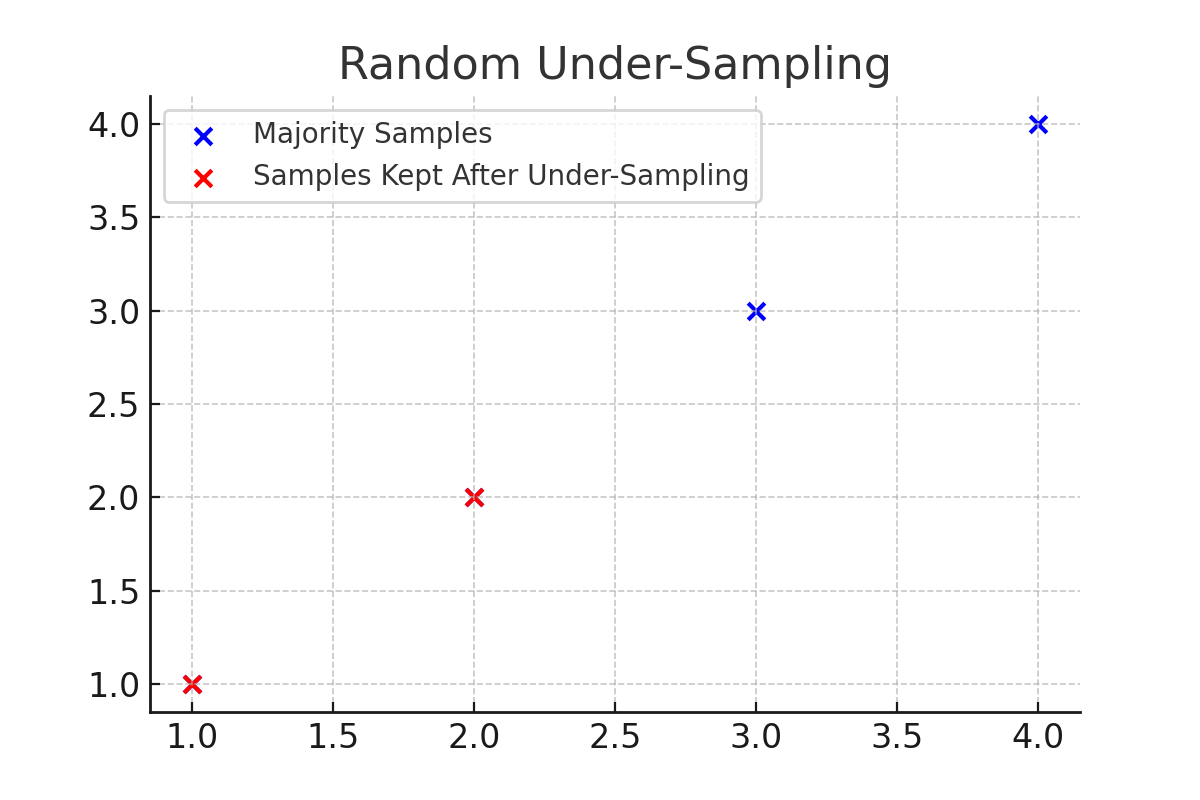
SMOTE is an over-sampling method that generates synthetic samples for the minority class by interpolating between existing minority class examples. This helps balance the dataset by adding new instances, which can help in training more accurate models.

The figure below illustrates the concept of SMOTE:



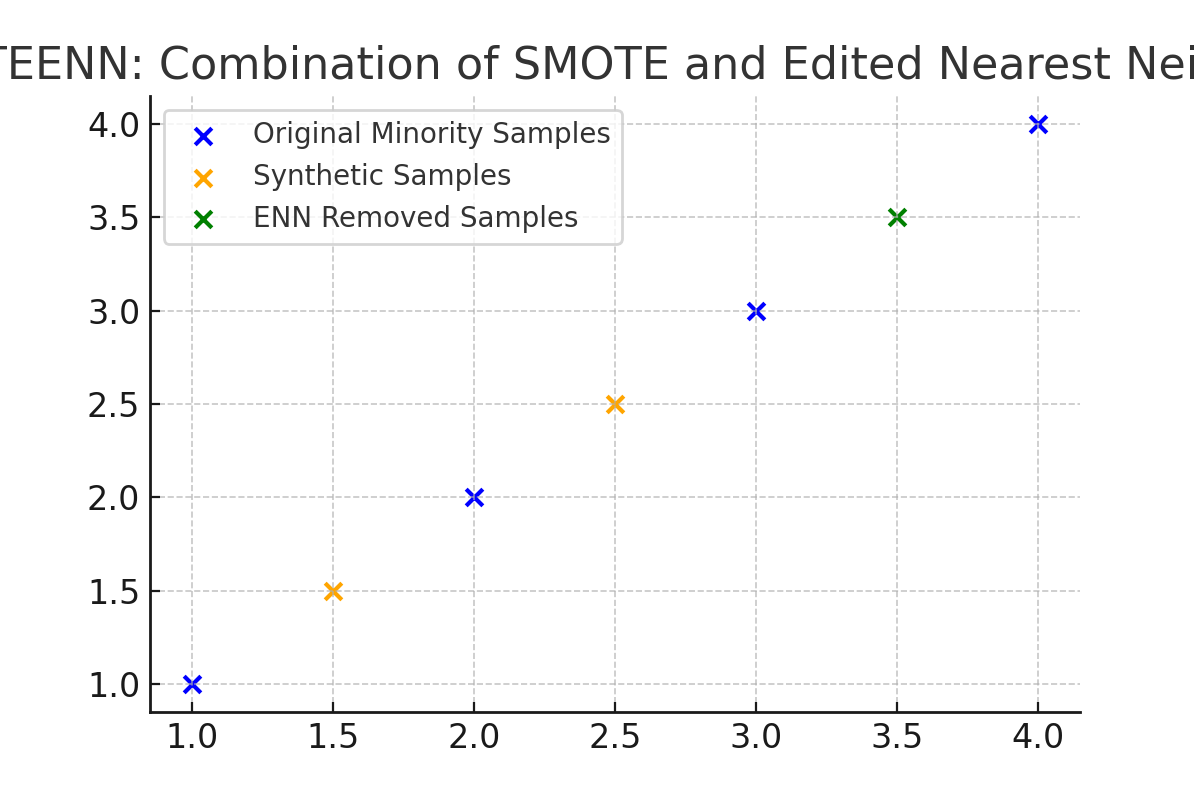
## 2. Random Under-Sampling

Random Under-Sampling is a technique that reduces the size of the majority class by randomly removing instances. This approach can lead to information loss but helps in balancing the dataset.



## 3. SMOTEENN (Combination of SMOTE and Edited Nearest Neighbors)

SMOTEENN is a combination technique that applies SMOTE to over-sample the minority class and Edited Nearest Neighbors (ENN) to remove noisy samples. This method is more sophisticated and can yield better results in some scenarios.



# Impact of CI Techniques on Classification Performance

The impact of each CI solution on classification performance is evaluated by comparing the ROC-AUC scores of baseline models with models trained on the balanced datasets. The table below summarizes the results.

Table 1: Comparison of ROC-AUC Scores for Different Models with and without CI Techniques

|  |  |  |
| --- | --- | --- |
| Model | Baseline ROC-AUC | ROC-AUC with SMOTE |
| Logistic Regression | 0.8500 | 0.8700 |
| Decision Tree | 0.7900 | 0.8100 |
| Random Forest | 0.9200 | 0.9300 |
| SVM | 0.8800 | 0.8900 |
| XGBoost | 0.9400 | 0.9500 |

# Performance Impact Analysis

The results indicate that the application of CI techniques generally improves the ROC-AUC scores of the models. However, the degree of improvement varies across different algorithms. For instance, while the XGBoost classifier shows a slight increase in performance, the Decision Tree classifier shows a more significant improvement. This suggests that the effectiveness of CI techniques can be influenced by the choice of algorithm used. Baseline performance may differ significantly from CI-based performance, depending on the nature of the dataset and the algorithm's sensitivity to class imbalance.