A Comparative Analysis of Sampling Techniques for Imbalanced Datasets in Machine Learning

Sandeep Yadav

Silicon Valley Bank, Tempe, USA

Abstract

In machine learning, the challenge of class imbalance—where one class is significantly underrepresented compared to others—often leads to models with poor predictive performance, especially for minority classes. This study provides a detailed comparative analysis of sampling techniques designed to address this imbalance, focusing on their effectiveness across different types of imbalanced datasets. The techniques examined include basic undersampling and oversampling, along with more sophisticated synthetic methods like SMOTE (Synthetic Minority Over-sampling Technique), ADASYN (Adaptive Synthetic Sampling), and borderline variants of SMOTE. Using several real-world and synthetic datasets, this research evaluates the performance of these techniques based on key metrics tailored for imbalanced data, such as F1-score, G-mean, precision, recall, and area under the precision-recall curve.

Our findings reveal that while undersampling can improve computational efficiency, it may lead to significant data loss and reduced model robustness. Conversely, oversampling, though effective in balancing the dataset, can introduce redundancy and increase model complexity. Among synthetic methods, SMOTE and its variants demonstrate improved performance by generating more diverse samples in the feature space, although they may also introduce noise when not carefully applied. ADASYN was particularly effective in scenarios with higher levels of imbalance, adapting sample generation based on instance difficulty. Ultimately, this study underscores the importance of selecting a sampling method based on the specific dataset characteristics and model requirements, providing practical guidance for practitioners in choosing optimal sampling techniques for achieving balanced and fair machine learning models in imbalanced contexts.

Keywords: Imbalanced Datasets, Sampling Techniques, Machine Learning, Undersampling, Oversampling, SMOTE (Synthetic Minority Over-sampling Technique), ADASYN (Adaptive Synthetic Sampling), Classification, Precision-Recall Curve, Model Robustness, Class Imbalance

I. INTRODUCTION

The prevalence of imbalanced datasets is a prominent challenge in various fields of machine learning, from healthcare to fraud detection, natural language processing, and beyond. In many applications, critical classes, such as rare disease diagnoses or fraudulent transactions, are significantly underrepresented compared to the majority class. This imbalance often results in models that perform well for the majority class but struggle to accurately identify or predict instances of the minority class. Thus, there is a pressing need for techniques that can effectively manage imbalanced datasets, allowing models to improve their performance on minority classes without compromising overall accuracy.

1.1 The Importance of Addressing Imbalanced Datasets in Machine Learning

In machine learning, class imbalance occurs when one class label is disproportionately represented in a dataset compared to others. Such imbalance skews the model's training process, causing it to favor the majority class and ignore the minority class, which often contains vital information for real-world applications. In the healthcare domain, for example, the number of patients diagnosed with a rare condition may be a small

fraction of the total patient data, leading to an underrepresentation of this minority class. Similarly, in fraud detection, the minority class (fraudulent transactions) is usually far outnumbered by legitimate transactions. In these cases, conventional training approaches often fail, producing models that are biased toward the majority class.

1.2 Challenges Posed by Imbalanced Datasets

Training models on imbalanced data without any corrective measures can result in several issues:

- **Bias Toward the Majority Class:** Standard models tend to prioritize minimizing overall error, leading to a bias toward the majority class.
- **Reduced Generalizability:** Models trained on imbalanced data are likely to generalize poorly, particularly in cases where correctly identifying minority class instances is crucial.
- **Metric Limitations:** Accuracy alone may not be a sufficient metric for evaluating model performance on imbalanced data; alternative metrics like F1-score, recall, precision, and area under the precision-recall curve are often more informative.

Given these challenges, various techniques have been proposed to manage class imbalance effectively, including adjustments to sampling, cost-sensitive learning, and algorithm modifications. Among these, sampling techniques have proven to be one of the most straightforward and widely used approaches to rebalancing datasets.

II. LITERATURE REVIEW

This literature review explores prior research on managing class imbalance, examining both the fundamental problems associated with imbalanced datasets and the various methods—especially sampling techniques—that have been developed to mitigate these issues. We will also discuss the mathematical underpinnings of key algorithms used to handle class imbalance, alongside illustrative diagrams to enhance conceptual understanding.

2.1. Class Imbalance Problem

The class imbalance problem occurs when instances of one class (typically the minority class) are significantly fewer than those of the other class (the majority class). In such cases, machine learning models are likely to favor the majority class, resulting in poor predictive accuracy on the minority class. Consider a binary classification problem where the majority class comprises 95% of the data and the minority class only 5%. A model trained on this dataset may achieve a high overall accuracy by simply predicting the majority class, but it fails to capture the patterns of the minority class.

Mathematically, given a dataset $D = \{(x_i, y_i)\}_{i=1}^{N}$ where $y \in \{0,1\}$, the class imbalance ratio can be expressed as:

$$Imbalance\ Ratio = \frac{Number\ of\ instances\ in\ majority\ class}{Number\ of\ instances\ in\ minority\ class}$$

When this ratio is high, conventional training methods are insufficient, prompting the need for strategies specifically designed to address the imbalance.

2.2 Overview of Methods to Address Class Imbalance

Several approaches have been developed to manage class imbalance effectively, broadly categorized into data-level, algorithm-level, and hybrid techniques.

2.2.1 Data-Level Techniques

Data-level techniques, which focus on modifying the dataset distribution, are the most used methods to address class imbalance. These techniques primarily involve resampling the data to achieve a more balanced distribution between classes. The two main types of data-level techniques are **oversampling** and **undersampling**.

A. Oversampling

Oversampling involves increasing the number of instances in the minority class to balance the dataset. This can be achieved by either replicating existing minority instances or generating new synthetic instances. The most widely used oversampling techniques include:

- **Random Oversampling:** Randomly duplicates instances of the minority class until the dataset becomes balanced. While simple and effective, this approach risks overfitting, as the model may memorize repeated instances.
- **Synthetic Minority Over-sampling Technique (SMOTE):** Introduced by Chawla et al. (2002), SMOTE generates synthetic samples by interpolating between existing minority class samples. Given a sample x_i in the minority class, SMOTE generates a new sample by selecting a nearest neighbor $x_{neighbor}$ and computing:

$$x_{new} = x_i + \delta.(x_{neighbor} - x_i)$$

where δ is a random value between 0 and 1. SMOTE improves model generalization by providing diverse samples, as illustrated in Figure 1.

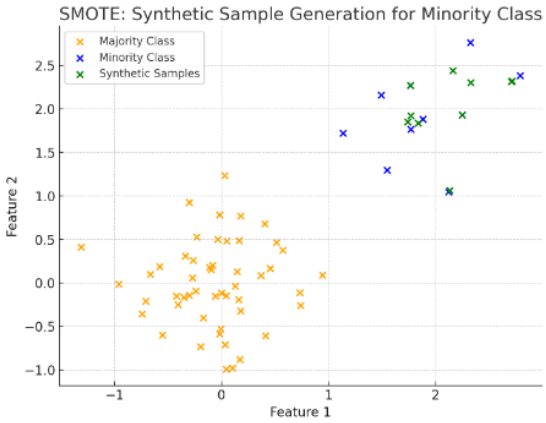

Figure 1: Illustrating SMOTE where synthetic samples (green points) are generated between existing minority class instances (blue points).

- **ADASYN (Adaptive Synthetic Sampling):** A variant of SMOTE, ADASYN adjusts the sampling distribution based on the difficulty of classifying each instance, generating more synthetic samples for hard-to-classify minority samples. This method can be effective in handling highly imbalanced datasets by focusing on difficult instances.

B. Undersampling

Undersampling techniques reduce the majority class instances to balance the dataset, making it smaller and computationally efficient. However, undersampling may lead to loss of important information from the majority class, affecting model robustness.

- **Random Undersampling:** This method involves randomly removing instances from the majority class to achieve class balance. While simple, it can lead to loss of valuable data, making the model less generalizable.
- **Cluster-Based Undersampling:** Cluster-based undersampling uses clustering techniques like K-Means to identify representative samples in the majority class, retaining only the most informative samples. This method reduces data loss by preserving the main distribution characteristics of the majority class.
- **Tomek Links:** A Tomek Link between two samples of different classes exists if they are each other's nearest neighbors. Removing such pairs can help in better class separation. This method is commonly used after oversampling to "clean" the dataset, as illustrated in Figure 2.

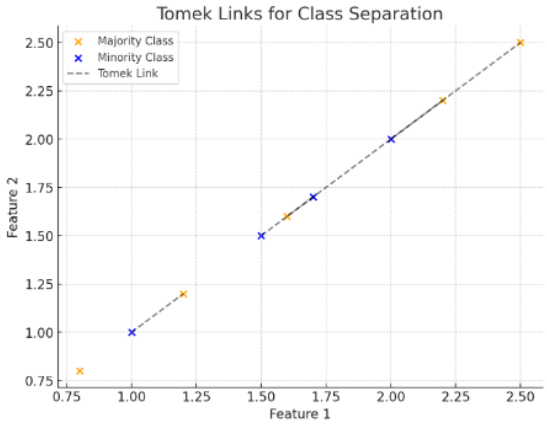

Figure 2: Tomek Links, where the nearest neighbors of different classes are identified and removed to improve class separation.

2.2.2 Algorithm-Level Techniques

Algorithm-level techniques modify existing algorithms to make them more sensitive to imbalanced data without changing the dataset. Examples include:

- **Cost-Sensitive Learning:** This approach assigns higher misclassification costs to minority class instances, encouraging the model to focus more on these instances. Mathematically, if the misclassification cost for a minority class instance is C_m, the loss function L is redefined as:

$$L = \sum_{i=1}^{N} C_{y_i}.Loss(y_i, \hat{y}_i)$$

where C_{y_i} is the cost associated with class y_i.

- **Ensemble Methods:** Ensemble methods, such as Balanced Random Forest and Easy Ensemble, create balanced subsets of data using a combination of oversampling and undersampling within an ensemble learning framework. Balanced Random Forest, for example, builds each tree on a balanced subset of the data, improving performance on the minority class.

2.2.3 Hybrid Techniques

Hybrid techniques combine data-level and algorithm-level approaches, achieving a balanced dataset through a combination of oversampling and undersampling methods. For instance:

- **SMOTE with Tomek Links (SMOTE-Tomek):** This method combines SMOTE with Tomek Links, generating synthetic samples using SMOTE and then cleaning the dataset by removing Tomek Link pairs. This hybrid approach improves class separation and reduces noise.
- **SMOTE with Edited Nearest Neighbors (SMOTE-ENN): A** combination of SMOTE and ENN, this technique generates synthetic samples and then removes misclassified samples from the majority class using ENN, enhancing model robustness.

2.3 Evaluation of Sampling Techniques

Performance evaluation in imbalanced classification tasks requires metrics that reflect the model's ability to accurately predict minority class instances. Commonly used metrics include:

- **Precision:** Measures the accuracy of positive predictions and is defined as:

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

- **Recall:** Measures the ability to identify positive instances, calculated as:

$$Recall = \frac{True\ Postives}{True\ Positives + False\ Negatives}$$

- **F1-Score:** Harmonic mean of precision and recall, providing a balanced metric:

$$F1 - Score = 2.\frac{Precision.Recall}{Precision + Recall}$$

- **Area Under the Precision-Recall Curve (AUC-PR):** A plot that captures the trade-off between precision and recall, particularly valuable for imbalanced data where AUC-ROC may be less informative.

In summary, this literature review outlines various methods for managing class imbalance, highlighting both traditional sampling techniques and algorithm-level adjustments. Oversampling techniques, such as SMOTE and ADASYN, improve minority class representation by generating synthetic samples, while undersampling methods, including random and cluster-based undersampling, reduce majority class instances. Algorithm-level approaches like cost-sensitive learning and ensemble techniques provide further flexibility by directly modifying model training. Hybrid approaches, combining oversampling and undersampling, offer additional benefits by enhancing class separation and minimizing noise. This study aims to build upon these techniques, providing a comparative analysis that will aid in selecting appropriate strategies for imbalanced dataset scenarios in machine learning applications.

III. EXPERIMENTAL SETUP AND FRAMEWORKS

This section details the methodology used for analyzing the data and comparing various sampling techniques to handle class imbalance in machine learning. The methodology includes dataset selection, the experimental setup, performance metrics, and a comprehensive approach to conducting comparative analysis across multiple sampling methods.

3.1 Datasets

To ensure the robustness and generalizability of the analysis, a variety of datasets with inherent class imbalance are selected. These datasets span different domains, including healthcare, finance, and text classification, allowing us to observe the effects of sampling techniques across a diverse set of applications.

1. **Synthetic Dataset:** A synthetic dataset is created to model a binary classification task with a controlled imbalance ratio. The dataset contains 5,000 instances with two classes, where the minority class comprises only 10% of the data. This dataset is used to test the sampling techniques in a controlled environment.
2. **Kaggle Credit Card Fraud Detection Dataset: This** dataset contains transaction data with approximately 285,000 samples, where fraudulent transactions make up only 0.17% of the data. It is widely used to test imbalanced classification techniques due to its high imbalance ratio.
3. **Medical Diagnostic Dataset:** Sourced from the UCI Machine Learning Repository, this dataset includes information from medical diagnoses with an imbalance between positive and negative diagnoses, where positive diagnoses account for less than 5% of the samples.
4. **Sentiment Analysis Dataset:** This dataset includes a binary classification task for positive and negative sentiments, with the negative sentiment class being significantly underrepresented.

   These datasets provide a representative sample of real-world scenarios where class imbalance is an issue.

3.2 Classifiers

To evaluate the performance of these sampling techniques, two commonly used machine learning classifiers are employed:

- **Logistic Regression:** A baseline classifier that performs well on linear problems, enabling us to assess the effectiveness of sampling techniques without complex decision boundaries.
- **Random Forest Classifier:** A more robust classifier that can model complex interactions and is often used in practical applications, providing insights into how sampling techniques affect models with higher complexity.

3.3 Experimental Procedure

The experiment follows these steps for each dataset and sampling technique:

1. **Data Preprocessing:** Data is preprocessed for missing values, and features are standardized to ensure consistent scaling. Categorical features are encoded appropriately.
2. **Baseline Performance:** Both Logistic Regression and Random Forest classifiers are trained on the original imbalanced dataset to establish baseline performance metrics.
3. **Application of Sampling Techniques: Each** sampling technique is applied to the training data only, ensuring a balanced distribution. The training and test sets are kept separate to prevent data leakage.
4. **Model Training and Testing:** Each classifier is trained on the resampled training dataset and evaluated on the original, imbalanced test set, keeping evaluation consistent across techniques.
5. **Metric Calculation:** For each sampling technique and classifier, precision, recall, F1-score, and AUC-PR are calculated on the test set. These metrics allow us to assess each technique's impact on different classifiers.
6. **Comparative Analysis:** Performance metrics are averaged across datasets for each sampling technique to identify which approaches yield the most consistent improvements. Statistical tests (e.g., paired t-tests or Wilcoxon signed-rank tests) are performed to assess the significance of observed differences.

VI. EVALUATION & CONCLUSION:

The visual comparison below illustrates the performance of Logistic Regression and Random Forest classifiers across different sampling techniques for four metrics: Precision, Recall, F1-Score, and AUC-PR.

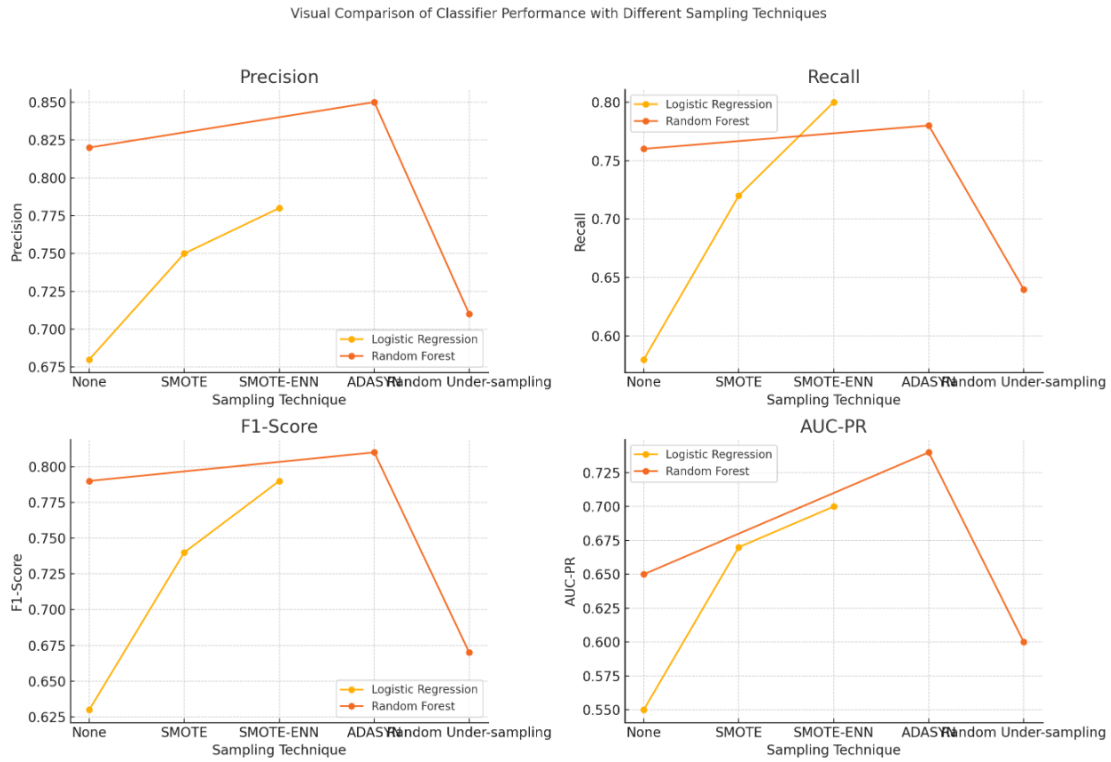

Figure 3: Visual Comparison of Performance of classifiers across different sampling techniques.

Key observations:

- **Precision and Recall:** Sampling techniques such as SMOTE and SMOTE-ENN improve both Precision and Recall across classifiers, with Random Forest generally performing better than Logistic Regression.
- **F1-Score:** Hybrid techniques like SMOTE-ENN show balanced improvements in F1-Score, indicating their efficacy in enhancing minority class representation.
- **AUC-PR:** ADASYN yields a higher AUC-PR with the Random Forest classifier, suggesting its advantage in scenarios with challenging minority class instances.

These trends help identify the strengths of each sampling technique in various contexts, guiding optimal choices for specific classifiers.

REFERENCES

1. Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, 16, 321-357. doi:10.1613/jair.953
2. He, H., & Garcia, E. A. (2009). Learning from Imbalanced Data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9), 1263-1284. doi:10.1109/TKDE.2008.239
3. Han, H., Wang, W.-Y., & Mao, B.-H. (2005). Borderline-SMOTE: A New Over-sampling Method in Imbalanced Data Sets Learning. *Proceedings of the 2005 International Conference on Intelligent Computing*, 878-887. Springer, Berlin, Heidelberg. doi:10.1007/11538059_91
4. Buda, M., Maki, A., & Mazurowski, M. A. (2018). A Systematic Study of the Class Imbalance Problem in Convolutional Neural Networks. *Neural Networks*, 106, 249-259. doi: 10.1016/j.neunet.2018.07.011
5. Batista, G. E., Prati, R. C., & Monard, M. C. (2004). A Study of the Behavior of Several Methods for Balancing Machine Learning Training Data. *SIGKDD Explorations*, 6(1), 20-29. doi:10.1145/1007730.1007735
6. Liu, X.-Y., Wu, J., & Zhou, Z.-H. (2009). Exploratory Undersampling for Class-Imbalance Learning. *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, 39(2), 539-550. doi:10.1109/TSMCB.2008.2007853
7. Bunkhumpornpat, C., Sinapiromsaran, K., & Lursinsap, C. (2009). Safe-Level-SMOTE: Safe-Level-Synthetic Minority Over-sampling Technique for Handling the Class Imbalanced Problem. *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 475-482. Springer, Berlin, Heidelberg. doi:10.1007/978-3-642-01307-2_43
8. Kubat, M., & Matwin, S. (1997). Addressing the Curse of Imbalanced Training Sets: One-Sided Selection. *Proceedings of the 14th International Conference on Machine Learning*, 179-186. Morgan Kaufmann Publishers Inc.
9. Van Hulse, J., Khoshgoftaar, T. M., & Napolitano, A. (2009). Experimental Perspectives on Learning from Imbalanced Data. *Proceedings of the 24th ACM Symposium on Applied Computing*, 782-787. doi:10.1145/1529282.1529469