Jurnal Teknik Industri: Jurnal Hasil Penelitian dan Karya Ilmiah dalam Bidang Teknik Industri

Vol. 11, No 1, 2025

Overcoming Data Imbalance in Risk Management: A Comparative Study of Sampling Methods

Arya Wijna Astungkara[1], Achmad Pratama Rifai[2]
[1,2]Department of Industrial Engineering, Faculty of Engineering, Universitas Gadjah Mada
Bulaksumur, Caturtunggal, Kec. Depok, Kabupaten Sleman, Daerah Istimewa Yogyakarta
Email: aryawijna@mail.ugm.ac.id, achmad.p.rifai@ugm.ac.id

ABSTRACT

Data imbalance presents a significant and persistent challenge in risk management, particularly in classification tasks where critical events—such as loan defaults, employee attrition, or corporate bankruptcy—occur far less frequently than normal cases. This paper presents a comparative analysis of eight data-level sampling methods—Random Undersampling (RUS), Random Oversampling (ROS), Edited Nearest Neighbour (ENN), One-Sided Selection (OSS), Synthetic Minority Oversampling Technique (SMOTE), Adaptive Synthetic Sampling (ADASYN), and the hybrid methods SMOTEENN and SMOTETomek. These techniques are evaluated across three distinct and real-world imbalanced datasets: Taiwanese Bankruptcy Prediction, IBM HR Analytics Employee Attrition, and a large-scale Loan Prediction dataset. Using a suite of eight machine learning classifiers, the study assesses performance through two complementary metrics: the F1 Score, which balances precision and recall, and the Negative Predictive Value (NPV), which measures the reliability of pessimistic predictions. The results reveal a critical trade-off between maximising minority class detection and minimising false negatives. While ENN demonstrates strong performance in high-dimensional and severely imbalanced contexts, and SMOTE-based methods excel in large-scale datasets with moderate imbalance, RUS consistently delivers the highest NPV. This highlights its unparalleled effectiveness in reducing costly false negatives, supporting conservative, risk-averse decision-making. The findings underscore that selecting a sampling strategy is not merely a technical choice but a strategic one, which must be aligned with specific dataset characteristics and fundamental risk management objectives.

Keywords: *Risk Management, Classification, Sampling, Machine Learning, Data Imbalance*

Introduction

Risk management has evolved from siloed, reactive functions in recent decades into integrated, proactive frameworks like Enterprise Risk Management (ERM). ERM unifies diverse risk categories—strategic, operational, financial, and compliance—and prioritizes continuous surveillance and aggregate risk assessment to bolster organizational resilience [1], [2]. This evolution has been accelerated by the proliferation of digital technologies, which, while creating efficiencies, have also introduced novel risks such as IT failure, data breaches, and sophisticated cyberattacks. In response, organizations have embedded predictive analytics and machine learning (ML) into their risk governance frameworks, enabling real-time monitoring and rapid response capabilities essential for navigating complex digital landscapes. [3].

Machine learning models, intense learning and ensemble methods have demonstrated superior performance over traditional statistical models like Value-at-Risk (VaR) in domains such as credit scoring and fraud detection. [4], [5]. Their ability to model non-linear relationships and adapt to dynamic data environments makes them invaluable for predicting extreme events and identifying subtle anomalies in vast datasets. These models can be continuously updated, making them well-suited for the high-velocity information flows that characterize modern financial and operational environments. [6], [7].

Despite these advancements, a fundamental and pervasive challenge threatens the reliability of ML-driven risk systems: data imbalance. In most risk management contexts, the events of most significant interest—loan defaults, fraudulent transactions, corporate bankruptcies, critical employee attrition—are, by nature, rare. These minority class instances are often outnumbered by standard, non-event instances by orders of magnitude [8], [9]. This skewed distribution poses a significant problem for most standard classification algorithms, which are optimised for overall accuracy. A model trained on a dataset where 99% of instances are non-fraudulent can achieve 99% accuracy simply by predicting "no

fraud" every time, rendering it useless for its intended purpose. This phenomenon makes accuracy a dangerously misleading metric in imbalanced scenarios [10], [11].

The consequences of this bias are severe. A model that fails to identify potential defaulters (false negatives) can lead to substantial financial losses in credit risk analysis. [9]. Similarly, in medical diagnostics, an analogous problem, failing to detect a rare disease, can have life-threatening consequences. Therefore, the challenge of data imbalance is not merely a technical nuisance but a critical barrier to the effective deployment of ML in high-stakes decision-making. The imbalance itself is often not a data flaw but an intrinsic feature of the domain; for instance, a low default rate in a loan portfolio is the *goal* of successful credit risk management, meaning the data will inherently be imbalanced. To address this, performance metrics sensitive to the minority class, such as precision, recall, and the F1 Score, are essential for robust model evaluation. Furthermore, the Negative Predictive Value (NPV), which measures the probability that a pessimistic prediction is truly pessimistic, is exceptionally valuable in risk contexts where confidently clearing an entity of risk (e.g., approving a loan, certifying a system as secure) is a primary objective. [12].

Researchers have developed various specialized techniques to counteract the bias induced by imbalanced data. These methods can be broadly categorized into three families: data-level approaches, algorithm-level approaches, and hybrid methods, as illustrated in Figure 2.

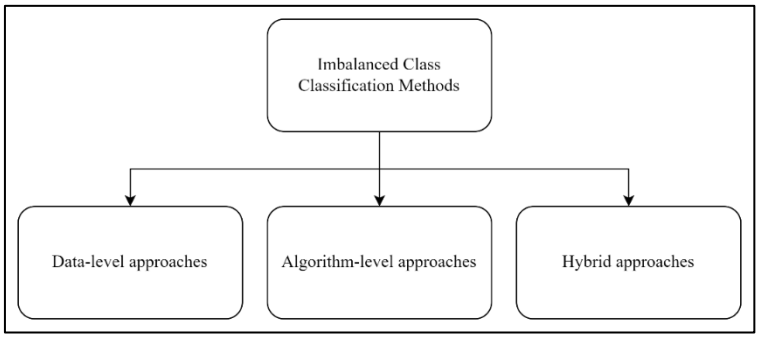

Figure 1. Imbalanced classification methods taxonomy

Data-level approaches address class imbalance by manipulating the training data before it is fed into the learning algorithm. This can be achieved through oversampling methods, which increase the number of minority class instances, such as the widely used Synthetic Minority Oversampling Technique (SMOTE)or undersampling methods, which reduce the number of majority class instances. Some techniques combine both oversampling and undersampling to maintain a balance while minimizing the risks of overfitting or information loss. These methods aim to create a more balanced dataset that allows classifiers to learn the characteristics of both classes better. [13].

Algorithm-level approaches tackle imbalance by modifying the learning algorithm itself. One common strategy is cost-sensitive learning, where misclassifications of minority class instances are penalized more heavily than those of the majority class. Other techniques include adjusting decision thresholds or designing specialized algorithms inherently more sensitive to class imbalance, such as ensemble methods or one-class classifiers. These methods focus on making the model more responsive to the minority class without changing the underlying data distribution [14].

Hybrid approaches combine data-level and algorithm-level strategies to leverage the advantages of both. For instance, a hybrid method may apply SMOTE to generate synthetic minority samples and then use an ensemble classifier like boosting to enhance model performance. These approaches are more robust and effective, particularly in complex real-world applications where data imbalance and model sensitivity must be addressed simultaneously. Research has shown that hybrid methods often outperform pure data-level or algorithm-level techniques in various domains such as credit scoring and medical diagnosis [15], [16].

Despite the widespread use of sampling methods, there is a lack of systematic, comparative studies that evaluate their performance across diverse risk management domains using metrics that reflect real-world decision-making trade-offs. This paper aims to fill this critical gap. The primary contribution of this research is a rigorous, empirical comparison of eight prominent sampling methods applied to three distinct, real-world risk datasets: corporate bankruptcy prediction, human resource attrition, and consumer loan default.

The study evaluates these methods using a suite of eight machine learning classifiers, providing a comprehensive view of how sampler-classifier interactions affect performance. Crucially, this analysis

is conducted through a dual-metric framework. The F1 Score assesses the model's ability to balance precision and recall for the minority class, a common goal in predictive modelling. Simultaneously, the Negative Predictive Value (NPV) is used to evaluate the model's reliability in identifying true negative cases, a paramount concern in conservative, risk-averse environments. By analysing performance through both lenses, this paper provides a nuanced understanding of the strategic trade-offs in selecting a sampling method, offering practical guidance for researchers and risk management professionals.

Research Methods

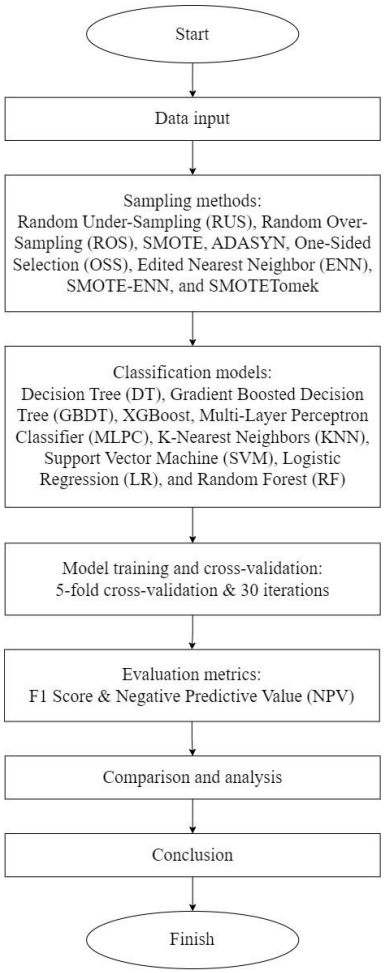

Figure 2. Research flow

The methodology of this study follows a structured, multi-stage process designed to systematically evaluate the impact of various data sampling techniques on machine learning performance in risk management contexts. The research begins with collecting and describing three distinct, real-world imbalanced datasets. The second stage involves a comprehensive data preprocessing pipeline, which includes standard cleaning and scaling, followed by applying eight different sampling methods (undersampling, oversampling, and hybrid) to address the core issue of class imbalance. In the third stage, the original and resampled datasets train a diverse suite of eight machine learning classifiers, covering a range of algorithmic approaches. Finally, the performance of each classifier-sampler combination is rigorously evaluated using two key metrics appropriate for imbalanced data: the F1 Score and the Negative Predictive Value (NPV). This systematic process, illustrated in Figure 2, ensures a rigorous and comprehensive comparison of the sampling methods.

Dataset Description

In this research, we utilize three disparate data sets, each appropriate to different areas in risk management, namely human resources, finance, and company financial well-being. As represented in Table 1, the data sets differ in sample size, attribute number, and imbalance ratio (IR).

The IBM HR Analytics Employee Attrition & Performance dataset [17], comprises 1,470 instances and 32 attributes describing employee demographics and job performance. Featuring an imbalance ratio (IR) measure of 6.2, the problem is moderately imbalanced and well suited to predicting employee voluntary attrition.

The second dataset, the Loan Prediction Based on Customer Behavior dataset [18], is composed of 252,000 samples and 11 attributes. Some primary qualities are income, loan size, employment type, credit history, and credit score. With an imbalance rate equal to 8.1, the dataset is one where most of the applications get accepted. Hence, it is appropriate to construct credit risk models that identify potential defaulting borrowers.

The third dataset, the Taiwanese Bankruptcy Prediction dataset [19], contains 6,819 records and 95 attributes including extensive financial ratios and operating metrics of Taiwanese firms. It is extensively utilized in both financial distress prediction and bankruptcy prediction tasks. With an incredibly high imbalance ratio value of 31, the dataset portrays an acute imbalance in which bankrupt companies constitute a small minority.

Table 1. Dataset description

No.	Dataset	*Attributes*	*Sample*	*IR Rate*
1.	IBM HR Analytics Employee Attrition & Performance	32	1470	6.2
2.	Loan Prediction	11	252000	8.1
3.	Taiwanese Bankruptcy Prediction	95	6819	31

These databases collectively cover multiple application domains—HR turnover, loan-risk assessment, and company bankruptcy prediction—differing in degree and severity of imbalance and complexity. Table 1 outlines their size, attributes, and classes. They form a robust testbed for comparing and verifying sampling methods in risk assessment based on imbalanced data.

Machine Learning Classification Algorithms

Decision tree Decision Tree (DT) is one Supervised Machine Learning method to solve regression and classification problems by repeatedly dividing data based on some parameter [20]. It is modeled as a structure using trees, where a node denotes testing on an attribute, an edge denotes the testing result, and a leaf node denotes the class name. To divide, the most used criteria are “gini” for Gini Impurity and “entropy” for information gain, which can be written as [21].

$$Entropy: H(x) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i) \quad (1)$$

$$\text{Gini}(E) = 1 - \sum_{i=1}^{c} p_i^2 \quad (2)$$

A decision tree is a series of if-then-else decision rules that split the data into branches, allowing the analyst to conclude about the target value. The deepest branches on the tree often conclude the target variable, making them easy to interpret but susceptible to overfitting if not pruned appropriately. Decision Trees can handle numerical and categorical data and can intuitively model complex interactions between different variables without requiring transformation of features.

Gradient-boosted decision trees Gradient-boosted decision Trees (GBDT) are an ensemble methodology that uses the predictions of multiple base estimators constructed sequentially to enhance accuracy. [22]. Each new tree seeks to rectify the mistakes made by the earlier trees. GBDT works exceptionally well in both regression and classification problems.

In GBDT, every subsequent tree is constructed to rectify the mistakes committed by the trees already built. Learning is conducted by fitting the new predictor to the residual errors committed by the current predictors and then including it in the ensemble, i.e., improving the model where it exhibited inferior performance in earlier rounds. GBDT is exceedingly flexible and can optimize on multiple loss functions and offers multiple hyperparameter tuning options that render the model robust to even non-standard data.

K-nearest neighbors K-Nearest Neighbors (KNN) is an easy, instance-based learning algorithm that makes the prediction based on the majority vote among the k nearest neighbours [23]KNN is both non-parametric and lazy. It does not make any assumptions regarding the data distribution or train an

explicit model. Instead, it stores all the training instances and classifies runtime by comparing the new example to those stored earlier.

KNN is very simple to implement in its simplest form and yet can carry out sophisticated classification tasks. It is a form of lazy learning as it does not possess any specific training phase and uses all accessible data for training to classify any new data instance or point.

Logistic Regression (LR) is one type of linear model applied to binary classification problems. It estimates the probability that any input is in any specific class using the logistic function to transform calculated values to probabilities. [24], which is also called the mathematically defined sigmoid function in Eq. 3.

$$g(z) = \frac{1}{1 + \exp(-z)} \tag{3}$$

Logistic Regression is heavily utilized in binary classification tasks, including spam filtering, medical diagnosis, and credit scoring, owing to its simplicity and interpretability. Logistic Regression is effective for extensive data but does not handle complex relationships, relying on the supposition that there is a linear relationship between log-odds of the target and input features.

Multilayer perceptron Multilayer Perceptron (MLP) is a class of artificial neural networks that consists of multiple layers of nodes, each fully connected to the next one. It is a fundamental architecture for various machine learning tasks, including classification and regression. The structure of an MLP includes an input layer, one or more hidden layers, and an output layer. The input layer consists of input nodes, with the number of input nodes corresponding to the number of features in the dataset. Hidden layers, which lie between the input and output layers, contain nodes that apply non-linear activation functions to weighted sums of their inputs. These activation functions can include ReLU (Rectified Linear Unit), sigmoid, or tanh functions. [25].

Random Forest (RF) is an ensemble learning algorithm which builds many decision trees at training time and returns the mode class (for classification) or the mean prediction (for regression) over individual trees. Random Forest generalises and minimises overfitting by introducing randomness in feature choice and data sampling, rendering it a strong and commonly practised technique. [26]. Thus, the model based on RF and multiple trees is generally more accurate than that based on one single decision tree. [27].

Support vector machines (SVMs) are supervised learning algorithms that locate the best hyperplanes that maximise margins among classes. SVM can deal with linear and non-linear data by applying kernel functions to map data into higher-dimensional feature spaces where linear separators can be discovered. SVM can deal well in high-dimensional spaces and can handle non-linear classification[28].

Extreme Gradient Boosting (XGBoost) is a sophisticated form of gradient tree boosting initially proposed by [29]. It's a method that employs a series of weak classifiers to construct a robust classifier. The method begins with a base learner and trains the strong learner iteratively [30]. Gradient boosting and XGBoost share the same fundamental principles but differ mainly in their implementation. XGBoost enhances performance through effective regularisation techniques that manage tree complexity [30].

Sampling Method

Random undersampling (RUS) is a technique for preprocessing data, especially when reducing instances in the majority class to obtain a balanced class distribution. Here, the majority class is randomly chosen and removed by taking out subsets of majority class examples until the needed balance is attained. [31].

Edited nearest neighbor (ENN) was introduced by [32] ENN is a method for reducing dataset noise and improving the decision boundaries for nearest neighbour classification. It works by examining each instance in the training set and removing those whose class label differs from most of its k-nearest neighbours. This method eliminates noisy instances and outliers, leading to more robust classifiers.

ENN is especially useful for addressing unbalanced datasets by filtering out noisy instances from the majority class and consequently equalising the class distribution. It simplifies the task for classifiers to pinpoint minority class instances more easily. The process improves the aggregate value of the training data, resulting in improved classifier performance based on measures such as F-value as well as AUC. [33].

One-sided selection One-Sided Selection (OSS) is an ensemble-based technique that integrates Tomek links, and the Condensed Nearest Neighbour (CNN) rule introduced by [34], to successfully decrease the samples in an unbalanced dataset without compromising the integrity of the minority group. The OSS method removes the majority class samples that are Tomek links, which are defined as pairs of nearest enemy samples of different classes and cannot be separated without error by the nearest neighbour

rule. This step is followed by applying the CNN rule, which aims to ensure that all the remaining samples are near the decision boundary, further refining the dataset by focusing on crucial instances for classification.

Random oversampling (ROS) is one solution to handle class-biased data. It involves creating multiple copies of the minority class samples, called naïve random oversampling (ROS). New samples are formed by randomly taking samples from the initial data with replacement. The process is iterated several times to give more representation to the minority class.

SMOTE: This method, proposed by [35], offers a more effective alternative to naïve random oversampling. Instead of replicating samples, this technique generates new synthetic examples for the minority class based on feature space similarities. The process involves identifying the K-nearest neighbors of each minority sample, randomly selecting one of these neighbors, and then performing linear interpolations to create a new synthetic sample.

ADASYN, an enhanced adaptive version of SMOTE, was proposed by [36]. Unlike SMOTE, which generates the same number of synthetic samples for each original minority sample, ADASYN uses a density distribution to determine the number of synthetic samples produced for each minority sample. This algorithm adaptively weights the minority samples in proportion to their difficulty in being classified. More difficult-to-classify samples are weighted higher, generating more synthetic samples for these difficult instances than easier ones.

SMOTEENN (Synthetic Minority Over-sampling Technique - Edited Nearest Neighbours) is a hybrid method developed by [37] to handle class imbalance in data. In this technique, the SMOTE over-sampling technique is coupled with the ENN under-sampling technique. To begin with, SMOTE-ENN applies SMOTE to create synthetic samples belonging to the minority class, which helps balance class distribution. Afterwards, ENN was applied to preprocess the data by eliminating synthetic samples misclassified by their nearest neighbours. Through this hybrid technique, the count of minority-class samples is increased, and their quality and relevance are also assured, thus the overall classification is enhanced.

Experimental Setup and Reproducibility

All experiments were executed in a Python programming environment using open-source libraries. Data manipulation was performed with pandas, and numerical operations using NumPy. Sampling methods were applied via the imbalanced-learn library. All machine-learning classifiers—each initialized with random_state=None for reproducibility—were implemented using scikit-learn. All machine learning classifiers were implemented using the scikit-learn library, except XGBoost, for which its dedicated library was used. The experiments were run on a system equipped with an Intel Core i5-10400F CPU, 16GB of 3200MHz RAM, and an NVIDIA GeForce GTX 1660 SUPER GPU.

Performance Measurement

The choice of evaluation metrics is critical in imbalanced classification, as standard accuracy can be highly misleading. To ensure robust and stable performance evaluation, all reported metrics in this study were averaged from a 5-fold cross-validation procedure, repeated over 30 iterations. This study employs two metrics that provide a more nuanced assessment of model performance: the F1 Score and the Negative Predictive Value (NPV).

The confusion matrix is a commonly used tool to describe the performance of a classification model on a dataset with known true values. It is a table that presents four different combinations of predicted and actual values: True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN), as shown in Figure 3.

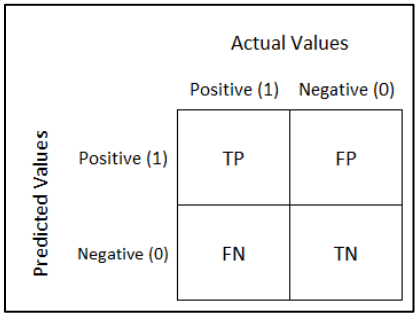

Figure 3. Confusion matrix

The F1 Score is a statistical measure used to evaluate the performance of a binary classification model. The harmonic means of precision and recall provides a single metric that balances sensitivity and positive predictive accuracy. The formula for the F1 Score is as follows:

$$F1\ Score = \frac{2 \times Sensitivity\ \times Precision}{Sensitivity + Precision}, \quad (1)$$

Where:

Precision (Positive Predictive Value) is the proportion of correct identifications, calculated as $\frac{TP}{TP+FP}$.

Recall (Sensitivity) measures the ability of the model to find all the relevant cases within a dataset, calculated as $\frac{TP}{TP+FN}$.

Negative Predictive Value (NPV) is an important performance metric in classification tasks that indicates the probability that subjects with a negative result are truly negative. The formula defines it:

$$NPV\ = \frac{TN}{TN + FN}. \quad ((2)$$

Results and Discussion

The experimental results provide a multifaceted view of how sampling methods interact with different classifiers and datasets. The analysis is presented in two parts, focusing first on the F1 Score to assess balanced predictive accuracy for the minority class, and second on the Negative Predictive Value (NPV) to evaluate performance from a risk-aversion perspective. The full results are detailed in Table 2 and Table3.

Analysis of F1 Score Performance

The F1 score, the harmonic mean of precision and recall, is a critical indicator of a model's effectiveness in detecting minority-class instances. As shown in Tables 2, the optimal sampling strategy depends on dataset characteristics such as dimensionality, class-imbalance ratio, and total sample size. In each table, the boldface value in every row identifies the highest F1 score achieved by the corresponding classifier, thereby highlighting the most effective sampling method for that scenario.

Table 2. F_1 of all algorithms in the collected datasets

Dataset	Classifier	RUS	ENN	OSS	ROS	SMOTE	ADASYN	SMOTEENN	SMOTETomek
Taiwan-Bankruptcy	DT	0.208	**0.299**	0.252	0.263	0.264	0.256	0.283	0.268
	GBDT	0.265	**0.363**	0.287	0.339	0.322	0.309	0.312	0.321
	KNN	0.236	**0.381**	0.243	0.283	0.250	0.245	0.235	0.250
	LR	0.249	**0.388**	0.259	0.269	0.272	0.260	0.257	0.273
	MLPC	0.240	**0.379**	0.274	0.292	0.287	0.293	0.334	0.293
	RF	0.272	0.384	0.250	0.302	**0.388**	0.376	0.365	0.385
	SVM	0.226	**0.372**	0.219	0.263	0.263	0.251	0.254	0.263
	XGBoost	0.263	0.352	0.257	0.330	0.344	0.335	**0.375**	0.344
HR-Employee-Attrition	DT	0.347	0.362	0.367	0.329	0.373	0.376	**0.386**	0.378
	GBDT	0.456	0.505	0.454	**0.520**	0.482	0.478	0.489	0.478
	KNN	**0.396**	0.328	0.240	0.383	0.378	0.372	0.352	0.378

	LR	0.470	**0.517**	0.471	0.490	0.492	0.483	0.419	0.492
	MLPC	0.448	**0.512**	0.481	0.466	0.459	0.460	0.463	0.462
	RF	0.465	0.400	0.301	0.344	0.402	0.405	**0.495**	0.403
	SVM	0.467	**0.504**	0.454	0.488	0.488	0.482	0.419	0.488
	XGBoost	0.456	0.495	0.465	0.460	0.474	0.470	**0.506**	0.474
Loan-Prediction	DT	0.616	0.618	0.562	0.623	**0.628**	0.626	0.555	0.628
	GBDT	0.274	0.018	0.003	**0.276**	0.265	0.190	0.212	0.265
	KNN	0.518	**0.599**	0.021	0.572	0.573	0.579	0.598	0.573
	LR	0.217	0.000	0.000	0.217	**0.218**	0.180	0.171	0.218
	MLPC	0.338	0.249	0.083	0.368	0.372	0.335	0.345	**0.374**
	RF	0.579	0.578	0.522	0.587	0.601	**0.607**	0.532	0.601
	SVM	0.217	0.000	0.000	0.217	**0.218**	0.180	0.167	0.218
	XGBoost	0.541	0.523	0.326	0.574	**0.577**	0.522	0.444	0.577

According to the dataset's characteristics and Table 2, there are some observations about the performance of sampling methods across the three datasets.

For Taiwan-Bankruptcy, the Edited Nearest Neighbour (ENN) sampling approach generates the highest scores in terms of F1 values for most classifiers. ENN surpasses other methods when employed by DT (0.299), GBDT (0.363), KNN (0.381), LR (0.388), MLPC (0.379), and SVM (0.372). RF scores best using SMOTE (0.388). These observations depict ENN as highly effective on highly imbalanced, high-dimensional data.

For the HR-Employee-Attrition data, performance is spread more across sampling methods. SMOTEENN yields top F1 values for DT (0.386), RF (0.495), and XGBoost (0.506). RUS is performing high in the case of KNN (0.396), whereas ENN yields top values for LR (0.517), MLPC (0.512), and SVM (0.504). ROS is high for GBDT (0.520). This trend indicates that no technique outperforms all classifiers.

Performance of the sampling methods differs for classifiers in the case of the Loan-Prediction dataset. ENN yields the highest accuracy using KNN (0.599), whereas ROS yields the highest accuracy using GBDT (0.520). SMOTE yields the highest accuracy using DT (0.628), LR (0.218), SVM (0.218), and XGBoost (0.577). ADASYN is the most accurate using RF (0.607), whereas SMOTETomek is the most accurate using MLPC (0.374). These observations prove that synthetic sampling methods like SMOTE and SMOTETomek perform well on moderately imbalanced, large-scale financial data.

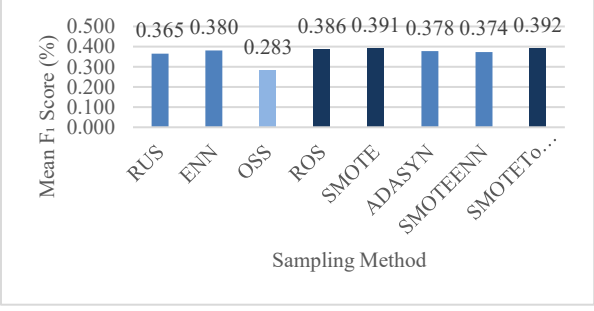

Figure 4. Mean value of F1 on all algorithms

As illustrated by Figure 4, SMOTETomek has the best average F1 score, just ahead of SMOTE and ROS. This indicates that both oversampling and hybrid strategies perform best to enhance classifier accuracy on skewed data. Conversely, OSS posts the lowest averages, implying that under-sampling can degrade performance by ignoring valuable majority class information. Although good on individual occasions, the overall average is lower for ENN, demonstrating its sensitivity to the dataset's properties.

Analysis of Negative Predictive Value (NPV)

While the F1 score provides insight into balanced classification, the Negative Predictive Value (NPV) is arguably more critical in many risk management applications. NPV measures the model's reliability when predicting a negative outcome (i.e., an absence of risk). A high NPV is essential in domains like bankruptcy prediction or loan approval, where a false negative—classifying a high-risk entity as low-risk—can have severe financial consequences.

Table 3. NPV of all algorithms in the collected datasets

Dataset	Classifier	RUS	ENN	OSS	ROS	SMOTE	ADASYN	SMOTEENN	SMOTETomek
Taiwan-Bankruptcy	DT	**0.795**	0.405	0.290	0.275	0.419	0.416	0.571	0.428
	GBDT	**0.860**	0.381	0.244	0.652	0.587	0.581	0.734	0.585
	KNN	**0.843**	0.359	0.187	0.505	0.644	0.649	0.724	0.644
	LR	**0.820**	0.427	0.203	0.773	0.736	0.754	0.801	0.736
	MLPC	**0.832**	0.459	0.264	0.320	0.313	0.317	0.588	0.319
	RF	**0.868**	0.359	0.172	0.238	0.477	0.463	0.674	0.472
	SVM	**0.795**	0.388	0.156	0.755	0.722	0.742	0.791	0.721
	XGBoost	**0.856**	0.359	0.213	0.319	0.405	0.398	0.660	0.406
HR-Employee-Attrition	DT	**0.624**	0.443	0.400	0.325	0.415	0.417	0.577	0.419
	GBDT	**0.712**	0.443	0.349	0.564	0.390	0.379	0.681	0.384
	KNN	0.574	0.241	0.149	0.513	0.652	0.658	**0.844**	0.654
	LR	0.739	0.485	0.368	0.727	0.714	0.711	**0.833**	0.712
	MLPC	**0.710**	0.548	0.433	0.451	0.444	0.446	0.692	0.447
	RF	**0.684**	0.291	0.189	0.235	0.293	0.295	0.617	0.293
	SVM	0.737	0.452	0.343	0.726	0.710	0.715	**0.836**	0.712
	XGBoost	**0.701**	0.439	0.357	0.373	0.376	0.370	0.643	0.378
Loan-Prediction	DT	0.847	0.797	0.584	0.840	**0.859**	0.855	0.608	0.859
	GBDT	**0.596**	0.009	0.002	0.592	0.485	0.249	0.277	0.487
	KNN	0.637	**0.747**	0.011	0.636	0.639	0.653	0.720	0.639
	LR	0.556	0.000	0.000	0.556	**0.557**	0.370	0.314	**0.557**

MLPC	0.695	0.174	0.047	0.716	0.726	0.652	0.497	**0.728**
RF	0.775	0.724	0.535	0.774	0.789	**0.797**	0.562	0.789
SVM	**0.557**	0.000	0.000	0.556	**0.557**	0.369	0.307	**0.557**
XGBoost	**0.799**	0.551	0.234	0.796	0.763	0.607	0.401	0.764

According to Table 3, it can be concluded based on several key points regarding the performance of the different sampling schemes on different datasets and classifiers. RUS (Random Under-Sampling) performed consistently well, especially standing out in the Taiwanese Bankruptcy Prediction dataset using DT (0.986), GBDT (0.985), KNN (0.982), LR (0.981), MLPC (0.988), RF (0.984), SVM (0.980), and XGBoost (0.983). Likewise, in the IBM HR Attrition dataset, the top NPV scores were attained by RUS using DT (0.914), GBDT (0.906), MLPC (0.929), RF (0.930), and XGBoost (0.924), indicating robustness under moderate imbalance conditions. These observations imply that RUS effectively reduces false negatives, particularly when the feature structure is simpler and the dataset is highly imbalanced.

The SMOTEENN approach also provides robust performances, especially in the IBM dataset, with the maximum NPV using KNN (0.910), LR (0.920), and SVM (0.914). That implies that hybrid methods using oversampling and instance filtering can efficiently enhance minority class detection without seriously penalizing the model for missing true negatives.

In the Loan Prediction data, the performance of sampling methods is mixed. RUS is competitive again, obtaining the best scores using GBDT (0.985), SVM (0.983), and XGBoost (0.984). SMOTE does best using DT (0.985) and LR (0.962), while ADASYN obtains the best NPV using RF (0.980). ENN is best with KNN (0.981), and SMOTETomek is best used using LR (0.962), MLPC (0.982), and SVM (0.983). These observations reinforce the necessity to match sampling methods to the properties of specific classifiers in large, moderately imbalanced data.

Overall, RUS remains the most reliably high-performing technique for achieving maximum NPV, particularly in severe or moderate class imbalance cases. Hybrid techniques such as SMOTEENN and SMOTETomek also have high potential for balancing sensitivity and specificity across different scenarios.

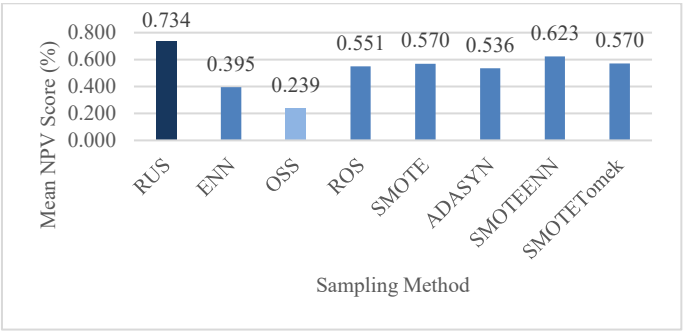

Figure 5. Mean value of NPV on all algorithms

Figure 5 shows the average NPV values for every sampling technique across several classifiers and datasets; the highest average NPV is obtained using the RUS (Random Under-Sampling) technique. This indicates the excellent ability of RUS to label true negatives well in imbalanced classes. One reason for this effectiveness is that it simplifies class distribution, making it easier for classifiers to separate classes and lowering the rate of false negatives. Therefore, it is instrumental in applications sensitive to risks where missing the majority class (negatives) can result in high repercussions.

Conclusion

This comprehensive comparative analysis of eight sampling methods across three distinct risk management domains has yielded several key findings. The core insight is that the effectiveness of a

sampling method is context-dependent, shaped by the analytical goal, the characteristics of the dataset, and the classification algorithm in use. Our dual-metric evaluation using F1 Score and Negative Predictive Value (NPV) highlights a strategic trade-off inherent in applying machine learning to imbalanced risk data. For applications seeking balanced predictive performance on rare, critical events—where detecting the minority class is as important as managing false positives—oversampling and hybrid techniques such as SMOTE, Random Over Sampling, and SMOTETomek proved most effective. These methods enhance classifier sensitivity to the minority class, particularly in large datasets with more reliable synthetic sample generation.

Conversely, in risk-averse applications where minimizing costly false negatives is the top priority—such as in bankruptcy or credit default prediction—Random Undersampling (RUS) is the most effective method. It consistently achieved the highest NPV across nearly all classifiers and datasets, producing conservative models that reliably identify truly safe cases. This study affirms that no single sampling method universally outperforms the rest; instead, method selection must align with the specific goals of the risk assessment task. Practitioners should determine whether their priority lies in balanced detection (F1 Score) or maximizing the certainty of adverse outcomes (NPV). This decision, informed by the nature of the dataset, should guide the strategic choice of sampling technique for building robust and context-appropriate risk prediction models.

While this study provides a systematic comparison of data-level sampling techniques, it has certain limitations that suggest directions for future research. The focus on algorithm-agnostic sampling methods excluded algorithm-level solutions, such as cost-sensitive learning, where misclassification costs are embedded directly into the learning objective. Future studies could evaluate the comparative effectiveness of such cost-sensitive classifiers alongside the sampling approaches presented here to understand their relative strengths better. Additionally, emerging paradigms like meta-learning—where models learn to adapt quickly to new tasks—offer promising opportunities as imbalanced learning evolves. These frameworks are designed to handle severe class imbalance and shifting data distributions by leveraging prior knowledge and task-specific adaptation. Exploring their application in risk management contexts could lead to more robust and flexible predictive models. This study thus serves as a foundational benchmark for assessing the performance of such future innovations.

References

[1] G. Niehaus, “Enterprise Risk Management and the Risk Management Process,” *The Palgrave Handbook of Unconventional Risk Transfer*, pp. 109–142, Aug. 2017, doi: 10.1007/978-3-319-59297-8_5.

[2] V. K. Shrivastava, J. Balasubramanian, A. Katyal, A. Yadav, and S. Yogananthan, “Understanding the significance of risk management in enterprise management dynamics,” *Multidisciplinary Reviews*, vol. 6, 2024, doi: 10.31893/MULTIREV.2023SS093.

[3] R. Gorvett, “Behavioral Economics and Its Implications for Enterprise Risk Management,” 2012.

[4] F. Ahmed, K. Nizam, Z. Sajid, S. Qamar, and Ahsan, “Striking a Balance: Evaluating Credit Risk with Traditional and Machine Learning Models,” *Bulletin of business and economics*, vol. 13, no. 2, pp. 999–1004, Aug. 2024, doi: 10.61506/01.00425.

[5] H. Xu, K. Niu, T. Lu, and S. Li, “Leveraging artificial intelligence for enhanced risk management in financial services: Current applications and prospects,” *Engineering Science & Technology Journal*, vol. 5, no. 8, pp. 2402–2426, Aug. 2024, doi: 10.51594/ESTJ.V5I8.1363.

[6] W. C. Aaron, O. Irekponor, N. T. Aleke, L. Yeboah, and J. E. Joseph, “Machine learning techniques for enhancing security in financial technology systems,” *International Journal of Science and Research Archive*, vol. 13, no. 1, pp. 2805–2822, Oct. 2024, doi: 10.30574/IJSRA.2024.13.1.1965.

[7] Y. Zhao, “Integrating Advanced Technologies in Financial Risk Management: A Comprehensive Analysis,” *Advances in Economics, Management and Political Sciences*, vol. 89, no. 1, pp. 49–54, Jun. 2024, doi: 10.54254/2754-1169/89/20241908.

[8] Q. Zhang, “A Survey on Imbalanced Data Learning Method,” *Computer Science*, 2005.

[9] S. Birla, K. Kohli, and A. Dutta, “Machine Learning on imbalanced data in Credit Risk,” *IEEE Annual Information Technology, Electronics and Mobile Communication Conference*, Nov. 2016, doi: 10.1109/IEMCON.2016.7746326.

[10] Y. Zhang, “Stroke Prediction Based on Machine Learning,” *ITM Web of Conferences*, vol. 70, p. 04029, Jan. 2025, doi: 10.1051/ITMCONF/20257004029.

[11] B. Ozturk, T. Lawton, S. Smith, and I. Habli, "Balancing Acts: Tackling Data Imbalance in Machine Learning for Predicting Myocardial Infarction in Type 2 Diabetes," *Medical Informatics Europe*, vol. 316, pp. 626–630, Aug. 2024, doi: 10.3233/SHTI240491.

[12] K. Okada *et al.*, "Abstract 15027: Predicting Recurrence of Myocardial Infarction in Post-PCI Patients Using Machine Learning," *Circulation*, vol. 148, no. Suppl_1, Nov. 2023, doi: 10.1161/CIRC.148.SUPPL_1.15027.

[13] N. N. Zhang, S. Z. Ye, and T. Y. Chien, "Imbalanced Data Classification Based on Hybrid Methods," *Proceedings of the 2nd International Conference on Big Data Research*, pp. 16–20, Oct. 2018, doi: 10.1145/3291801.3291812.

[14] S. Budania, T. Kumar, H. Kumar, and G. Nikam, "Hybrid Machine Intelligence for Imbalanced Data," *Social Science Research Network*, May 2020, doi: 10.2139/SSRN.3602531.

[15] J. Zhao, J. Jin, S. Chen, R. Zhang, B. Yu, and Q. Liu, "A weighted hybrid ensemble method for classifying imbalanced data," *Knowl. Based Syst.*, vol. 203, Sep. 2020, doi: 10.1016/J.KNOSYS.2020.106087.

[16] S. Shi, J. Li, D. Zhu, F. Yang, and Y. Xu, "A hybrid imbalanced classification model based on data density," *Inf. Sci.*, vol. 624, pp. 50–67, May 2023, doi: 10.1016/J.INS.2022.12.046.

[17] Ajmal M S, Tanmay Deshpande, and IBM Data Scientists, "IBM HR Analytics Employee Attrition & Performance," 2023, *IEEE Dataport*. doi: 10.21227/2m1g-6v47.

[18] S. Surana, "Loan Prediction Based on Customer Behavior." Accessed: Jun. 02, 2025. [Online]. Available: https://www.kaggle.com/datasets/subhamjain/loan-prediction-based-on-customer-behavior

[19] "Taiwanese Bankruptcy Prediction," 2020, *UCI Machine Learning Repository*. doi: 10.24432/C5004D.

[20] S. Ray, "A Quick Review of Machine Learning Algorithms," in *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*, IEEE, Feb. 2019, pp. 35–39. doi: 10.1109/COMITCon.2019.8862451.

[21] F. Pedregosa *et al.*, "Scikit-learn: Machine Learning in Python," *J. Mach. Learn. Res.*, vol. 12, no. null, pp. 2825–2830, Nov. 2011.

[22] J. H. Friedman, "Greedy function approximation: A gradient boosting machine.," *The Annals of Statistics*, vol. 29, no. 5, Oct. 2001, doi: 10.1214/aos/1013203451.

[23] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Trans Inf Theory*, vol. 13, no. 1, pp. 21–27, Jan. 1967, doi: 10.1109/TIT.1967.1053964.

[24] D. W. Hosmer, S. Lemeshow, and R. X. Sturdivant, *Applied Logistic Regression*. Wiley, 2013. doi: 10.1002/9781118548387.

[25] X. Glorot, A. Bordes, and Y. Bengio, "Deep Sparse Rectifier Neural Networks," in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, G. Gordon, D. Dunson, and M. Dudík, Eds., in Proceedings of Machine Learning Research, vol. 15. Fort Lauderdale, FL, USA: PMLR, May 2011, pp. 315–323. [Online]. Available: https://proceedings.mlr.press/v15/glorot11a.html

[26] L. Breiman, "Random Forests," *Mach Learn*, vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.

[27] I. H. Sarker, A. S. M. Kayes, and P. Watters, "Effectiveness analysis of machine learning classification models for predicting personalized context-aware smartphone usage," *J Big Data*, vol. 6, no. 1, p. 57, Dec. 2019, doi: 10.1186/s40537-019-0219-y.

[28] C. Cortes and V. Vapnik, "Support-vector networks," *Mach Learn*, vol. 20, no. 3, pp. 273–297, Sep. 1995, doi: 10.1007/BF00994018.

[29] J. H. Friedman, "Greedy function approximation: A gradient boosting machine.," *The Annals of Statistics*, vol. 29, no. 5, Oct. 2001, doi: 10.1214/aos/1013203451.

[30] T. Chen and C. Guestrin, "XGBoost," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA: ACM, Aug. 2016, pp. 785–794. doi: 10.1145/2939672.2939785.

[31] Haibo He and E. A. Garcia, "Learning from Imbalanced Data," *IEEE Trans Knowl Data Eng*, vol. 21, no. 9, pp. 1263–1284, Sep. 2009, doi: 10.1109/TKDE.2008.239.

[32] D. L. Wilson, "Asymptotic Properties of Nearest Neighbor Rules Using Edited Data," *IEEE Trans Syst Man Cybern*, vol. SMC-2, no. 3, pp. 408–421, Jul. 1972, doi: 10.1109/TSMC.1972.4309137.

[33] D. Guan, W. Yuan, Y.-K. Lee, and S. Lee, "Nearest neighbor editing aided by unlabeled data," *Inf Sci (N Y)*, vol. 179, no. 13, pp. 2273–2282, Jun. 2009, doi: 10.1016/j.ins.2009.02.011.

[34] M. Kubat and S. Matwin, "Addressing The Curse Of Imbalanced Training Sets: One-sided Selection," *International Conference on Machine Learning*, vol. 97, p. 179, 1997.

[35] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, Jun. 2002, doi: 10.1613/jair.953.

[36] Haibo He, Yang Bai, E. A. Garcia, and Shutao Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, IEEE, Jun. 2008, pp. 1322–1328. doi: 10.1109/IJCNN.2008.4633969.

[37] G. E. A. P. A. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," *ACM SIGKDD Explorations Newsletter*, vol. 6, no. 1, pp. 20–29, Jun. 2004, doi: 10.1145/1007730.1007735.