Cumhuriyet Sci. J., 46(4) (2025) 990-1002
DOI: https://doi.org/10.17776/csj.1605507

Cumhuriyet Science Journal

| csj.cumhuriyet.edu.tr | Founded: 2002 ISSN: 2587-2680 e-ISSN: 2587-246X Publisher: Sivas Cumhuriyet University

Advanced Machine Learning Strategies for Effective Rare Event Classification: A Comparative Study

Olcay Alpay[1,a,*]

[1] *Department of Data Science and Analytics, Sinop University, Sinop, Turkiye.*

**Corresponding author*

Research Article

History
Received: 22/12/2024
Accepted: 24/12/2025

ABSTRACT

As data science and machine learning continue to evolve, binary event classification has become increasingly important. Logistic Regression (LR) is a standard baseline, yet it can underestimate probabilities in rare-event settings. This study combines a 16-scenario simulation (rarity 5–10%, n∈{1000,5000}, p∈{3,5,7,10}, 100 repeats) with a real-world application to assess Support Vector Machines (SVM), Random Forest (RF), and Gradient Boosting (GB) as alternatives to LR. Training data were balanced using the SMOTETomek hybrid method. In simulation, LR attained the highest balanced performance (G_{mean}) across cases, with GB the closest competitor; SVM lagged, and RF yielded the lowest G_{mean} despite often leading test accuracy and precision. On a wine dataset adjusted to 5% and 10% rarity, RF/GB achieved top test accuracy/recall (e.g., ACC=0.998/0.997 with REC=0.958/0.989 at 5% and ACC=0.998/0.994 with REC=0.989/0.977 at 10%), mirroring their strong aggregate accuracy. Overall, the "best" model depends on the target metric: LR/GB when balanced minority–majority performance is critical, and RF when overall accuracy/precision is prioritized.

Keywords: Machine learning, Rare event, Performance metrics, SMOTETomek

[a] olcayb@sinop.edu.tr https://orcid.org/0000-0003-1446-0801

Introduction

Binary event classification is increasingly central to data science and ML across engineering, economics, healthcare, and politics. Yet class rarity—e.g., pandemics, fraud, landslides, earthquakes, conflicts [1]—poses well-documented challenges [2]: minority classes are under-represented, models bias toward the majority, and predictive performance deteriorates [3]. These obstacles motivate methods that explicitly address rarity, such as resampling and metric choices that prioritize minority-class detection.

If the classifier predicts that all data belongs to the majority class, it may achieve nearly 100% accuracy. However, this doesn't accurately reflect the predictive performance, especially when dealing with rare class instances. Moreover, classifiers may treat rare class instances as insignificant and might eliminate or misclassify them [4]. The use of resampling methods is a prevalent strategy utilized to confront the challenges posed by rarity in the data. These methods aim to oversample the minority class, undersample the majority class, or use a combination of both. Some commonly used resampling techniques include oversampling, undersampling, and hybrid sampling i.e. SMOTE (Synthetic Minority Over-sampling Technique), SMOTETomek. These techniques are essential in improving the performance of classification algorithms when dealing with rare events.

There are existing studies in the literature that investigate the impact of resampling methods on the application of classification algorithms to rare events. [5] experimented with various resampling methods in machine learning techniques on a dataset containing student performance, and they found that hybrid methods were more successful. [6] conducted a study to evaluate the effectiveness of SMOTE, Tomek, and SMOTETomek resampling techniques when classifying rare text data from a women's newspaper using SVM and LR. The study demonstrated the usefulness of using the SMOTE or SMOTETomek approach as supported by experimental results. [7] compared ten different resampling methods, including SMOTETomek, using nine different classification algorithms such as Logistic Regression (LR), Support Vector Machines (SVM), Random Forest (RF), and Gradient Boosting (GB), among others, to predict students' academic performance. The results indicated that SMOTETomek is an effective method for reducing class rarity.

LR, owing to its effectiveness and simplicity, stands out as one of the most preferred algorithms in this context. In this study, we adopt LR as the baseline because the outcome is binary/nominal and LR directly models class probabilities via the canonical logit link. Its coefficients are interpretable (odds ratios), and LR is widely used as a reference in rare-event studies, enabling transparent comparison with non-linear alternatives. However, its effectiveness in classification performance may be limited in the case of rare events due to biased predictions. LR tends to underestimate event probabilities, leading to misclassification issues. Various classification algorithms have been explored as alternatives to LR in rare event classification problems, demonstrating potential improvements. Nevertheless, it is important to acknowledge that the rarity level may lead classification algorithms to overfitting errors, and many learning methods may struggle to identify rare events during training [8].

Logistic regression has many applications in various fields and is a widely used method for classifying binary events. [9] investigated the performance of LR, SVM, RF and other classifiers on datasets containing rare events, and compared two techniques for improving classification performance. One technique involves increasing the number of data, while the other aims to balance the occurrence of rare events. The results showed that an increase in both the dataset size and the occurrence of rare events led to an improvement in performance. [10] conducted a study on quality monitoring in big data using four machine learning algorithms, including LR and SVM. The study analyzed four highly/ultra-rare datasets and presented a parsimonious scheme through feature and model selection. In the study [11] the classification performances of traditional classification algorithms logistic regression, and boosting algorithms were examined across different levels of rarity and sample measurements. The study indicated that logistic regression outperformed the boosting algorithm, particularly for rare datasets, asserting it as the optimal method. [12] conducted a study using logistic regression and undersampling to investigate the association between vitamin D and cancer incidence in a rare dataset on public health. The authors reported that balancing the dataset played a crucial role in the development of the model.

SVM is an effective tool for the analysis of high-dimensional data and the categorization of data sets. Moreover, it is an efficacious technique for binary classification. The remarkable capacity of the Support Vector Machine (SVM) for making generalizations, along with its optimal solution and its capacity to discriminate, has attracted considerable attention in the academic community [13]. In the course of reviewing the literature on SVM, a number of studies dealing with rare events stand out as particularly valuable contributions to the field. [14] introduced a two-stage meta-model importance sampling method that uses SVM to accurately evaluate the likelihood of infrequent structural failures in multiple failure zones. [15] have proposed a novel approach (linear support vector machine decision tree (LSVM-DT)) that combines SVM and decision tree (DT) algorithms to address the issues concerning model selection and identification of rare events. This algorithm utilizes only one regularization parameter in the model's selection process. [16] designed the anomaly frequency method (AFM) using the SVM algorithm in their study regarding extreme rainfall. They showed that the method gives better results in terms of false alarm and prediction. Other researchers have used SVM algorithm in their studies on rare natural phenomena such as earthquakes and landslides ([17], [18], [19], and [20]).

RF is an ensemble learning method that was introduced by [21] and has been widely applied to both classification and estimation problems due to its remarkable performance and relatively simple design [22]. The RF algorithm is capable of processing large datasets with high dimensionality and resisting overfitting. The method is notably effective in addressing rarity, due to its capacity for compensating for errors between classes. The following examples demonstrate the application of the RF algorithm to the analysis of rare events. [23] compared the effectiveness of LR and RF algorithms in predicting the rarity of civil wars. Using metrics such as AUC and F1_Score, the study found that the algorithmic approach was considerably more precise than any of the LR models in determining the onset of civil war in out-of-sample data. [24] evaluated the effectiveness of various algorithms (logistic regression, random forest, decision trees, and bagging) for classifying rare political events using ROC curves and error rates and demonstrated that RF was the most effective algorithm in his dataset. [25] introduced Ensemble Random Forests (ERFs) as a heuristic extension of the RF algorithm to deal with the bias of rare events in the data, and demonstrated by simulation the superior performance of ERFs compared to RF.

Another effective ensemble technique is GB, which builds models in a sequential manner, with each subsequent model correcting the errors of the previous model. These techniques constitute a substantial contribution to the discipline of rare event classification, primarily due to their capacity to process rare data sets. A number of notable studies have employed the GB algorithm for the classification of rare events, as illustrated below. In their study, [26] demonstrated that GB performs well on high-dimensional data; however, when applied to rare events, it tends to misclassify only a small portion of samples from the rare class. They suggested that this limitation can be addressed through corrections such as shrinkage and subsampling, as evidenced by their analysis of clinical data. The classifiers were evaluated by comparing their performance using performance metrics, ROC and G_{mean}. [27] employed the LightGBM machine learning classifier to identify anomalies within networks. They conducted a comparative analysis, contrasting their model with other widely favored classification algorithms. Through the evaluation of performance metrics, they demonstrated that their model outperforms alternative methods in the detection of network anomalies. [28] integrated the GB approach with the minority oversampling technique SMOTE to examine stroke data. They explored the impact of oversampling on classifier performance and conducted a comparative analysis with the results obtained using the random forest classifier. In a study conducted by [29], the researchers employed the SMOTETomek and SMOTE resampling techniques in conjunction with the Random Forest algorithm for the purpose of classifying Pima Indian

Diabetes. Their findings indicated that the utilization of SMOTETomek resulted in a notable enhancement in the efficiency of the Random Forest algorithm. In a recent study, [30] employed the SMOTETomek resampling technique for intrusion detection in wireless sensor networks (WSNs) with the objective of identifying the most effective algorithm among a number of algorithms, including random forest (RF). The findings indicate that RF consistently demonstrates excellent performance across a range of scenarios.

When exploring alternative approaches to logistic regression, it becomes clear that certain machine learning algorithms - including SVM, RF, and GB - have been utilized for analyzing rare event data. Nevertheless, the classification performance of these algorithms under diverse constraints, including model complexity, rarity levels, and sample sizes, has yet to be comprehensively examined. The objective of this study is to address this gap in literature by comparing the classification performance of machine learning algorithms under different rarity levels, sample sizes, and model complexities.

We employ SMOTETomek to address rarity. SMOTE synthesizes minority-class samples, while Tomek links remove noisy/borderline instances; together they increase minority representation and reduce overlap. SMOTETomek is applied only to the training split (and within CV folds); the test split is never resampled nor used for model selection, preserving an unbiased evaluation.

The evaluation of classification performance will employ accuracy, precision, recall, and F1_Score. While accuracy reflects overall correctness, precision, recall, and F1_Score are especially informative for rare-event settings because they emphasize minority-class detection. The objective of this comprehensive assessment is to utilize the metric suite to demonstrate the potential of machine learning algorithms for rare-event classification and, when appropriate, to establish their superiority over LR. The results of this assessment will guide practice and inform future research.

The section below describes the methodologies the LR, SVM, RF, and GB algorithms, SMOTETomek technique, as well as the metrics used to evaluate classification performance. The following section presents a simulation study and an application on the real data set. This will provide insight into the results in four different scenarios involving high-dimensional data with varying degrees of rarity and sample sizes. The results will be evaluated in detail in the final section. The final section comprehensively presents and discusses the results obtained, providing a deeper understanding of the conclusions and significance of the study.

Materials and Methods

The primary objective of this study is to focus on the classification of rare events, evaluating the performance of various machine learning algorithms such as logistic regression, support vector machine, random forest, and gradient boosting. Understanding the usage of these algorithms and their performance metrics contributes to identifying effective strategies for accurately classifying rare events. Across all scenarios, we used a 2/3:1/3 train–test split and repeated the entire pipeline 100 times with independent random seeds to ensure stability. All model selection and resampling were confined to the training split; the test split was never used during tuning.

Logistic Regression

Logistic Regression (LR) is a member of the generalized linear models [31]. Let $X \in R^{n \times p}$ the matrix of explanatory variables and the outcome variable *"y"* has a Bernoulli probability function with occurrence probability $\pi_i = P(y_i = 1|X_i)$ which denotes the class of interest. The logit link function, which represents the relationship between the variables, is defined as follows.

$$\eta = g(\pi_i) = ln\left(\frac{\pi_i}{1-\pi_i}\right) = \beta_0 + \beta_1 X_{i1} + \cdots + \beta_k X_{ip} = \mathrm{X_i^T \beta}$$

Where β_i are the coefficients. The probability of the event of interest is

$$P(y_i = 1|X_i) = \frac{e^{\mathrm{X_i^T\beta}}}{1 + e^{\mathrm{X_i^T\beta}}}$$

Log-likelihood function and maximum likelihood estimate of β are expressed respectively as:

$$lnL(y,\beta) = \sum_{i=1}^{n} y_i \mathrm{X_i^T\beta} - \sum_{i=1}^{n} \ln\left(1 + e^{\mathrm{X_i^T\beta}}\right)$$

and

$$\mathcal{L}(\beta) = \sum_{i=1}^{n}(y_i - \pi_i)\mathrm{X_i}$$

The primary objective of logistic regression is to maximize the estimation of parameters by identifying weights that minimize the negative log probability on the training set [32].

The use of logistic regression is advantageous in that it does not require any prior assumptions regarding the distribution of the data. The primary benefit is that the resulting output inherently conveys probability information, whereby each predicted label is accompanied by a corresponding probability. Furthermore, the objective function is a high-order differentiable convex function, which can be solved by numerical optimization methods. Conversely, a smooth and monotonically decreasing logistic loss function necessitates a greater number of training samples, which consequently increases the estimation cost [33].

Support Vector Machine

Support Vector Machine (SVM) is a kernel based supervised learning algorithm introduced by [34]. It learns a decision boundary to distinguish between two classes in a training set and applies this rule to classify new test samples. SVM is well-known for its strong performance in various applications, which can be attributed to its solid theoretical foundation in statistical learning [35].

Let us consider the scenario in which a set of n training examples (x_i, y_i) comprising binary outputs $y_i \in \{-1,1\}$

is provided, where y_i represents a binary classification problem. To learn an optimal hyperplane, the SVM uses a nonlinear mapping function, denoted as ϕ, to transform the input sample into a high-dimensional feature space [36]. To address the issue of classification, SVM identifies the solution with the largest margin that satisfies the equation $y_i(w.\Phi(x) + b) \geq 1$ for all $x_i \in X$, where w is the weight vector $w \in R^D$ and $b \in R$ is the bias. This is situated between the $w.\Phi(x) + b = 0$ decision hyperplanes that differentiate the samples in the training set. Consequently, it provides a solution that optimizes the margin between the classes [37].

The significant benefit of SVMs is that, during the learning phase, they typically obtain a subset of support vectors that represents only a relatively small proportion of the original data set. This set of support vectors represents a specific classification task and is generated by a small dataset. While SVMs offer a number of advantages, they are not without certain disadvantages. These include the difficulty of selecting appropriate parameters, the algorithmic complexity which can result in extended training times on large datasets, challenges in developing optimal classifiers for multi-class problems, and suboptimal performance on rare datasets [13].

Random Forest

Random Forest (RF) is a collection of tree-structured classifiers $\{h(\mathbf{x}, \Theta k), k = 1, \dots\}$, where $\{\Theta k\}$ are independent, identically distributed random vectors, and each tree votes for the most popular class at input $\mathbf{x}$ [21]. Despite its widespread use in practice, little is known about the mathematical properties of random forests [38].

A simplified pseudo-code representation of a random forest is provided below:

1. Input the data.
2. For each tree in the forest:

Draw a bootstrap sample from the training data.
Grow a decision tree on the bootstrap sample, selecting a random subset of features at each split.
Store the predicted outcome of the tree for the given test data.

3. Aggregate the predictions of all trees.
4. Determine the majority class for classification or average prediction for regression as the final outcome [39].

Random forest has the advantages of being partially robust to outliers and noise, faster than bagging or boosting, producing useful internal estimates of error, power, correlation, and variable importance, and being simple and easily concurrent [21]. With its straightforward implementation, RF often exhibits favorable performance in real-world data applications and is regarded as a valuable component of contemporary ensemble learning methods. Although RF initially demonstrates low performance, as more base learners are added to the ensemble, RF's generalization error gradually approaches a lower value [32]. However, it has some disadvantages, such as computational cost and interpretability due to the large size of the resulting models [40].

Gradient Boosting

Gradient Boosting (GB) is a machine learning algorithm that constructs a robust predictive model by taking into account inaccuracies in the weaker models (weak learners) that precede it. The idea behind this process is that as each tree is added, it attempts to correct the errors of the previous trees, allowing each tree to capture the complex relationships in the dataset and tune the model more comprehensively [41].

In the context of function estimation, where $\mathbf{x} = \{x_1, x_2, \dots, x_n\}$ represents the input variables and y represents the output variables, the objective is to obtain an estimate the function $F(\mathbf{x})$ that maps $\mathbf{x}$ to y, minimizing the expected value of a given loss function $\phi(y, F(\mathbf{x}))$ over the joint distribution of all $(y, \mathbf{x})$ values. This is achieved using training samples $\{y_i, \mathbf{x}_i\}_1^N$ [42].

$$\hat{F} = arg \min_F E_{y,\mathbf{x}}[\phi(y, F(\mathbf{x}))]$$

A simple gradient boosting classification pseudo-code is provided as follows:

1. Initialize the model.
2. Calculate residual errors.
3. Fit a weak learner to the residuals.
4. Adjust the model using the predictions of the weak learner and a learning rate.
5. Update the model.
6. Repeat steps 2-5 for the required number iteration.

The selection of the loss function to be employed in GBCs is at the discretion of the researcher. A variety of loss functions have been derived and are currently available for use in the algorithm. The flexibility of GBMs allows for the design of models that are highly adaptable to the specific requirements of any data-driven task. Nevertheless, the algorithm is relatively straightforward to implement, thereby affording the opportunity for experimentation with different model designs [43].

In general, the presence of rarity in data sets presents a significant challenge for classification algorithms, particularly in the context of predicting rare classes. There are a number of techniques that can be employed to enhance the performance of classifiers. These methods can be classified into two categories: external and internal methods. The first category of methods is external, taking into account the number of samples in each class and attempting to achieve a balance between them. The second category of methods is internal, and their purpose is to enhance the performance of classification algorithms by optimizing the parameters of the algorithms themselves [44]. This study will concentrate on external techniques designed to optimize the performance of machine learning algorithms.

Resampling Techniques

Sampling is a frequently employed methodology for addressing the issue of data imbalance resulting from the rarity of instances within a given data set. The aim of sampling is to allow traditional classifiers to more accurately capture the decision boundary between majority and rare classes by creating a dataset with a more balanced class distribution [45]. The following are some of the techniques:

- ***i) Undersampling:*** Undersampling refers to randomly sampling the dataset to exclude some observations of the majority class to obtain a balanced class. In this way, rare events are better represented in the dataset [46]. However, it is important to note that these methods result in the discarding of vast quantities of data [45].
- ***ii) Oversampling:*** Oversampling aims to equalize the number of samples in both classes and balance the distribution of events in the data set by creating new samples within the rare class [47]. Although oversampling techniques can improve the representation of rare events in the dataset, they can also lead to issues related to overfitting [45].
- ***iii) Hybrid sampling (SMOTETomek):*** SMOTETomek is a hybrid approach that combines the synthetic minority oversampling technique (SMOTE) with the Tomek link. SMOTE is an oversampling method that replicates the number of rare events until it equals the number of the dominant class. Tomek link, on the other hand, is an undersampling method that is performed as a cleaning step after data processing in the research [48]. This hybrid approach ensures that reducing the major class does not result in significant information loss, while also preventing the problem of overfitting that may arise from oversampling [45].

Amongst alternative sampling approaches, the SMOTETomek hybrid approach was preferred in this study due to its effectiveness proved in the precedent rare event studies as mentioned above in the introduction. Critically, SMOTETomek was applied only to the training split (and, when cross-validating, within each training fold). The test split was never resampled nor used during model selection, preserving an unbiased evaluation.

Performance Metrics

When performing binary classification by means of classifiers, a performance evaluation is needed by measuring the correct classification rate. Performance metrics such as Accuracy (ACC), Precision (PREC), Recall (REC) or F1_Score can be used to determine the predictive performance of a rare class. The confusion matrix is defined a

		Predicted	
		1	0
Observed	1(rare class)	TP	FN
	0	FP	TN

also ACC, PREC, REC and F1_Score are given below:

$$ACC = (TP + TN)/N,$$

$$PREC = TP/(TP + FP),$$

$$REC = TP/(TP + FN),$$

and

$$F1\_Score = 2\left(\frac{Prec \times Rec}{Prec + Rec}\right)$$

FP (False Positive) means the class 0 is classified as class 1, and the FN (False Negative) the class 1 is classified as class 0.

The Receiver Operating Characteristic (ROC) curve is a map that shows the relationship between sensitivity and 1-specificity [49]. It is used to evaluate the diagnostic ability of a classifier by plotting sensitivities against false positive rates obtained with respect to various cut-off values [50]. Although ROC is a powerful performance metric, it has been suggested that the area under the curve (AUC) is better suited for rare event classification [51]. AUC combines sensitivity and specificity metrics by summarizing the entire ROC curve rather than relying on a discrimination threshold [52].

The geometric mean (G_{mean}) aims to provide an objective evaluation of a classifier's performance using a numerical value. It is computed by taking the square root of the product of the rates of true positives and true negatives. The selection of a classifier on the basis of the geometric mean maximizes accuracy on both classes while maintaining balance [9]. The geometric mean is determined as

$$G_{mean} = \sqrt{TP_{rate} \times TN_{rate}}$$

Here, ACC, PREC, REC, F1_Score, AUC and G_{mean} are favored for comparing the performance of algorithms in the rarecase setting. All performance metrics (ACC, PREC, REC, F1_Score, AUC, G_{mean}) were computed on the test split. ROC curves and AUC values were derived from test predictions and, where summarized, aggregated over the 100 repetitions. For transparency, we also provide method-specific workflows for LR, SVM, and RF/GB in Appendix.

Simulation Study and Results

An empirical performance evaluation of LR, SVM, RF, and GB algorithms was conducted through a simulation study involving the creation of four samples with different rarity levels and sample sizes. In this study, we examine high-dimensional models with varying complexities consisting of p=3, 5, 7, and 10 variables. The model features standardized β_i coefficients set to 1, and explanatory variables assumed to follow a normal distribution $X_i \sim N(0,1)$. The intercept coefficient (β_0) requires particular attention, as its values can be employed to regulate the prevalence of a given class. Recognizing the critical role of sample size in reliably capturing the significance of rare events, we determined the sample size to be a minimum of 1000, aiming to ensure the reliable detection of true rarity levels. Rare events often exhibit low frequencies, and their impact might not be discernible in small sample sizes. Therefore, setting the sample size at a minimum of 1000 is intended to enhance statistical power and analyze the effects of rare events more robustly. Noting that the rare event of interest is represented as "1," and the class labels 0 and 1, we examined 16 different scenarios with class distributions of 95:5 and 90:10, each with a sample size of n=1000 and n=5000, labeled Case 1 (a, b, c, d)[1] to Case 4 (a, b, c, d), respectively (Table 1). To attain a 5% rarity level for the occurrence of interest, β_0 values of -4, -4.5, -5.2, and -6 were opted for a sample size of 1000 and the aforementioned number of variables (3 to 10), respectively. Similarly, to achieve a 10% level of rarity, the β_0 values were determined as -3.2, -3.5, -4, and -4.5, based on a sample size of 5000.

Then, the binary event with the above settings is generated as

$$y(\pi(x)) = \begin{cases} 0, & if \ \ \pi < u \\ 1, & otherwise \end{cases}$$

where $u \sim Uniform(0,1)$.

Table 1. Simulation design

	$n = 1000$	$n = 5000$
5%	Case1	Case3
10%	Case2	Case4

In this simulation study, we tuned model hyperparameters (excluding logistic regression) via 3-fold cross-validated grid search on the training split, using F1_Score as the primary selection criterion; the search spaces are given in Table 2. Our implementation evaluates all combinations within these spaces, and the reported results correspond to the combination achieving the best cross-validated F1_Score (with no information from the test split).

Table 2. Hyperparameters used in the search

Method	Parameter
SVM	*Kernel:* Linear, Sigmoid, Rbf
	C: $0.1, 0.5, 1, 10$
RF	*n_estimators:* $10, 50, 100$
	criterion: Gini, Entropy
GBC	*learning_rate:* $0.01, 0.1, 0.2$

The classification rates for all approaches were evaluated using ACC, PREC, REC, and F1_Score performance criteria in samples that were balanced using the SMOTETomek resampling method. The accompanying graphs illustrate the $\boldsymbol{G_{mean}}$ values of the algorithms, as shown in Figure 1 The simulations were conducted 100 times for all scenarios, and all calculations were executed in Spyder 5.4.3, utilizing Python.

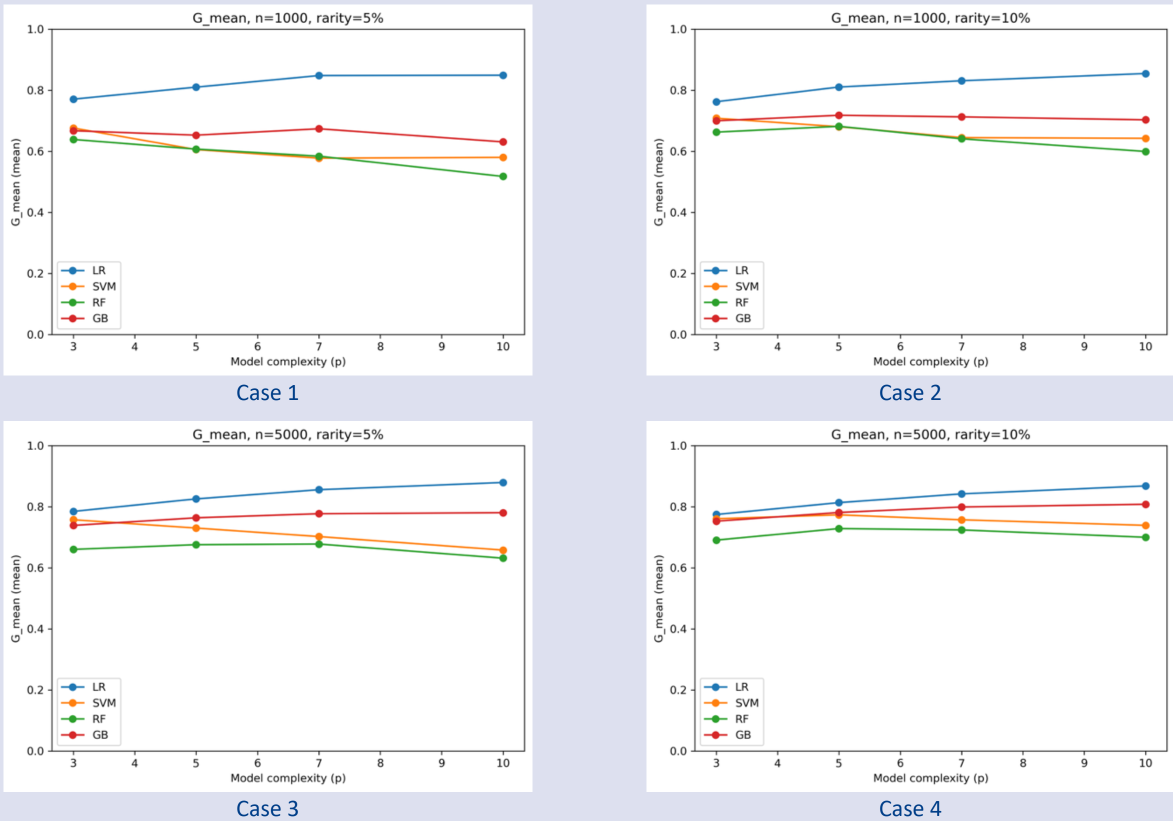
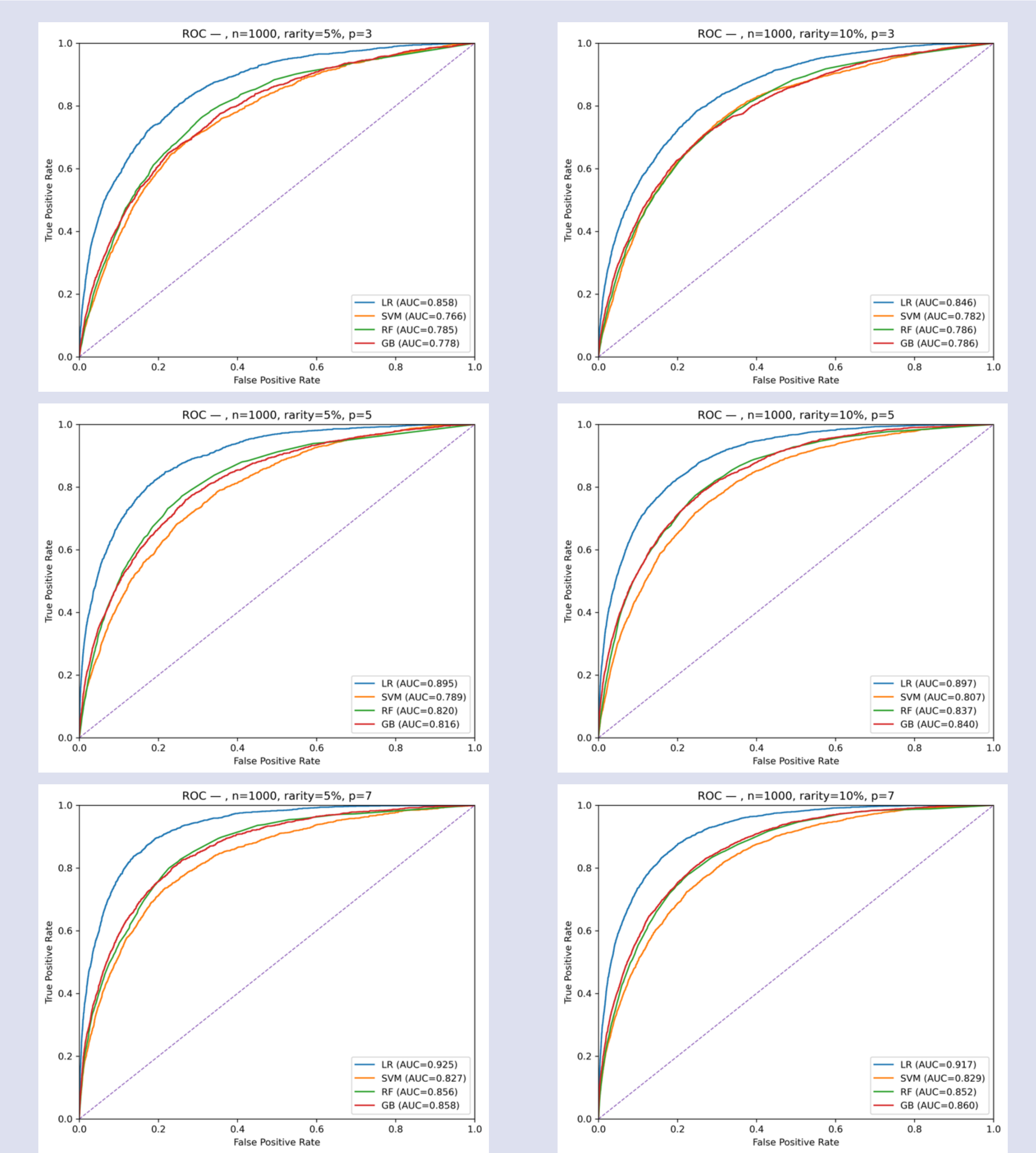


Figure 1. Variation of total energy (Ryd) versus lattice parameter (in Å) of WSe_2 2D- honeycomb structure.

[1] Lowercase letters represent the numbers of variables in model complexity. a:3,b:5,c:7 and d:10

In the presence of uniaxial strain, the energy bandgap value decreases with increasing tensile strain. Furthermore, the transition from semiconductor to semimetal and semimetal to metal occurs when the uniaxial strain takes higher values [18].

Initially, the optimised lattice parameter under strain was calculated by assigning a value less than 0.5 kB, and the Fermi energy value was set to zero in all the graphs drawn, as illustrated in Figure 2(a, b, c, d, e). The lattice parameter for the optimised geometry a_0, was 3.315 Å.



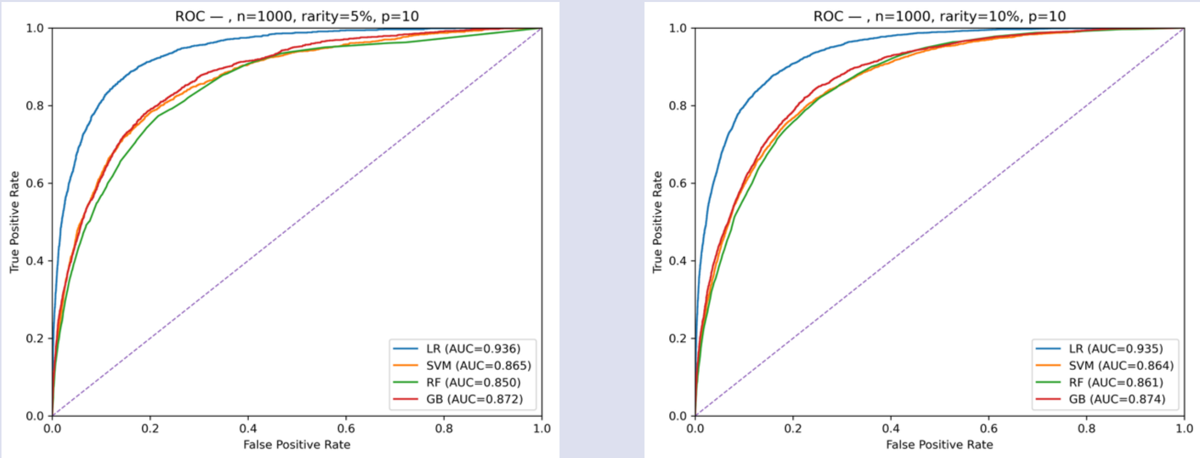
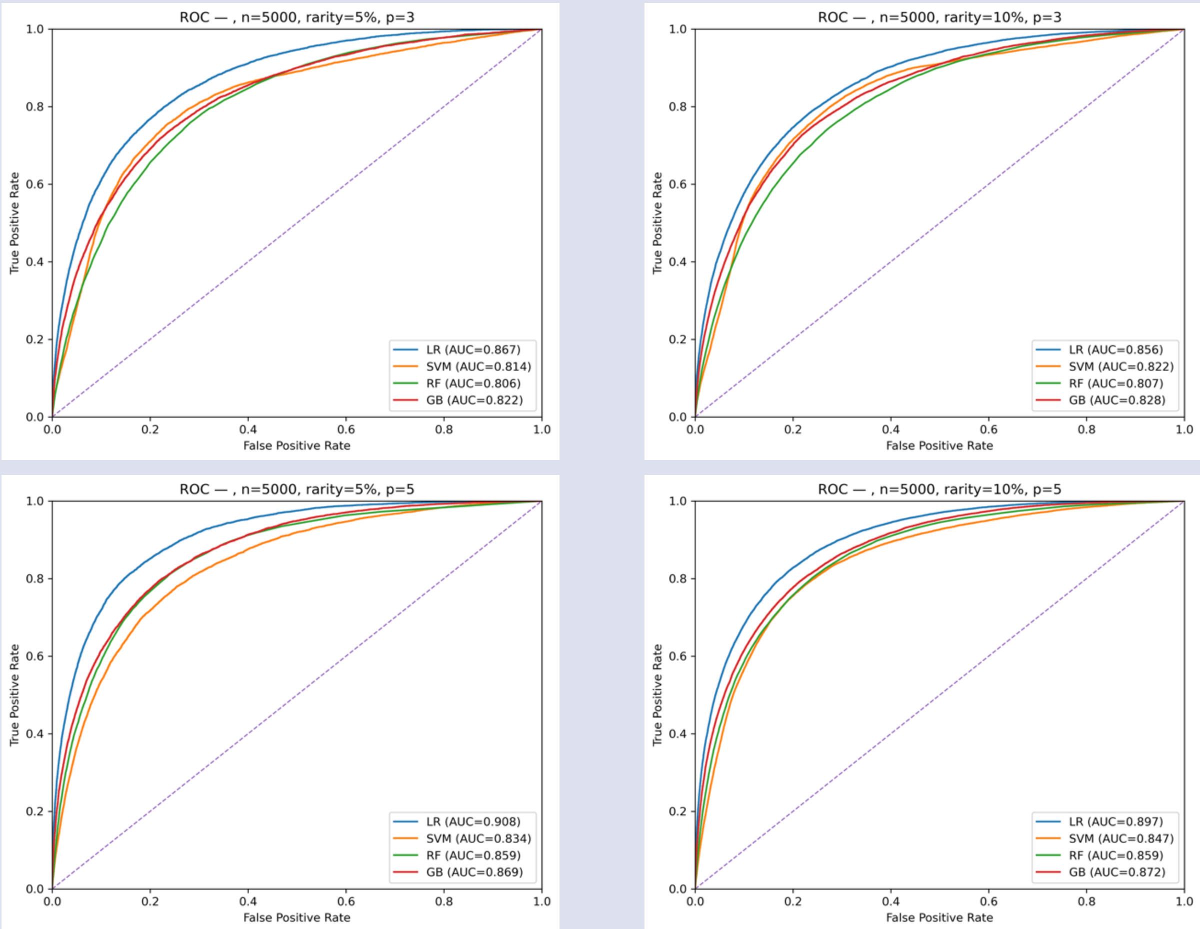


Figure 2. ROC graphics for sample size $\mathrm{n} = 1000$.



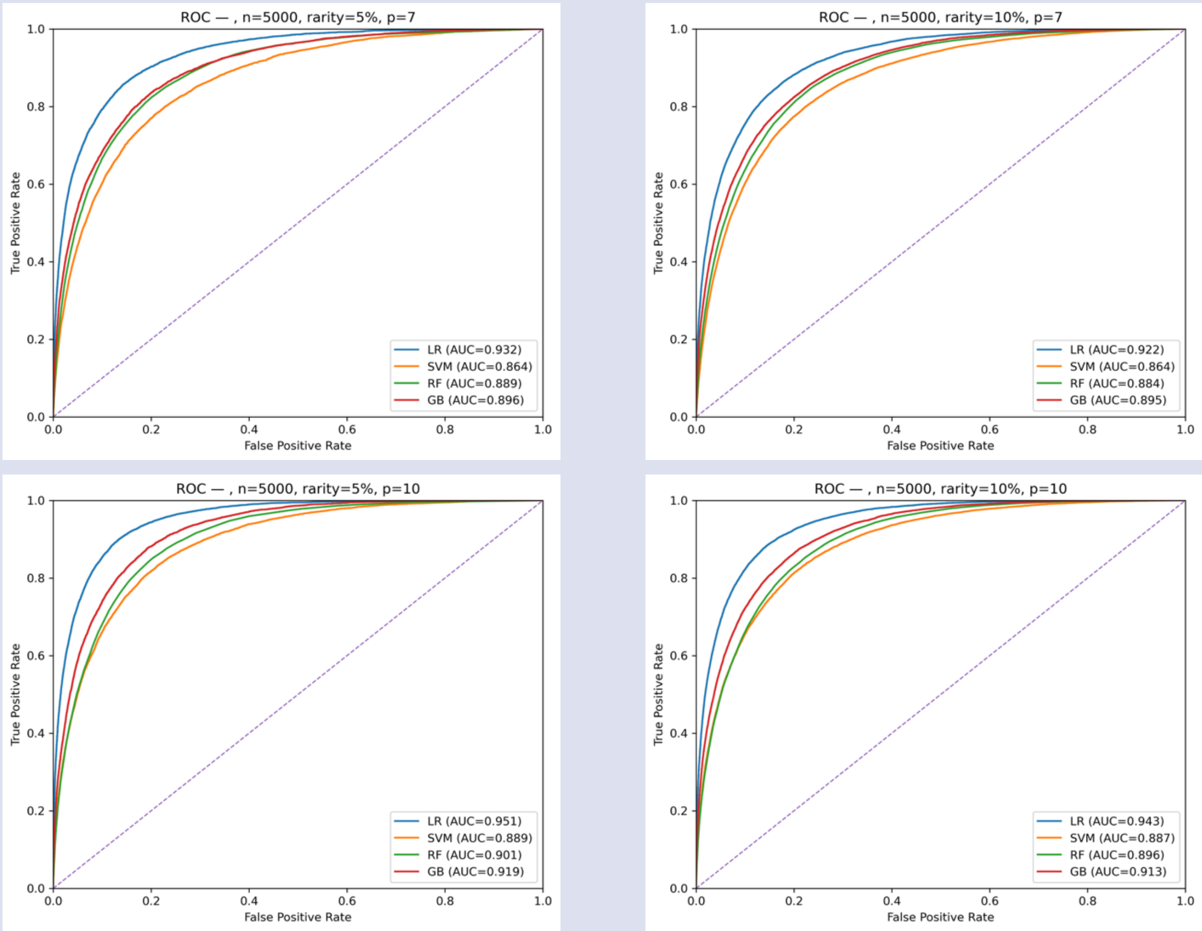


Figure 3. ROC graphics for sample size $n = 5000$.

Across all (n, p, rarity) settings, the ROC curves demonstrate a consistent ordering by AUC: LR consistently demonstrates the highest values, while RF and GB exhibit a close proximity to each other. SVM, on the other hand, exhibits the lowest values. An increase in the sample size results in an upward shift of all curves, but this does not lead to a significant alteration in the ordering of the curves.

Table 3. Performance metrics

		$p = 3$		$p = 5$		$p = 7$		$p = 10$	
		5%	10%	5%	10%	5%	10%	5%	10%
Models					ACC				
LR	$n = 1000$	0.837	0.796	0.821	0.809	0.827	0.854	0.880	0.846
SVM		0.822	0.769	0.832	0.724	0.813	0.851	0.872	0.838
RF		0.876	0.841	0.903	0.846	0.935	0.882	0.940	0.876
GB		0.856	0.816	0.801	0.784	0.880	0.860	0.868	0.871
LR	$n = 5000$	0.791	0.781	0.822	0.812	0.843	0.851	0.894	0.866
SVM		0.706	0.704	0.827	0.812	0.843	0.827	0.894	0.864
RF		0.866	0.831	0.899	0.867	0.890	0.922	0.940	0.894
GB		0.797	0.777	0.827	0.796	0.794	0.843	0.901	0.876
					PREC				
LR	$n = 1000$	0.232	0.267	0.207	0.364	0.158	0.423	0.253	0.396
SVM		0.216	0.241	0.213	0.269	0.149	0.419	0.240	0.384
RF		0.225	0.270	0.251	0.396	0.241	0.476	0.342	0.417
GB		0.218	0.261	0.174	0.313	0.165	0.416	0.187	0.421
LR	$n = 5000$	0.197	0.293	0.223	0.332	0.379	0.249	0.314	0.451
SVM		0.138	0.218	0.210	0.332	0.371	0.222	0.317	0.447
RF		0.228	0.312	0.286	0.396	0.466	0.347	0.427	0.533
GB		0.192	0.280	0.214	0.302	0.296	0.220	0.323	0.468
					REC				
LR	$n = 1000$	0.853	0.798	0.813	0.790	0.779	0.834	0.816	0.828
SVM		0.846	0.790	0.772	0.732	0.789	0.842	0.819	0.848
RF		0.525	0.500	0.404	0.531	0.292	0.475	0.315	0.379
GB		0.634	0.625	0.676	0.652	0.429	0.584	0.546	0.556
LR	$n = 5000$	0.782	0.804	0.798	0.815	0.848	0.836	0.875	0.858
SVM		0.717	0.720	0.692	0.816	0.779	0.836	0.807	0.859
RF		0.496	0.539	0.462	0.544	0.572	0.461	0.435	0.520
GB		0.720	0.749	0.696	0.739	0.727	0.683	0.687	0.718
					F1_Score				
LR	$n = 1000$	0.362	0.398	0.328	0.497	0.260	0.559	0.383	0.533
SVM		0.342	0.367	0.330	0.391	0.247	0.557	0.368	0.526
RF		0.311	0.347	0.304	0.450	0.254	0.471	0.316	0.390
GB		0.321	0.365	0.271	0.420	0.224	0.481	0.274	0.475
LR	$n = 5000$	0.314	0.429	0.348	0.472	0.523	0.383	0.462	0.591
SVM		0.231	0.334	0.321	0.471	0.502	0.350	0.445	0.588
RF		0.311	0.395	0.352	0.457	0.513	0.394	0.429	0.526
GB		0.303	0.407	0.326	0.428	0.419	0.330	0.433	0.566

According to Complexity

A comparison of the algorithm performance based on the criteria outlined in Table 3 indicates a general increase in the ACC values of the algorithms as the model complexity increases. The RF algorithm demonstrates superior performance in handling rare cases and achieving high classification accuracy on the balanced dataset when the SMOTETomek method is employed. A review of the PREC results indicates that the classification performance is relatively weak in comparison with other metrics. In this instance, the incorporation of a greater number of explanatory variables into the model has a beneficial impact on classification performance. Although RF is the most effective algorithm in this context, it is inadvisable to rely on PREC as a performance indicator in instances where data balancing is to be achieved through a hybrid approach. With regard to REC, the increase in model complexity resulted in a decline in the classification performance of the algorithms in comparison to the performance of the simpler models. In the case of relatively simple models comprising three or five variables, a change in the rarity of the distribution from 5% to 10% resulted in a slight decrease in performance. Conversely, for more complex models, distributions with lower rarity yielded superior results. In contrast, the opposite was observed in the case of large samples. Upon evaluation of the F1_Score value in consideration of model complexity, it is observed that both model complexity and classification performance exhibit a parallel movement across the basis of sample sizes.

According to Rarity

Upon analysis of the ACC metric in terms of rarity, it was observed that the algorithms exhibited enhanced classification performance in the samples when the data sets with a 5% rarity level were balanced. With respect to this metric, Random Forest (RF) produces the optimal results when coupled with the SMOTETomek method. With regard to the PREC values, the classification performance of all algorithms is enhanced by the application of a resampling method in data sets that are more balanced in terms of distribution. An increase in sample size has a further beneficial effect on the performance of the RF algorithm, which already exhibits the highest performance in this regard. With regard to the REC metric, the results demonstrate that an increase in model complexity and a more balanced distribution have a detrimental impact on classification performance. In the 1000-unit sample, all algorithms exhibited the highest classification values for the rarest and simplest model. As the sample size increased, model complexity and rarity of 10% resulted in a reversal of this situation. Finally, an examination of the F1_Score reveals that as the model complexity increases, the algorithms demonstrate enhanced performance for both sample sizes at the 10% rarity level. The performance is relatively superior for larger samples.

According to Sample Size

When the ACC values of the algorithms are evaluated according to the sample size, an increase in performance can be observed in complex models as the sample size is increased, with the exception of the simplest model. In the case of a 10% rarity, when the impact of model complexity is considered, the shift in sample size has no notable impact on the observed performance improvement. Upon examination of the PREC performance metric, it was observed that, at both levels of rarity, RF consistently demonstrated an increase in the correct classification performance with an expansion of the sample size, whether the model complexity was simple or complex. An examination of the REC metrics reveals that the algorithms exhibit a higher classification rate at larger sample sizes. For a sample size of 5,000 units, an increase in model complexity resulted in a notable enhancement in the REC values. Finally, an examination of the F1_Scores reveals a comparable outcome to that observed in the REC results. The expansion of sample size and model complexity has a beneficial impact on the classification performance of the algorithms.

Across the 16 scenarios, LR achieves the highest balanced performance (G_{mean}), with GB tracking closely and frequently ranking second. RF remains a strong alternative on overall correctness—its ROC curves are typically comparable to GB and it often attains the highest test accuracy/precision—but its G_{mean} is lower in our setup, reflecting a more conservative rare-class sensitivity under SMOTETomek rather than poor fit. Taken together, the results indicate two practical alternatives to the LR baseline: GB when balanced minority–majority detection is the priority, and RF when accuracy/precision objectives dominate. Increasing sample size and model complexity tends to lift all methods while preserving this ordering.

Application of Real Data and Results

In the real data application of this study, we employed a comprehensive dataset originally compiled by [53], which comprised 6497 samples of Portuguese wines obtained from the UCI Machine Learning Repository. This dataset provides foundation for analysis, featuring 11 measured explanatory variables along with a binary dependent variable that distinguishes between red and white wines. The features are fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, and alcohol.

The original dataset exhibits a distribution ratio of 0.25:0.75 for red and white wines, respectively. To control prevalence, we randomly subsample the minority class (red) while holding whites fixed, producing 5% ($n_{red}=259:n_{white}=4898$, $n=5157$) and 10% ($n_{red}=549:n_{white}=4898$, $n=5447$) scenarios. We then apply SMOTETomek only to the training split. Across both scenarios, we employ a 2/3:1/3 train–test split, repeat the entire pipeline 100 times with independent seeds, tune models only on training data via grid search, and report test averages. Table 4 presents the classification performance metrics for the algorithms in question following these operations.

Table 4. The results of the performance metrics for the real data set

	5%		10%	
Models	ACC	REC	ACC	REC
LR	0.985	0.958	0.979	0.972
SVM	0.988	0.968	0.992	0.989
RF	0.998	0.958	0.998	0.989
GB	0.997	0.989	0.994	0.977

Following the adjustment of the training distribution with SMOTETomek, RF demonstrates the highest level of test accuracy across both prevalence levels. The recall peaks with GB at 5%, and it is the co-highest for RF and SVM at 10%. This pattern is consistent with the findings from the simulations: tree ensembles (RF/GB) demonstrate superior performance when the objective is accuracy or low false positives, while LR/GB are the safer choices when balanced minority–majority detection is the priority. This underscores the notion that the metric of interest can alter the apparent 'best' model.

Relation to Prior Work

Our findings are consistent with studies showing that hybrid resampling improves rare-event classification and tends to boost performance as sample size/prevalence increase [5–7,9,28,29]. In our simulations, LR attains the most balanced performance (G_{mean}), echoing reports where LR can outperform boosting on rare datasets [11], while GB remains competitive but benefits from shrinkage/subsampling or oversampling [25,27]. In the real data, RF (and GB) achieve top ACC/REC, in line with applications where RF is reported as strongest under imbalance [22,23]. Mixed evidence for SVM under imbalance—highly sensitive to kernel/regularization and cost weighting—matches our results without cost-sensitive tuning [13]. Taken together, these patterns reconcile reports favoring LR under rare-event settings with those highlighting RF/GB under hybrid resampling: LR (and GB) better preserve balanced performance, while RF can dominate accuracy-oriented objectives.

Conclusion

In the context of binary event classification, Logistic Regression (LR) is widely regarded as the most straightforward and prevalent approach, particularly when the classes are equally distributed. However, LR may underestimate event probabilities at low prevalence, motivating comparisons with modern machine-learning alternatives. In this study, we systematically evaluated LR, Support Vector Machines (SVM), Random Forest (RF), and Gradient Boosting (GB) across multiple rarity levels, sample sizes, and model complexities. The evaluation used Accuracy (ACC), Precision (PREC), Recall (REC), F1_Score, ROC/AUC, and Geometric Mean (G_{mean}). SMOTETomek was applied only to the training split; the test split was never used for tuning, ensuring an unbiased assessment.

In the simulations, LR consistently achieved the highest balanced performance (G_{mean}) and generally improved with increasing model complexity and sample size. GB was the closest competitor, often narrowing the gap at higher complexity. RF and SVM trailed on G_{mean}, with RF's decline at higher complexity suggesting a sensitivity–specificity trade-off under SMOTETomek. These patterns are mirrored in the ROC panels: LR produced the largest AUC across cases, RF/GB tracked LR most closely (often nearly overlapping), while SVM formed a last tier. Although RF frequently led ACC and PREC, when balanced detection of both classes is prioritized (e.g., G_{mean}/F1), LR/GB can rank higher, illustrating that the choice of metric can invert the apparent model ranking.

Across the full range of rarity (5–10%), sample size (n=1000/5000), and complexity (p=3–10), RF consistently exhibits the highest accuracy and typically the best precision, with both metrics increasing as p and n rise. Conversely, recall and F1 present a differing case: LR (frequently in conjunction with SVM) generally exhibits superior performance, GB occupies a median position, and RF demonstrates the least sensitivity. Moving from 5% to 10% prevalence (i.e., less rare) and increasing n to 5000 lifted all metrics; the gains were larger for LR/GB on balanced metrics, while RF benefited most on ACC/PREC.

The real-data application complements these findings. On the wine dataset adjusted to 5% and 10% rarity and then balanced with SMOTETomek, RF achieved the top test ACC, and REC was highest for GB at 5% and tied for RF/SVM at 10%, confirming that tree-based ensembles can dominate accuracy-oriented objectives in practice even when their balanced performance in simulation is lower than LR's.

Metric-aware recommendation.

- Balanced minority–majority performance (equitable detection, operating-point robustness): LR remains a very strong baseline; GB is a practical alternative that often closes much of the gap.
- Overall accuracy or low false positives prioritized: RF is often the most effective choice, especially with balanced training via SMOTETomek and adequate model complexity.

In sum, the combination of extensive simulations and a real-world study shows when each method is preferable for rare-event classification. Rather than propose a single replacement for LR, we align metrics with models: use LR/GB for balanced detection objectives, and RF for accuracy-centric objectives. Coupled with SMOTETomek on the training split and strict test-set separation, this provides a practical framework for analysts confronting rarity across domains.

Conflicts of interest

There are no conflicts of interest in this work.

References

[1] Maalouf M., Trafalis T.B., Robust weighted kernel logistic regression in imbalanced and rare events data, *Computational Statistics & Data Analysis*, 55(1) (2011a) 168–183.

[2] Maalouf M., Trafalis T.B., Rare events and imbalanced datasets: an overview, *International Journal of Data Mining, Modelling and Management*, 3(4) (2011b) 375–388.

[3] Maalouf M., Homouz D., Trafalis T.B., Logistic regression in large rare events and imbalanced data: A performance comparison of prior correction and weighting methods, *Computational Intelligence*, 34(1) (2018) 161–174.

[4] Basha S.J., Madala S.R., Vivek K., Kumar E.S., Ammannamma T., A Review on Imbalanced Data Classification Techniques, *International Conference on Advanced Computing Technologies and Applications (ICACTA)*, India, (2022) 1-6.

[5] Ghorbani R., Ghousi R., Comparing different resampling methods in predicting students' performance using machine learning techniques, *IEEE Access*, 8 (2020) 67899-67911.

[6] Jonathan B., Putra P.H., Ruldeviyani Y., Observation imbalanced data text to predict users selling products on female daily with SMOTE, Tomek, and SMOTE-Tomek, *IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT)*, Indonesia, (2020) 81-85.

[7] Tariq M.A., Sargano A.B., Iftikhar M.A., Habib Z., Comparing Different Oversampling Methods in Predicting Multi-Class Educational Datasets Using Machine Learning Techniques, *Cybernetics and Information Technologies*, 23(4) (2023) 199-212.

[8] Stando A., Cavus M., Biecek P., The effect of balancing methods on model behavior in imbalanced classification problems, *In Fifth International Workshop on Learning with Imbalanced Domains: Theory and Applications*, Italy, (2024) 16-30.

[9] Seiffert C., Khoshgoftaar T.M., Van Hulse J., Napolitano A., Mining Data with Rare Events: A Case Study, *19th IEEE International Conference on Tools with Artificial Intelligence*, Greece, (2007) 132-139.

[10] Escobar C.A., Morales-Menendez R., Macias D., Process-monitoring-for-quality-a machine learning-based modeling for rare event detection, *Array*, 7 (2020) 100034.

[11] Lai S.B.S., Shahri N.H.N.B., Mohamad M.B., Rahman H.A.B.A., Rambli A.B., Comparing the performance of AdaBoost, XGBoost, and logistic regression for imbalanced data, *Mathematics and Statistics*, 9(3) (2021) 379-385.

[12] Meysami M., Kumar V., Pugh M., Lowery S.T., Sur S., Mondal S., Greene J.M., Utilizing logistic regression to compare risk factors in disease modeling with imbalanced data: a case study in vitamin D and cancer incidence, *Frontiers in Oncology*, 13 (2023) 1227842.

[13] Cervantes J., Garcia-Lamont F., Rodríguez-Mazahua L., Lopez A., A comprehensive survey on support vector machine classification: Applications, challenges and trends, *Neurocomputing*, 408 (2020) 189-215.

[14] Ling C., Lu Z., Support vector machine-based importance sampling for rare event estimation, *Structural and Multidisciplinary Optimization*, 63(4) (2021) 1609–1631.

[15] Chi H.M., Ersoy O.K., Support vector machine decision trees with rare event detection, *International Journal of Smart Engineering System Design*, 4(4) (2002) 225–242.

[16] Nayak M.A., Ghosh S., Prediction of extreme rainfall event using weather pattern recognition and support vector machine classifier, *Theoretical and Applied Climatology*, 114(3–4) (2013) 583–603.

[17] Kumar D., Thakur M., Dubey C.S., Shukla D.P., Landslide susceptibility mapping & prediction using support vector machine for Mandakini River Basin, Garhwal Himalaya, India, *Geomorphology*, 295 (2017) 115–125.

[18] Kalantar B., Pradhan B., Naghibi S.A., Motevalli A., Mansor S., Assessment of the effects of training data selection on the landslide susceptibility mapping: a comparison between support vector machine (SVM), logistic regression (LR) and artificial neural networks (ANN), *Geomatics Natural Hazards and Risk*, 9(1) (2018) 49–69.

[19] Nhu V.H., Shirzadi A., Shahabi H., Singh S.K., Al-Ansari N., Clague J.J., Jaafari A., Chen W., Miraki S., Dou J., Shallow landslide susceptibility mapping: a comparison between logistic model tree, logistic regression, naïve bayes tree, artificial neural network, and support vector machine algorithms, *International Journal of Environmental Research and Public Health*, 17(8) (2020) 2749.

[20] Zhu J., Li S., Song J., Magnitude Estimation for Earthquake Early Warning with Multiple Parameter Inputs and a Support Vector Machine, *Seismological Research Letters*, 93(1) (2021) 126–136.

[21] Breiman L., Random Forests, *Machine Learning*, 45(1) (2001) 5-32.

[22] Yan W., Application of Random Forest to Aircraft Engine Fault Diagnosis, *In the Proceedings of the Multiconference on Computational Engineering in Systems Applications*, 1 (2006) 468–475.

[23] Muchlinski D., Siroky D., He J., Kocher M., Comparing random forest with logistic regression for predicting class-imbalanced civil war onset data, *Political Analysis*, 24(1) (2016) 87–103.

[24] Hegelich S., Decision trees and random forests: Machine learning techniques to classify rare events, *European Policy Analysis*, 2(1) (2016) 98–120.

[25] Siders Z.A., Ducharme-Barth N.D., Carvalho F., Kobayashi D., Martin S., Raynor J., Jones T.T., Ahrens R.N.M., Ensemble Random Forests as a tool for modeling rare occurrences, *Endangered Species Research*, 43 (2020) 183–197.

[26] Lusa L., Gradient boosting for high-dimensional prediction of rare events, *Computational Statistics and Data Analysis*, 113 (2017) 19–37.

[27] Islam M.K., Hridi P., Hossain M.S., Narman H.S., Network anomaly detection using LightGBM: A gradient boosting classifier, *30th International Telecommunication Networks and Applications Conference (ITNAC)*, Australia, (2020) 1-7.

[28] Lyashevska O., Malone F., MacCarthy E., Fiehler J., Buhk J.H., Morris L., Class imbalance in gradient boosting classification algorithms: Application to experimental stroke data, *Statistical Methods in Medical Research*, 30(3) (2021) 916-925.

[29] Hairani H., Anggrawan A., Priyanto D., Improvement performance of the random forest method on unbalanced diabetes data classification using SMOTE-Tomek link, *JOIV: International Journal on Informatics Visualization*, 7(1) (2023) 258–266.

[30] Talukder M.A., Sharmin S., Uddin M.A., Islam M.M., Aryal S., MLSTL-WSN: Machine learning-based intrusion detection using SMOTETomek in WSNs, *International Journal of Information Security*, 23(3) (2024) 2139–2158.

[31] Musa A.B., Comparative study on classification performance between support vector machine and logistic

regression, *International Journal of Machine Learning and Cybernetics*, 4 (2013) 13-24.

[32] Kirasich K., Smith T., Sadler B., Random forest vs logistic regression: binary classification for heterogeneous datasets, *SMU Data Science Review*, 1(3) (2018) 9.

[33] Zhou Z.H., Machine Learning, *Springer Nature*, (2021).

[34] Vapnik V., Statistical Learning Theory, *John Wiley & Sons*, Chichester, (1998).

[35] Qiwen D., Xiaolong W., Lei L., Protein domain boundary prediction by combining support vector machine and domain guess by size algorithm, *High Technology Letters*, 13 (1994) 74-78.

[36] Rezvani S., Pourpanah F., Lim C.P., Wu Q.M., Methods for Class-Imbalanced Learning with Support Vector Machines: A Review and an Empirical Evaluation, *Soft Computing*, 28 (2024) 11873–11894.

[37] Gold C., Sollich P., Model selection for support vector machine classification, *Neurocomputing*, 55(1-2) (2003) 221-249.

[38] Scornet E., Biau G., Vert J.P., Consistency of random forests, *Annals of Statistics*, 43(4) (2015) 1716-1741.

[39] Kumar V., Evaluation of computationally intelligent techniques for breast cancer diagnosis, *Neural Computing and Applications*, 33(8) (2021) 3195-3208.

[40] Nazarenko E., Varkentin V., Polyakova T., Features of application of machine learning methods for classification of network traffic (features, advantages, disadvantages), *International Multi-Conference on Industrial Engineering and Modern Technologies (FarEastCon)*, Russian Federation, (2019) 1-5.

[41] Friedman J.H., Stochastic gradient boosting, *Computational Statistics & Data Analysis*, 38 (2002) 367–378.

[42] Friedman J.H., Greedy function approximation: a gradient boosting machine, *Annals of Statistics*, (2001) 1189-1232.

[43] Natekin A., Knoll A., Gradient boosting machines, a tutorial, *Frontiers in Neurorobotics*, 7 (2013) 21.

[44] Veropoulos K., Campbell C., Cristianini N., Controlling the sensitivity of support vector machines, *In Proceedings of the International Joint Conference on AI*, Stockholm, (1999) 55-60.

[45] Hoens T.R., Chawla N.V., Imbalanced datasets: from sampling to classifiers, *In Imbalanced Learning: Foundations, Algorithms, and Applications*, (2013) 43-59.

[46] Birla S., Kohli K., Dutta A., Machine learning on imbalanced data in credit risk, *IEEE 7th Annual Information Technology Electronics and Mobile Communication Conference (IEMCON)*, United States, (2016) 1-6.

[47] Tanimu J.J., Hamada M., Hassan M., Kakudi H., Abiodun J.O., A machine learning method for classification of cervical cancer, *Electronics*, 11(3) (2022) 463.

[48] Swana E.F., Doorsamy W., Bokoro P., Tomek link and SMOTE approaches for machine fault classification with an imbalanced dataset, *Sensors*, 22(9) (2022) 3246.

[49] Mandrekar J.N., Receiver operating characteristic curve in diagnostic test assessment, *Journal of Thoracic Oncology*, 5(9) (2010) 1315–1316.

[50] DeLong E.R., DeLong D.M., Clarke-Pearson D.L., Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach, *Biometrics*, (1988) 837–845.

[51] Oommen T., Baise L.G., Vogel R.M., Sampling bias and class imbalance in maximum-likelihood logistic regression, *Mathematical Geosciences*, 43(1) (2011) 99–120.

[52] Hajian-Tilaki K., Receiver operating characteristic (ROC) curve analysis for medical diagnostic test evaluation, *Caspian Journal of Internal Medicine*, 4(2) (2013) 627.

[53] Cortez P., Cerdeira A., Almeida F., Matos T., Reis J., Modeling wine preferences by data mining from physicochemical properties, *Decision Support Systems*, 47(4) (2009) 547-553.