A comparative study of over-sampling techniques as applied to seismic events

Mpho Mokoatle[1,2][0000−0001−9252−3914], Toshka Coleman[1][1111−2222−3333−4444], and Paul Mokilane[1][0000−0002−2649−8711]

[1] Council for Scientific and Industrial Research
Cluster: Next Generation Enterprises and Institutions, Data Science
https://www.csir.co.za/
[2] University of Pretoria, Pretoria, South Africa

Abstract. The likelihood that an earthquake will occur in a specific location, within a specific time frame, and with ground motion intensity greater than a specific threshold is known as a **seismic hazard**. Predicting these types of hazards is crucial since doing so can enable early warnings, which can lessen the negative effects. Research is currently being executed in the field of machine learning to predict seismic events based on previously recorded incidents. However, because these events happen so infrequently, this presents a class imbalance problem to the machine learning or deep learning learners. As a result, this study provided a comparison of the performance of popular over-sampling techniques that seek to even out class imbalance in seismic events data. Specifically, this work applied SMOTE, SMOTENC, SMOTEN, BorderlineSMOTE, SVMSMOTE, and ADASYN to an open source Seismic Bumps dataset then trained several machine learning classifiers with stratified K-fold cross-validation for seismic hazard detection. The SVMSMOTE algorithm was the best over-sampling method as it produced classifiers with the highest overall accuracy, F1 score, recall, and precision of 100%, respectively, whereas the ADASYN over-sampling methodology showed the lowest performance in all the reported metrices of all the models. To our understanding, no research has been done comparing the effectiveness of the aforementioned over-sampling techniques for tasks involving seismic events.

1 Introduction

Seismic hazards, such as earthquakes, pose significant risks to human life, infrastructure, and the environment. Accurate prediction of these events is crucial for implementing early warning systems and taking preventive measures to minimize their impact. Additionally, accurate prediction of these events informs crucial decisions on infrastructure development, land-use planning, and disaster preparedness, guiding the implementation of building codes and engineering practices to withstand earthquakes [3, 21].

Over the years, researchers have turned to machine learning techniques to forecast seismic events based on historical data. However, due to the rare oc-

currence of such events, imbalanced data presents a challenge to these machine-learning models. One specific challenge is the class imbalance problem, where the number of negative instances (non-seismic events) is significantly outnumbered by the number of positive instances (seismic events) [9]. This class imbalance can lead to biased models and limited predictive performance due to the scarcity of positive instances available for learning and generalization [11].

Several studies [4, 19, 31] have explored the application of machine learning techniques for seismic hazard prediction. Some of these studies have employed the publicly available Seismic Bumps dataset, which records energy readings and bump counts in a coal mine in Poland. Researchers have utilized various classifiers, including Naïve Bayes, Support Vector Machine (SVM), and neural networks, to tackle this prediction problem. While existing research has shown promising results in utilizing machine learning for seismic hazard prediction, one critical gap remains unaddressed. Little attention has been given to the comparison and evaluation of different over-sampling techniques as a potential solution to the class imbalance problem in this domain. Over-sampling methods create synthetic instances of the minority class or modify existing instances to balance the class distribution [6]. The comparative study proposed in this research aims to fill this gap by assessing the performance of popular over-sampling techniques applied to seismic events data. Specifically, this study applies SMOTE, SMOTENC, SMOTEN, BorderlineSMOTE, SVMSMOTE, and ADASYN (which are described in Section 2.2.) to the Seismic Bumps dataset. To evaluate the efficacy of these over-sampling approaches, various machine learning classifiers were trained using K-fold cross-validation. By analyzing the impact of these over-sampling methods, the study seeks to identify the most effective technique for handling the class imbalance and improving the performance of seismic event prediction models, helping researchers and practitioners in the field to make informed decisions when building predictive models for rare events like seismic hazard.

2 Related work

2.1 Review of existing research studies focused on seismic hazard prediction using machine learning

Numerous research studies have delved into the application of machine learning techniques for detecting seismic hazards, with several of them utilizing the Seismic Bumps dataset employed in our own study. A noteworthy paper [19] uses the dataset and proposed a new approach that incorporates negation handling in the Naïve Bayes classifier to improve accuracy. Experimental results show that the proposed approach achieves a higher accuracy of 76.98% compared to the traditional Naïve Bayes classifier without negation handling (64.5%) and the native MATLAB Naïve Bayes classifier without negation handling (65.09%). Another research study [31] using the dataset developed a prediction model for detecting periods of increased seismic activity that pose a threat to miners in coal mines. Various classification models, including Random Forest, Naïve Bayes Classifier,

Logistic Regression, and Support Vector Machine (SVM), are applied to evaluate their performance. The experimental results demonstrate that the Random Forest and SVM models achieve the highest accuracy, with 92.2%, respectively, for the Seismic Bumps dataset.

Additionally, this dataset was used in another study [4] where the authors proposed two deep temporal convolution neural network (CNN) models: dilated causal temporal convolution network (DCTCNN) and CNN long short-term memory hybrid model (CNN-LSTM). DCTCNN utilizes dilated CNN kernels, a causal strategy, and residual connections, while CNN-LSTM combines the advantages of CNN and LSTM. Both models are designed to extract long-term historical features from monitoring seismic data. The proposed models were evaluated using two real-life coal mine seismic datasets and compared with a traditional time series prediction method, two classic machine learning algorithms, and two standard deep learning networks. The results demonstrate that DCTCNN and CNN-LSTM outperform the other algorithms, indicating their effectiveness in completing the seismic prediction task. However, the issue of class imbalance is overlooked in these studies [4, 19, 31] and not addressed through any oversampling, undersampling or resampling methods, resulting in models that may not adequately capture the characteristics of seismic events. Furthermore, different methods for over-sampling to address this have not yet been compared in tasks predicting seismic hazards. The study by [15] employed resampling for the Seismic Bumps dataset, but chose a single resampling method without comparing it against other possible methods for addressing its class imbalance. Previous studies have demonstrated the shortcomings of traditional machine learning algorithms when faced with imbalanced datasets, including decreased accuracy, sensitivity, and precision in detecting seismic events.

2.2 Overview of over-sampling techniques for class imbalance

Over-sampling techniques have gained attention as a potential solution to address the class imbalance problem [6,16]. These techniques aim to re-balance the class distribution by generating synthetic instances of the minority class (seismic events) or modifying existing instances to amplify their representation. By artificially increasing the number of positive instances, over-sampling methods aim to provide a more balanced training dataset, enabling machine learning models to better capture the characteristics of seismic events [6]. We provide descriptions below of the over-sampling techniques used in this study.

SMOTE: Synthetic Minority Over-sampling Technique SMOTE is a popular over-sampling approach introduced in 2002. It works by over-sampling the minority class through creating synthetic instances rather than over-sampling with replacement. The SMOTE algorithm works by first identifying a minority class sample and choosing five nearest neighbors of the minority class sample. Then, for example, if the amount of over-sampling that is required is 200%, then two nearest neighbors will be selected from the initial five nearest neighbors.

Finally, the synthetic instance is generated by drawing a line between the two nearest neighbors and the initial minority class sample [2, 14].

BorderlineSMOTE The BorderlineSMOTE algorithm is a variant of the SMOTE algorithm and seeks to address the shortcomings of SMOTE. For instance, if there are minority class samples that are outliers and appears in the majority class, a synthetic instance will be created using the samples of the majority class. BorderlineSMOTE alleviates this problem by classifying any minority sample as noise if all its neighbors are the majority class samples. The noise minority samples will be ignored when creating new synthetic instances. Additionally, the BorderlineSMOTE algorithm classifiers a few samples as border points that majority and minority samples as neighbors and creates synthetic instances completely from these border points [6, 25, 28].

Adaptive Synthetic (ADASYN) ADASYN is a universal over-sampling method. For each of the minority samples it first determines the impurity of the neighbourhood by taking the ratio of the majority samples in the neighbourhood and k. The higher the ratio, the more synthetic examples will be created for that instance [7].

Synthetic Minority Over-sampling TEchnique-Nominal Continuous (SMOTENC) Since the dataset used in this study is a combination of continuous and categorical features, a SMOTE variant named SMOTENC was applied as it is known to generalize well across mixed-data. SMOTENC's algorithm involves the computation of the median and the nearest neighbors and also populates the synthetic instance using the same approach as in SMOTE [2, 8].

SVMSMOTE The primary distinction between SVMSMOTE and other SMOTE is that the technique would use the Support Vector Machine (SVM) algorithm to determine the mis-classification in the Borderline-SMOTE rather than K-nearest neighbours [20].

2.3 Previous applications of over-sampling techniques for class imbalance

Several studies have investigated the effectiveness of over-sampling techniques. According to some studies, the inclusion of over-sampling techniques has shown to achieve better True Positive (TP) rates and improve model performance [10]. One study introduced two new minority over-sampling methods, Borderline-SMOTE1 and Borderline-SMOTE2, which focused on over-sampling only the minority examples near the borderline. The experiments showed that these approaches achieved better True Positive (TP) rate and F-value compared to SMOTE and random over-sampling methods for the minority class [6]. Furthermore, in the prediction of environmental complaints related to construction

projects, an over-sampling-based method was developed using imbalanced empirical data. The method involved over-sampling techniques combined with machine learning algorithms to predict complaints due to environmental pollutants. The study reported performance improvements ranging from 8% to 23% using the over-sampling-based method compared to non-over-sampling approaches [32]. In the domain of Medical Artificial Intelligence, SMOTE was also used in a study [30] to address class imbalance in detecting COVID-19 cases. The authors used SMOTE to generate synthetic samples of the minority class (COVID-19 cases) and improve classifier performance. This study demonstrated the effectiveness of SMOTE in the specific domain of COVID-19 detection. Another research study [18] focused on determining the effective rate of minority class over-sampling using five over-sampling methods, including SMOTE, SVMSMOTE, and BorderlineSMOTE. The study aimed to maximize the performance of machine learning models and found that different datasets required different effective over-sampling rates. Additionally, a comparative analysis examined the performance of SMOTE and ADASYN, concluding that both techniques were effective in handling class imbalance. However, the performance varied depending on the classifier and the minority class [24]. Drawing from these two studies, we note that the effectiveness of these over-sampling techniques can vary depending on dataset characteristics, imbalance severity, and the choice of machine learning algorithms. Therefore, conducting experiments and evaluating these techniques within the context of specific applications is crucial to determine the most suitable approach.

In the context of seismic hazard detection, the application of over-sampling techniques to address class imbalance is relatively limited. Limited studies have explored the use of these techniques specifically for seismic event prediction. However, insights gained from their application in other domains highlight the potential benefits of employing over-sampling techniques to improve the performance of seismic hazard detection models. Thus, the objective of this study is to provide a comparative assessment of over-sampling methods. Specifically, this study applies SMOTE, SMOTENC, SMOTEN, BorderlineSMOTE, SVMSMOTE, and ADASYN to the Seismic Bumps dataset.

3 Materials and methods

The dataset, the pre-processing procedures, the over-sampling methods, as well as an overview of the performance indicators used to assess the efficacy of the various over-sampling methods are all described in this section (Fig. 1).

3.1 Data description and pre-processing

In this study, the Seismic Bumps dataset from Kaggle, which is available to the public, was used. This dataset, made up of 2584 observations and 19 columns, contains energy readings and bump counts that were captured during the preceding shifts in a coal mine in Poland. All input features were used in this

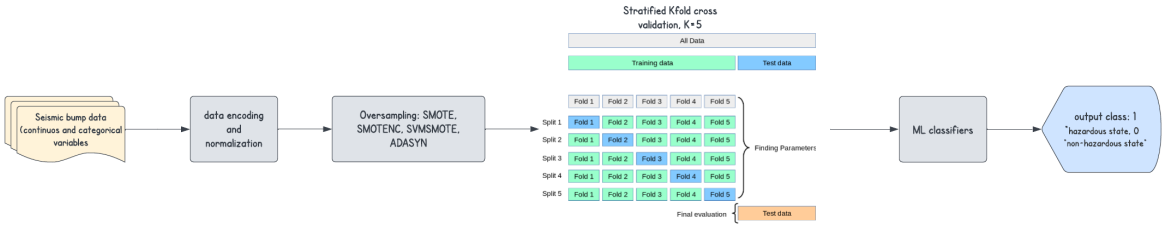


Fig. 1: This figure illustrates the strategy taken in this study to evaluate the effectiveness of various over-sampling strategies.

work, with the exception of "nbumps6," "nbumps7," and "nbumps86," which exclusively contained zero values. The omission of these features was crucial to prevent their conversion to null during normalization. The target class is a binary variable where "1" indicates a high energy seismic bump occurred in the following shift (a "hazardous state"), and "0" indicates no high energy seismic bumps in the following shift (a "non-hazardous state"). This target variable had 2414 *non-hazardous state* and only 170 *hazardous state*, which posed a significant class-imbalance problem.

For data processing, the scikit-learn LabelEncoder was used to encode the non-numeric variables then, a data normalization procedure (min-max scaling) was performed to transform the numeric columns to a standard scale to ensure that the features with high values do not dominate the learning process.

3.2 Machine learning models

To compare the effectiveness of the different over-sampling methods, seven different machine learning models from the scikit-learn library [22] were used: KNeighborsClassifier, DecisionTreeClassifier, RandomForestClassifier, AdaBoostClassifier, MLPClassifier, GaussianNB, and LogisticRegression. Table 1 provides a description of all the models used.

3.3 Stratified K-Fold cross validation

This study evaluated the classification performance of the models with the over-sampling techniques after performing 5-fold stratified cross-validation. Stratified K-Fold cross validation is a variation of the standard K-Fold cross-validation. As opposed to splitting the data at random, stratified k-fold cross-validation ensures that the ratio of the target classes is the same in each fold as it is across the entire dataset. This method is particularly useful in small datasets with class imbalance problems [23, 33].

Table 1: This table gives a description of the machine learning classifiers

ML model	Description
KNeighbors	This algorithm makes predictions on new data by assuming that data with similar characteristics cluster together [5]. In this work, the default metric *minkowski* [29] was used to compute the distance.
DecisionTree	This algorithm is applicable to both classification and regression. Its structure resembles a tree and is composed of leaf nodes, internal nodes, branches, and root nodes [12]. In this work, the split's quality was assessed using the *gini* default criterion.
RandomForest	A type of bagging estimator that fits several decision trees on various sub-samples of the data then uses averaging to increase the predictive performance [26]. As with the decision tree, in this work, the *gini* criterion was used to measure the quality of a split.
AdaBoost	AdaBoost is a boosting technique that trains classifiers sequentially so as to minimise the errors produced by earlier learners [27].
MLP	Multi-layer perceptrons (MLPs), are a sort of feed-forward networks in which data is only sent in one direction. This model was trained with 100 neurons, ReLU activation function, and Adam optimizer [13] .
GaussianNB	Gaussian Naive Bayes (NB) is a classification algorithm based on the probabilistic method and Gaussian distribution. GaussianNB assumes that each independent variable has an independent capacity of predicting the target class [17].
LogisticRegression	An algorithm that uses a number of independent variablesto predict a binary target class [1].

3.4 Performance measures

Accuracy, precision, recall, and F1-score measurements were used to assess how well the over-sampling techniques worked in conjunction with the machine learning classifiers. These metrics are all defined below in terms of true positives (TP), true negatives (TN), false negatives (FN), and false positives (FP). The mean and standard deviation are reported for each metric.

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$precision = \frac{TP}{TP + FP} \tag{2}$$

$$recall = \frac{TP}{TP + FN} \tag{3}$$

$$F1score = \frac{2TP}{2TP + FP + FN} \tag{4}$$

4 Experimental results

The results of all the models are shown (Table 2, 3, 4,5) in terms of their average accuracy, F1-score, recall and precision over 5 k-folds.

When no over-sampling techniques were used on the data, the machine learning models' accuracy was quite good. The confusion matrices, however, showed a considerable bias towards the minority class and some models only predicted one class label (Table 2 and Fig. 2, 3, 4).

The ADASYN over-sampling approach produced the least accurate classifiers (49-65%), while the SVMSMOTE over-sampling method produced the best classifiers with an average accuracy of 98-100% and F1 score of 98-100% (Table 3, Fig. 5).

Table 2: Mean accuracy

	KNeighbors	DecisionTree	RandomForest	AdaBoost	MLP	GaussianNB	LogisticRegression
No over-sampling	0.93 ± 0.003	0.81 ± 0.07	0.92 ± 0.01	0.83 ± 0.11	0.93 ± 0.003	0.10 ± 0.02	0.93 ± 0.003
ADASYN	0.52 ± 0.03	0.59 ± 0.04	0.65 ± 0.04	0.62 ± 0.05	0.49 ± 0.11	0.49 ± 0.03	0.52 ± 0.12
BorderlineSMOTE	0.83 ± 0.05	0.86 ± 0.06	0.89 ± 0.05	0.82 ± 0.06	0.84 ± 0.08	0.84 ± 0.07	0.83 ± 0.07
SMOTE	0.75 ± 0.05	0.75 ± 0.06	0.82 ± 0.07	0.78 ± 0.06	0.77 ± 0.06	0.78 ± 0.05	0.75 ± 0.06
SVMSMOTE	**0.98 ± 0.01**	**1. 0 ± 0.00**	**1.0 ± 0.00**	**1.0 ± 0.00**	**0.99 ± 0.01**	**0.99 ± 0.00**	**0.99 ± 0.01**
SMOTENC	0.76 ± 0.02	0.81 ± 0.04	0.89 ± 0.03	0.81 ± 0.04	0.80 ± 0.02	0.56 ± 0.09	0.71 ± 0.02

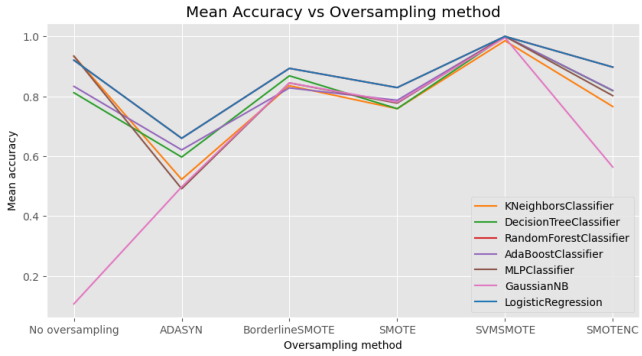
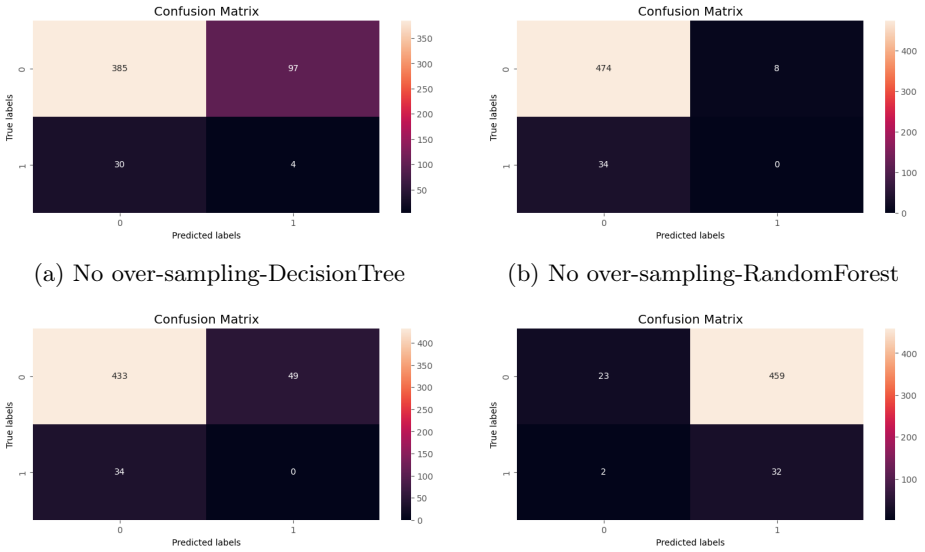


Fig. 2: Mean accuracy vs over-sampling method

Table 3: Mean F1 score

	KNeighbors	DecisionTree	RandomForest	AdaBoost	MLP	GaussianNB	LogisticRegression
No over-sampling	0.0 ± 0.0	0.10 ± 0.08	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.12 ± 0.01	0.0 ± 0.0
ADASYN	0.55 ± 0.07	0.60 ± 0.04	0.64 ± 0.03	0.63 ± 0.03	0.45 ± 0.07	0.64 ± 0.03	0.54 ± 0.12
BorderlineSMOTE	0.83 ± 0.04	0.86 ± 0.05	0.88 ± 0.04	0.83 ± 0.05	0.84 ± 0.06	0.84 ± 0.06	0.83 ± 0.06
SMOTE	0.76 ± 0.03	0.76 ± 0.04	0.84 ± 0.04	0.80 ± 0.04	0.77 ± 0.04	0.77 ± 0.03	0.76 ± 0.03
SVMSMOTE	**0.98 ± 0.01**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.99 ± 0.01**
SMOTENC	0.76 ± 0.02	0.83 ± 0.03	0.89 ± 0.04	0.82 ± 0.03	0.79 ± 0.03	0.69 ± 0.04	0.71 ± 0.02



(a) No over-sampling-DecisionTree

(b) No over-sampling-RandomForest

(c) No over-sampling-AdaBoost

(d) No over-sampling-GaussianNB

Fig. 3: Machine learning models' confusion matrices without the use of an over-sampling technique

Table 4: Mean recall

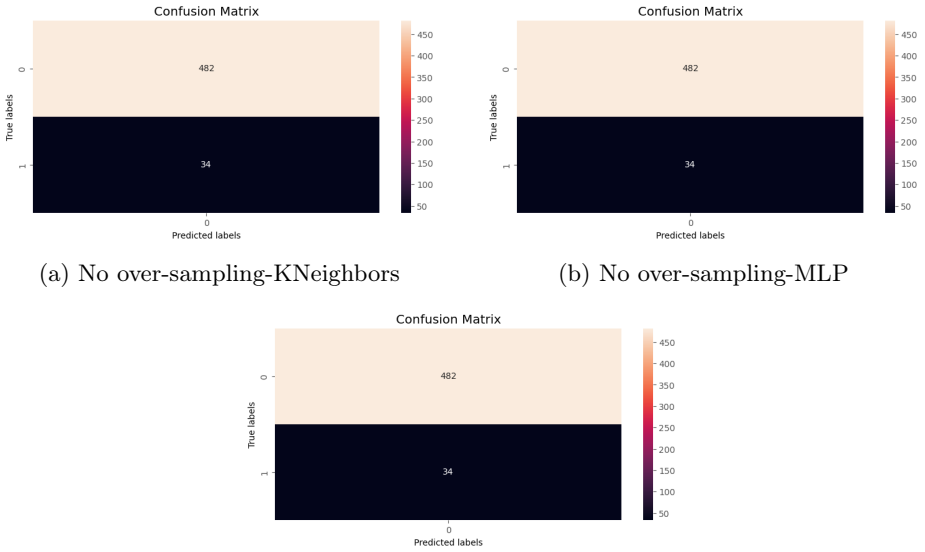
	KNeighbors	DecisionTree	RandomForest	AdaBoost	MLP	GaussianNB	LogisticRegression
No over-sampling	0.0 ± 0.0	0.11 ± 0.06	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.93 ± 0.07	0.0 ± 0.0
ADASYN	0.62 ± 0.16	0.62 ± 0.09	0.66 ± 0.09	0.66 ± 0.06	0.43 ± 0.07	0.93 ± 0.08	0.60 ± 0.19
BorderlineSMOTE	0.84 ± 0.04	0.91 ± 0.02	0.91 ± 0.03	0.86 ± 0.02	0.83 ± 0.02	0.81 ± 0.04	0.83 ± 0.04
SMOTE	0.78 ± 0.03	0.86 ± 0.04	0.87 ± 0.02	0.83 ± 0.03	0.75 ± 0.06	0.72 ± 0.06	0.75 ± 0.05
SVMSMOTE	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1.0 ± 0.0**
SMOTENC	0.74 ± 0.05	0.86 ± 0.03	0.91 ± 0.02	0.84 ± 0.03	0.77 ± 0.06	0.99 ± 0.01	0.69 ± 0.05

Since the F1 scores were inadequate when no over-sampling technique was used, as was expected, the average recall scores and precision were similarly quite poor as well when no over-sampling technique was applied to the data. Additionally, across all over-sampling techniques, SMVSMOTE still maintained the highest average recall and precision scores of 100% respectively, whereas ADASYN had the lowest recall and precision (Table 4, 5, and Fig. 6, 7).

The confusion matrices of the best performing over-sampling method is shown (Fig. 8).

5 Discussion

This study presented a seismic hazard prediction problem using machine learning. More often than not, the machine learning classifiers face a class imbalance problem as a result of how rarely these types of hazards occur. In an effort to ex-



(a) No over-sampling-KNeighbors

(b) No over-sampling-MLP

(c) No over-sampling-LogisticRegression

Fig. 4: Confusion matrices of the models that predicted a single class label when no over-sampling technique was used

Table 5: Mean precision

	KNeighbors	DecisionTree	RandomForest	AdaBoost	MLP	GaussianNB	LogisticRegression
No over-sampling	0.0 ± 0.0	0.04 ± 0.04	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.06 ± 0.01	0.0 ± 0.0
ADASYN	0.51 ± 0.03	0.59 ± 0.04	0.68 ± 0.06	0.61 ± 0.07	0.51 ± 0.14	0.49 ± 0.02	0.52 ± 0.09
BorderlineSMOTE	0.84 ± 0.09	0.82 ± 0.08	0.87 ± 0.07	0.81 ± 0.10	0.88 ± 0.11	0.87 ± 0.10	0.84 ± 0.11
SMOTE	0.76 ± 0.08	0.72 ± 0.06	0.81 ± 0.10	0.78 ± 0.11	0.80 ± 0.12	0.84 ± 0.11	0.78 ± 0.10
SVMSMOTE	**0.97 ± 0.01**	**1.0 ± 0.0**	**1.0 ± 0.0**	**1. 0 ± 0.0**	**0.99 ± 0.01**	**0.98 ± 0.0**	**0.98 ± 0.02**
SMOTENC	0.78 ± 0.05	0.79 ± 0.07	0.87 ± 0.06	0.81 ± 0.07	0.84 ± 0.07	0.54 ± 0.06	0.73 ± 0.05

amine strategies for addressing class inequality, this study proposed to evaluate the influence of five over-sampling techniques—ADASYN, BorderlineSMOTE, SMOTE, SVMSMOTE, SMOTENC, and no over-sampling. Considering that the dataset was too small and we did not want to lose any information, which would have impacted the accuracy of our models, over-sampling was chosen rather than under-sampling.

To assess the effectiveness of the aforementioned over-sampling methods, this work trained several machine learning techniques, including KNeighbors, DecisionTree, RandomForest, AdaBoost, MLP, GaussianNB, and LogisticRegression. Then, the average accuracy, F1 score, recall, and precision for each classifier and over-sampling technique were reported.

In the first run, all seven models were tested using a dataset that had not been over-sampled and several observations were made: all the learning algorithms—aside from the GaussianNB model—returned very high accuracies. The

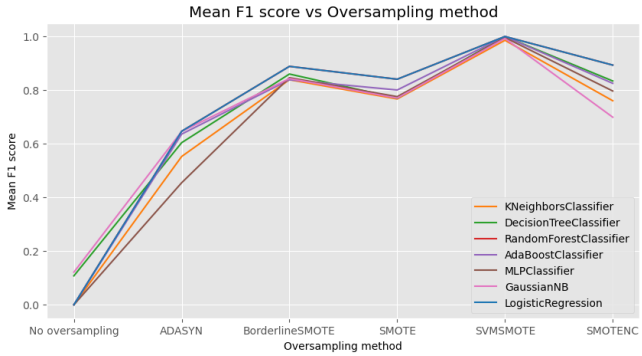


Fig. 5: Mean F1 vs over-sampling method

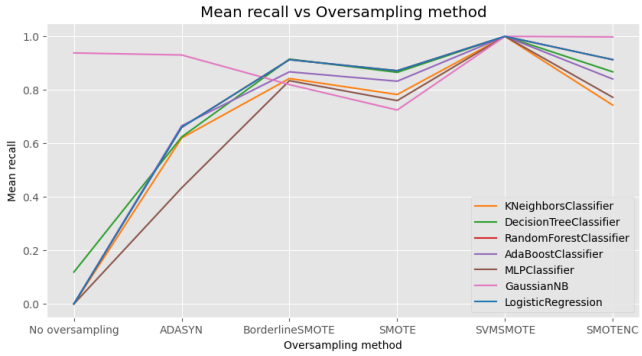


Fig. 6: Mean recall vs over-sampling method

F1 scores, recall, and precision scores, however, were incredibly poor. This was an apparent indication that the class imbalance problem posed a significant issue.

In the second run of the experiments, the dataset was over-sampled with the above over-sampling techniques. The SVMSMOTE algorithm was the best over-sampling method as it provided the highest overall accuracy, F1 score, recall and precision whereas the ADASYN over-sampling methodology showed the lowest performance in all the reported metrics of all the models.

Although using different methodologies, the dataset used in this work has previously been applied in prior studies [4, 15, 31] and some of the observations made in this work are in agreement with those in the literature.

For example, prior to over-sampling, the accuracy in [15] was consistently good, but the F1 scores, precision, and recall were subpar. Similar to our work, the performance metrics only increased after accounting for class imbalance. The sole significant distinction between this work [15] and ours is that only one sampling strategy was suggested and put to the test while our study examined several other strategies.

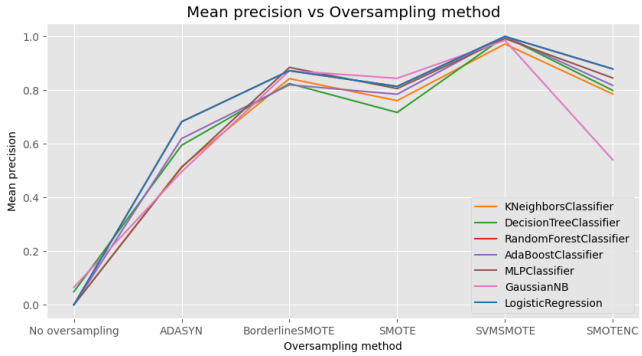
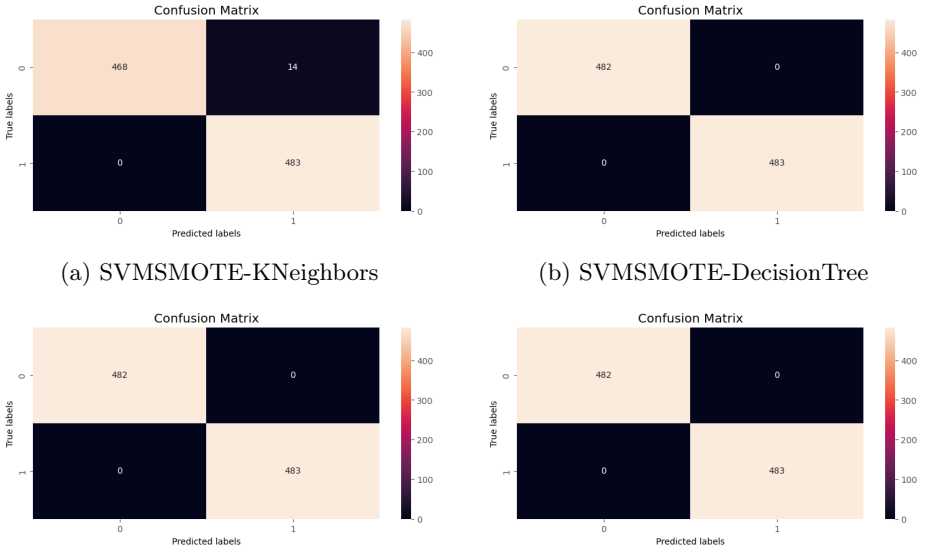


Fig. 7: Mean precision vs over-sampling method



(a) SVMSMOTE-KNeighbors

(b) SVMSMOTE-DecisionTree

(c) SVMSMOTE-RandomForest

(d) SVMSMOTE-AdaBoost

Fig. 8: Confusion matrices of some the best over-sampling technique: SVMSMOTE

A study [31] also used the dataset that was used in this work and a key difference between our work and theirs is that the models were biased towards a single class as the authors did not account for class imbalance.

In another paper, the authors [4] employed the same dataset as in our study, but used a different response variable to describe the seismic hazard problem. For instance, the authors of this work employed the *energy* variable as the response variable, which signifies the total energy of seismic bumps reported within a

previous shift, whereas our study used the categorical variable *hazardous state/ or non hazardous* state as the response variable. As a result, this problem was automatically transformed into a regression problem rather than being modelled as a classification problem.

6 Conclusion

Predicting seismic hazards is essential for reducing the negative effects of earthquakes, such as casualties and property loss. Early warning systems can be created, enabling prompt evacuation and disaster response preparation, by precisely predicting seismic hazards. However, machine learning or deep learning models face a class imbalance issue because seismic events occur infrequently. This study therefore offered a comparative evaluation of widely used over-sampling techniques for the problem of seismic events. This study discovered that classifiers for seismic hazard can perform noticeably better when machine learning techniques incorporating SVMSMOTE are used.

References

1. Bisong, E., Bisong, E.: Logistic regression. Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners pp. 243–250 (2019)
2. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: Smote: synthetic minority over-sampling technique. Journal of artificial intelligence research **16**, 321–357 (2002)
3. Cutter, S.L.: Vulnerability to environmental hazards. Progress in human geography **20**(4), 529–539 (1996)
4. Geng, Y., Su, L., Jia, Y., Han, C., et al.: Seismic events prediction using deep temporal convolution networks. Journal of Electrical and Computer Engineering **2019** (2019)
5. Guo, G., Wang, H., Bell, D., Bi, Y., Greer, K.: Knn model-based approach in classification. In: On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE: OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2003, Catania, Sicily, Italy, November 3-7, 2003. Proceedings. pp. 986–996. Springer (2003)
6. Han, H., Wang, W.Y., Mao, B.H.: Borderline-smote: a new over-sampling method in imbalanced data sets learning. In: International conference on intelligent computing. pp. 878–887. Springer (2005)
7. He, H., Bai, Y., Garcia, E.A., Li, S.: Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In: 2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence). pp. 1322–1328. Ieee (2008)
8. Islahulhaq, W.W., Ratih, I.D.: Classification of non-performing financing using logistic regression and synthetic minority over-sampling technique-nominal continuous (smote-nc). Int. J. Adv. Soft Comput. Appl **13**, 115–128 (2021)
9. Japkowicz, N., Stephen, S.: The class imbalance problem: A systematic study. Intelligent data analysis **6**(5), 429–449 (2002)

10. Kalaycioglu, O., Akhanli, S.E., Mentese, E.Y., Kalaycioglu, M., Kalaycioglu, S.: Using machine learning algorithms to identify predictors of social vulnerability in the event of an earthquake: Istanbul case study. Natural Hazards and Earth System Sciences Discussions **2022**, 1–32 (2022)
11. Kiani, J., Camp, C., Pezeshk, S.: On the application of machine learning techniques to derive seismic fragility curves. Computers Structures **218**, 108–122 (2019). https://doi.org/https://doi.org/10.1016/j.compstruc.2019.03.004, https://www.sciencedirect.com/science/article/pii/S0045794918318650
12. Kotsiantis, S.B.: Decision trees: a recent overview. Artificial Intelligence Review **39**, 261–283 (2013)
13. Kruse, R., Mostaghim, S., Borgelt, C., Braune, C., Steinbrecher, M.: Multi-layer perceptrons. In: Computational intelligence: a methodological introduction, pp. 53–124. Springer (2022)
14. Maldonado, S., López, J., Vairetti, C.: An alternative smote oversampling strategy for high-dimensional datasets. Applied Soft Computing **76**, 380–389 (2019)
15. Menon, A.P., Varghese, A., Joseph, J.P., Sajan, J., Francis, N.: Performance analysis of different classifiers for earthquake prediction: Pace. IJIRT (2), 142–146 (2020)
16. Mohammed, R., Rawashdeh, J., Abdullah, M.: Machine learning with oversampling and undersampling techniques: overview study and experimental results. In: 2020 11th international conference on information and communication systems (ICICS). pp. 243–248. IEEE (2020)
17. Mugdha, S.B.S., Kuddus, M.B.M.M., Salsabil, L., Anika, A., Marma, P.P., Hossain, Z., Shatabda, S.: A gaussian naive bayesian classifier for fake news detection in bengali. In: Emerging Technologies in Data Mining and Information Security: Proceedings of IEMIS 2020, Volume 2. pp. 283–291. Springer (2021)
18. Naim, F.A., Hannan, U.H., Humayun Kabir, M.: Effective rate of minority class over-sampling for maximizing the imbalanced dataset model performance. In: Proceedings of Data Analytics and Management: ICDAM 2021, Volume 2. pp. 9–20. Springer (2022)
19. Netti, K., Radhika, Y.: An efficient naïve bayes classifier with negation handling for seismic hazard prediction. In: 2016 10th International Conference on Intelligent Systems and Control (ISCO). pp. 1–4. IEEE (2016)
20. Nguyen, H.M., Cooper, E.W., Kamei, K.: Borderline over-sampling for imbalanced data classification. International Journal of Knowledge Engineering and Soft Data Paradigms **3**(1), 4–21 (2011)
21. Nicolis, O., Plaza, F., Salas, R.: Prediction of intensity and location of seismic events using deep learning. Spatial Statistics **42**, 100442 (2021)
22. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research **12**, 2825–2830 (2011)
23. Prusty, S., Patnaik, S., Dash, S.K.: Skcv: Stratified k-fold cross-validation on ml classifiers for predicting cervical cancer. Frontiers in Nanotechnology **4**, 972421 (2022)
24. Rachburee, N., Punlumjeak, W.: Oversampling technique in student performance classification from engineering course. International Journal of Electrical and Computer Engineering **11**(4), 3567 (2021)
25. Revathi, M., Ramyachitra, D.: A modified borderline smote with noise reduction in imbalanced datasets. Wireless Personal Communications **121**, 1659–1680 (2021)
26. Rigatti, S.J.: Random forest. Journal of Insurance Medicine **47**(1), 31–39 (2017)

27. Schapire, R.E.: Explaining adaboost. In: Empirical Inference: Festschrift in Honor of Vladimir N. Vapnik, pp. 37–52. Springer (2013)
28. Shen, W., Fan, W., Chen, C.: An electric vehicle charging pile fault diagnosis system using borderline-smote and lightgbm. In: Tenth International Symposium on Precision Mechanical Measurements. vol. 12059, pp. 615–622. SPIE (2021)
29. Singh, A., Yadav, A., Rana, A.: K-means with three different distance metrics. International Journal of Computer Applications **67**(10) (2013)
30. Turlapati, V.P.K., Prusty, M.R.: Outlier-smote: A refined oversampling technique for improved detection of covid-19. Intelligence-based medicine **3**, 100023 (2020)
31. Verma, L.K., Kishore, N., Jharia, D.: Predicting dangerous seismic events in active coal mines through data mining. International Journal of Applied Engineering Research **12**(5), 567–571 (2017)
32. Wang, D., Liang, Y., Yang, X., Dong, H., Tan, C.: A safe zone smote oversampling algorithm used in earthquake prediction based on extreme imbalanced precursor data. International Journal of Pattern Recognition and Artificial Intelligence **35**(13), 2155013 (2021)
33. Widodo, S., Brawijaya, H., Samudi, S.: Stratified k-fold cross validation optimization on machine learning for prediction. Sinkron: jurnal dan penelitian teknik informatika **7**(4), 2407–2414 (2022)