Proceedings of 12th IOE Graduate Conference
Peer Reviewed
ISSN: 2350-8914 (Online), 2350-8906 (Print)
Year: 2022 Month: October Volume: 12

Earthquake Affected Buildings Damage Classification Using Machine Learning

Sobit Regmi [a], Babu Ram Dawadi [b]

[a, b] *Department of Electronics & Computer Engineering, Pulchowk Campus, IOE, Tribhuvan University, Nepal*
[a] regmi.sobit@gmail.com, [b] baburd@ioe.edu.np

Abstract
The ability to assess the impact of a seismic event on buildings are very crucial for quick recovery and emergency response planning. Several classical methods previously existed to assess such damaged buildings take significant time and resources. This study investigates the effectiveness of using machine learning algorithms in the vulnerability prediction of buildings. Three machine learning classifiers knn, decision tree, and xgboost have been utilized in this research to predict the damage grade of the building using a dataset from the Nepal earthquake that has several building's specific characteristics like (e.g., age, number of floor area, ground floor type, etc.). A portion of data from this dataset is used to evaluate the model on unseen data and rest of the dataset is used to train and select the best-performing model using stratified 5 fold validation methods. The investigation of this research illustrates that the xgboost can accurately predict the damage grade with 74.4 % accuracy in the test dataset. Furthermore, this paper suggests that the accuracy of the model can be increased even more provided that there are balanced and huge datasets.

Keywords
Earthquake, Building Damage, Classification, Machine Learning

1. Introduction

Although the occurrence of earthquakes is less frequent, they contribute significantly to physical and social damage. Situational awareness is an important part of the decision-making process for facility owners, users, emergency responders, and local and state officials. A lack of such situational awareness can lead to a catastrophic societal response. Similarly, there is a rapid increase in insurance companies in global industries, and they play a critical role in today's economy. In Nepal as well, companies like "Sagarmatha Insurance", "Standard chartered" along with others insure the house and the variables within against disasters. But it is widely reported that there is an enormous problem of fake claims (false positives) So, predicting the earthquake damages is crucial not only for immediate post-disaster recovery and emergency response planning but also to decrease the case of fake claims in property insurance.

Although a holistic view of the impact of earthquakes on structures can be obtained from the post-earthquake inspection process with the help of volunteers by inspecting and tagging each building, it is a time-consuming process and resource intensive.With the rise in available data and computer resources, there is a rapid evolution of the application of machine learning over recent years with substantial promise in various disciplines. With the power of learning complex nonlinear functions, and treating uncertainties, machine learning can help society to facilitate decision-making in studying the seismic effect on structural buildings as well. In this research, we will be modeling the damages to the building from the Gorkha Earthquake in April 2015 which killed over 9000 people using state-of-the-art machine learning techniques. The same model can be used in the future for emergency response planning and quick post-recovery provided the required features for the model. The main aim of this research is to forecast the damage to the individual house as an integer variable between one and three given the information about its location, secondary usage, and the materials used to build the house where 0 represents low damage, 1 represents a medium account of damage and 2 represents almost complete destruction. To achieve those, we would be using different machine learning algorithms and comparing their results. The final proposed classifier will be able to predict the damage provided by the basic aspect of the building with good accuracy.

2. Related Works

The study of damage to buildings began a long time ago. [1] presented a consistent method for earthquake intensity classification based on the theory of statistical pattern recognition and developed a discriminative function for such identifications based on the Bayesian criterion. [2] used the application of fuzzy logic to earthquake damage predictions. These classical methods for estimating damage require a lot of information on building and earthquake ground motion, which is costly and time-consuming. Although there is rapid progress in AI tools and their application there are only a few research on their application in rapid seismic assessment. Riedel et al.[3] researched the application of support vector machines for earthquake assessment at urban or regional scales. Similarly, [4] worked on assessing the seismic impact on the earthquake damage data portfolio of 2014 South Napa earthquake using machine learning techniques. They concluded the use of AI provides a reliable estimate of the earthquake-induced potential building damage and indicated that the random forest algorithm provides the best performance among the evaluated techniques [5] used a Neural network and random forest for predicting damage to the buildings caused by the earthquake which is almost the earliest application of machine learning in this type of application. [6] used Generative Adversarial Networks to classify structural damage caused by earthquakes. [7] tried four algorithms to develop a damage prediction model from 340 post-earthquake buildings in the Mexico City and achieved more than 65 percent prediction accuracy. Similarly, [8] used machine learning models and aerial photographs to classify buildings in the Kumamoto earthquake into four damage levels. [9] used the XGBoost classifier to model the earthquake damage by formulating the problem as multi class classification models. [10] worked on deep learning methods with focus on CNN to detect earthquake damage in building with the optical data obtained from sensing satellite. In this research, we will be applying all these works in addition to other methods on much larger datasets, trying to achieve higher prediction accuracy.

3. Methodology

3.1 Dataset Descriptions

The dataset in this research was collected by Kathmandu Living Labs and the Central Bureau of Statistics to identify beneficiaries for assistance from the government and was further processed by [11] . It has 260601 rows, 39 features, and a target variable that represents the damage grade where the building id column is a unique and random identifier.

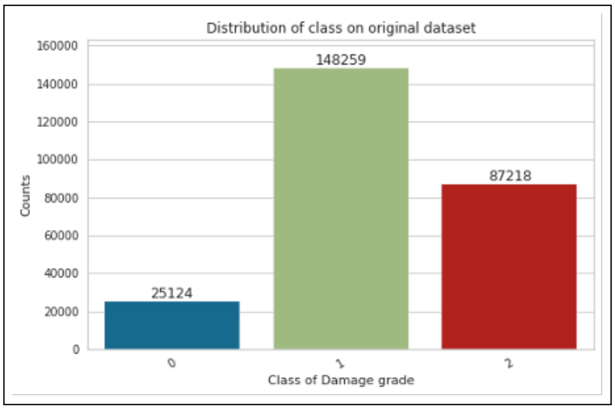

Figure 1: Distribution of class on original dataset

Following features were extracted from the dataset in this research study.

3.1.1 Numerical Features

- geo level1 id, geolevel2 id, geolevel3 id geographical region of buildings
- Count floors pre eq : number of floors of building
- Age : age of building
- Area Percentage : normalized area of building
- Height Percentage : normalized height of building

3.1.2 Categorical Features

- Land Surface Condition : surface condition of land
- Foundation Type : foundation used in the building
- Roof Type : roof used in the building
- Ground Floor Type : category of ground floor
- Other Floor Type : construction used in other than ground floor
- Position : position of building
- Plan Configuration : building plan configuration

3.1.3 Boolean Features

- has superstructure adobe mud,
- has superstructure mud mortar stone,
- has superstructure stone flag,
- has superstructure cement mortar stone,
- has superstructure mud mortar brick,
- has superstructure cement mortar brick,
- has superstructure timber,
- has superstructure bamboo,
- has superstructure rc non engineered,
- has superstructure rc engineered,
- has superstructure other,
- legal ownership status,
- count families
- has secondary use
- has secondary use agriculture
- has secondary use hotel
- has secondary use rental

3.2 Research Design

This section describes each stage of the methodology implemented in this research to build the classifiers. Each of the stages is followed sequentially that starts with data pre-processing and ending with the testing of the model on unseen data. Figure 2 shows the different stages of the methodology implemented to build the ML classifiers. They are further elaborated in the below sections.

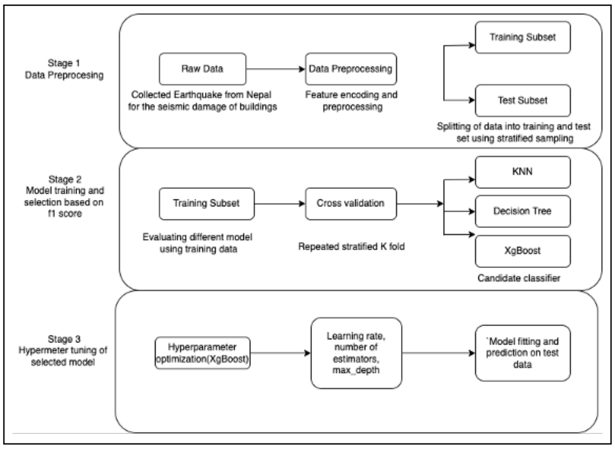

Figure 2: Flowchart of methodology followed in this research

3.2.1 First Stage Data Preprocessing

Data preprocessing plays an important step in machine learning before feeding it to the model for training, validation, and testing purpose. Data preprocessing steps taken in this research are as follows:

One hot encoding : Each of the categorical values is converted into a numerical value before training a model as most of the machine learning algorithms cannot directly deal with them. In this research, each level of categories was converted into a separate feature in the dataset containing binary values (1 or 0).

Stratified sampling : Under this technique, the entire dataset was divided into subgroups called strata, and then apply normal random sampling techniques within those created Strata. It helps to reduce the sampling error when there is unbalanced data by preserving the ratio of examples in each class they are in the original dataset. Validation error of stratified sampling is very close to the generalization error as compared to just random sampling. This research used this approach to split the dataset into train and test sets, a training set with 80 percent data and testing sets with 20 percent data.

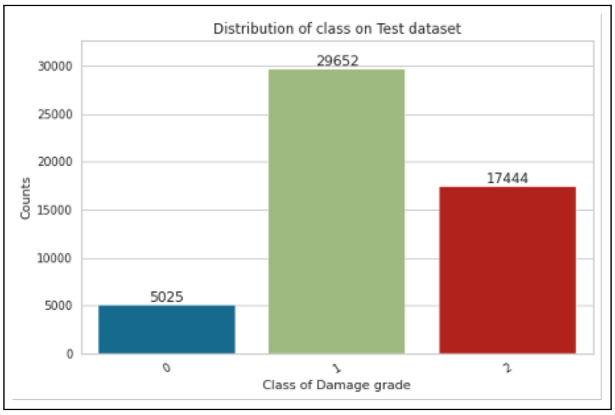

Figure 3: Distribution of class on test set after stratified sampling

3.2.2 Second Stage Model Evaluation

Knn, decision tree and xgboost were selected as candidate algorithm for this research. Knn and decision tree were chosen known for their simplicity and model interpretability where as an xgboost for their capacity to handle imbalance data and data parallelism during training. Three candidate algorithms knn, decision tree, and xgboost models were trained and fit using default parameters using the stratified K fold validation methods. To examine the efficiency of the model, metrics like F1, AUC and Accuracy were evaluated on both validation sets and test sets. Table 1 and figure 4 represent the different evaluation metrics of the candidate's model. The observed result suggests that the xgboost performs best on the validation datasets. The overall accuracy is

0.7227,0.699 and 0.6498 for xgboost, knn, and decision tree respectively.

Table 1: Evaluation metrics of each candidate algorithm

Metric	Xgboost	Knn	Decision Tree
AUC	0.8174	0.7818	0.682
F1	0.7146	0.6958	0.6503
Prec.	0.723	0.6954	0.651
Recall	0.6291	0.6282	0.6016
Accuracy	0.7227	0.699	0.6498

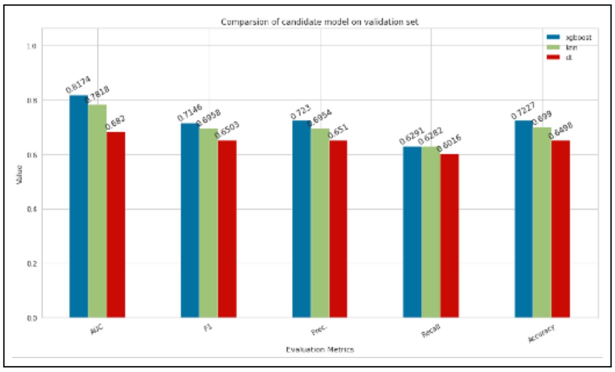

Figure 4: This chat shows evaluation metrics of each candidate model trained on the training dataset

From figure 4 it is clear that xgboost outperforms all other candidate models on all the evaluation metrics AUC, F1, Precision, Recall, and Accuracy. The decision tree performs the worst among all the model evaluated on the validation sets.

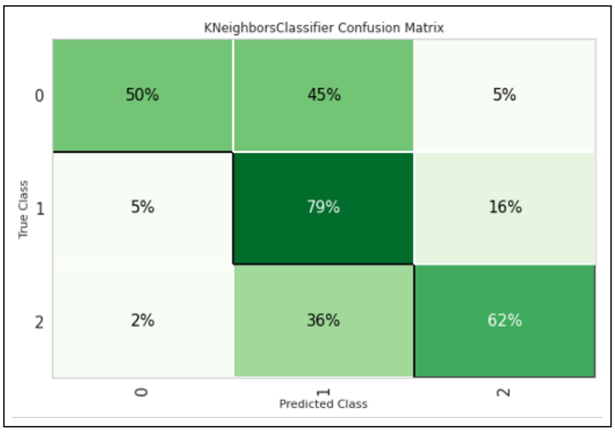

Figure 5: Confusion matrix of knn on validation set

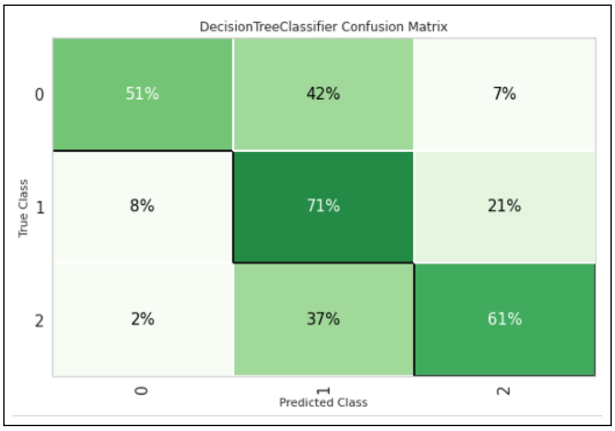

Figure 6: Confusion matrix of a decision tree on validation set

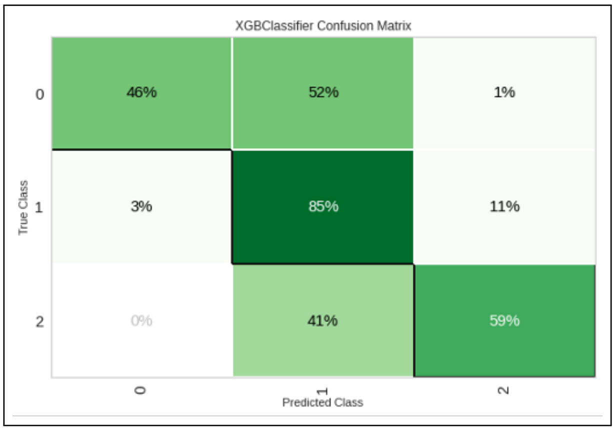

Figure 7: Confusion matrix of xgboost on the validation set

Figure 5, figure 6 and figure 7 shows the confusion matrix of knn, decision tree and xgboost respectively. From them, it can be clearly seen that all the candidate model used in this research performs better on class 1 as compared to the other class and class 0 has the worst performance.It is understandable given that the datasets are highly imbalanced where class 1 is the majority class and class 0 is the minority class.

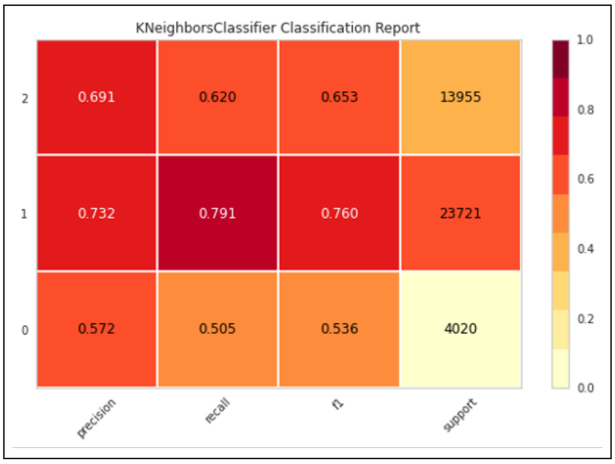

Figure 8: Class report of knn on the validation set

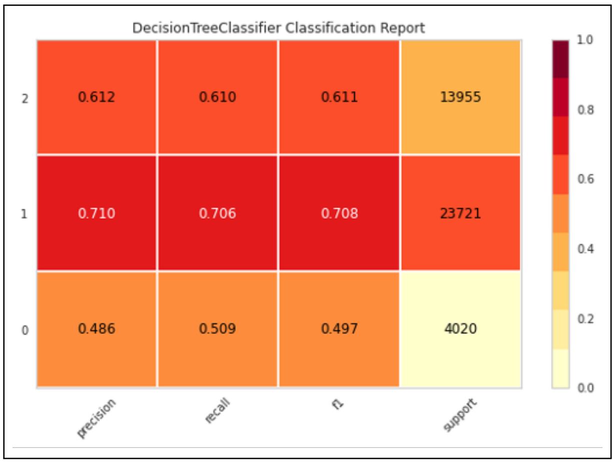

Figure 9: Class report of a decision tree on the validation set

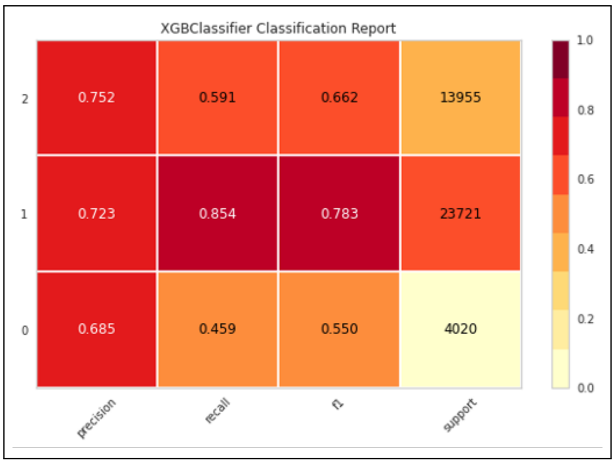

Figure 10: Class report of xgboost on the validation set

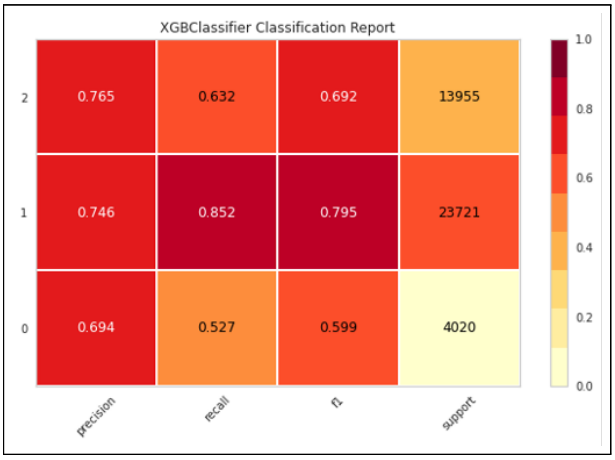

Figure 11: Class report of xgboost on the validation set after hyperparameter tuning

Figure 8, figure 9, and figure 10 show the class report of knn, decision tree, and xgboost respectively. Each of these class reports suggests that class 1 has the best precision, recall, and f1 score as compared to class 0 and class 2. This behavior is repeated among all the classifiers as class 1 has more training examples to learn hidden pattern better as compared to the others.

3.2.3 Hypermeter Tuning and Prediction on Test Data

Hyperparameters are tunable parameters and affect the performance of the model. These values can dominate the learning process of the classifier. Thus, to achieve the maximal accuracy of the model, it is a must to select the hyperparameter and optimize them. After selecting xgboost as the best-performing classifier and visualizing its performance, the final stage is to tune its relevant hyperparameter using a random search algorithm and compare the results before and after the tuned hyperparameter. The study includes optimizing the three most important xgboost hyperparameters.

1. Maximum depth : It indicates the maximum number of nodes allowed in the classifier counting from the root to the farthest leaf of a tree. A higher value of maximum depth can learn more complex relationships in the data but they allow a model to learn particular sample-specific patterns which could just have been noise.

2. Number of trees : The number of trees is one of the critical hyperparameters for xgboost classifier. Generally speaking, the number of trees can be increased until the performance on unseen data increases but the downside is that it may increase the learning time and may lead to overfitting.

3) Learning rate : This hyperparameter defines the amount of correction or step size after each iteration to correct the errors of the previous model. Generally, lower learning rate makes our model robust to the overfitting problem but increases the time complexity even for marginal improvement.

Table 2: Hyperparameter value of xgboost after using random search

Hyperparameter	Value
Maximum depth	8
Number of tress	300
Learning rate	0.2

Table 3: Performance of tuned xgboost on the validation set

Evaludation Metrics	Mean value of 5 fold
Accuracy	0.7397
AUC	0.8362
Recall	0.6617
Precision	0.7388
F1	0.7344

From table 3 and figure 11, it is clear that evaluation metrics of xgboost have increased slightly after hypermeter tuning. Besides that, tuned xgboost handles the class imbalance problem better than the untuned one when compared to figure 11 and figure 10.

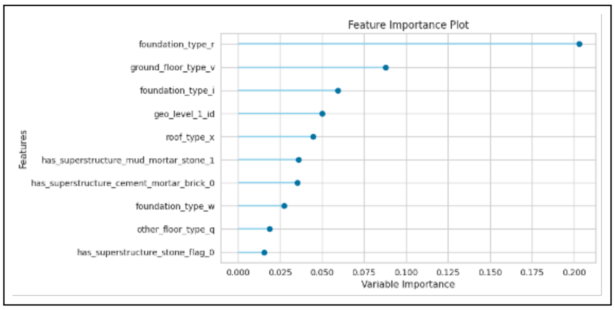

Figure 12: Feature importance plot of xgboost classifier

Figure 12 depicts the most important feature of datasets to classify the target variable which is damage grade while classifying using the xgboost classifier.

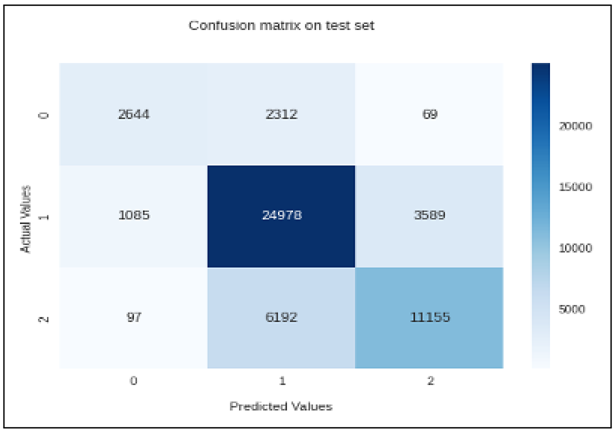

Figure 13: Confusion matrix of tuned xgboost classifier on the test set

Table 4: Class report of tuned xgboost on the test set

Class	Precision	Recall	F1	Support
0	0.691061	0.526169	0.597447	5025
1	0.746013	0.842372	0.791269	29652
2	0.753055	0.639475	0.691633	17444

Figure 13 and Table 4 show the performance of the model on the test set. The performance of the model on the test set nearly resembles its performance on the validation set which suggests that the model is not suffering from overfitting of the model. Table 4 clearly shows that the model has high predictive accuracy on class 1 and less predictive accuracy on class 0 which is as expected given that class 1 is the majority class and class 0 is the minority class. It also suggests that the accuracy of the minority class can be increased provided enough training examples to learn the hidden pattern. The overall accuracy of the tuned model on unseen data is found to be 74.4 %.

4. Conclusion

This study utilized the knn, decision tree, and xgboost classifier for categorizing the building based on their vulnerability during the seismic event using the input dataset collected from the Nepal earthquake. This study concludes that xgboost performed well as compared to other classifiers in classifying the target variable. Furthermore, the xgboost model hyperparameter was configured expecting to increase the efficiency of the model. The model after hyperparameter tuning has shown slight improvement as compared to the one with a default value. The potential reasons behind not achieving substantial accuracy especially on the minority classes even after tuning might be the imbalanced datasets, quality of datasets, size of datasets, and high variability on the datasets.

Machine learning classifier requires enough training examples to learn the hidden pattern in the data. Since the input datasets have a smaller number of training examples in the minority class, we can expect that this model would perform better provided enough training examples. Furthermore, the study may get extended using big and balanced datasets.

References

[1] AC Boissonnade, WM Dong, SC Liu, and HC Shah. Use of pattern recognition and bayesian classification for earthquake intensity and damage estimation. *International Journal of Soil Dynamics and Earthquake Engineering*, 3(3):145–149, 1984.

[2] S Mohanaselvi and G Hemapriya. Application of fuzzy logic to earthquake damage predictions. In *AIP Conference Proceedings*, volume 2112, page 020043. AIP Publishing LLC, 2019.

[3] Ismaël Riedel, Philippe Guéguen, Mauro Dalla Mura, Erwan Pathier, Thomas Leduc, and Jocelyn Chanussot. Seismic vulnerability assessment of urban environments in moderate-to-low seismic hazard regions using association rule learning and support vector machine methods. *Natural hazards*, 76(2):1111–1141, 2015.

[4] Sujith Mangalathu and Jong-Su Jeon. Machine learning–based failure mode recognition of circular reinforced concrete bridge columns: Comparative study. *Journal of Structural Engineering*, 145(10):04019104, 2019.

[5] Kuldeep Chaurasia, Samiksha Kanse, Aishwarya Yewale, Vivek Kumar Singh, Bhavnish Sharma, and BR Dattu. Predicting damage to buildings caused by earthquakes using machine learning techniques. In *2019 IEEE 9th International Conference on Advanced Computing (IACC)*, pages 81–86. IEEE, 2019.

[6] Gian P Delacruz. Using generative adversarial networks to classify structural damage caused by earthquakes. 2020.

[7] Samuel Roeslin, Quincy Ma, Hugon Juárez-Garcia, Alonso Gómez-Bernal, Joerg Wicker, and Liam Wotherspoon. A machine learning damage prediction model for the 2017 puebla-morelos, mexico, earthquake. *Earthquake Spectra*, 36(2_suppl):314–339, 2020.

[8] Shohei Naito, Hiromitsu Tomozawa, Yuji Mori, Takeshi Nagata, Naokazu Monma, Hiromitsu Nakamura, Hiroyuki Fujiwara, and Gaku Shoji. Building-damage detection method based on machine learning utilizing aerial photographs of the kumamoto earthquake. *Earthquake Spectra*, 36(3):1166–1187, 2020.

[9] Aishwarya Kumaraswamy, Bhargava N Reddy, and Rithvik Kolla. Richter's predictor: Modelling earthquake damage using multi-class classification models. In *2020 Third International Conference on Advances in Electronics, Computers and Communications (ICAECC)*, pages 1–5. IEEE, 2020.

[10] Francesco Nex, Diogo Duarte, Fabio Giulio Tonolo, and Norman Kerle. Structural building damage detection with deep learning: Assessment of a state-of-the-art cnn in operational conditions. *Remote sensing*, 11(23):2765, 2019.

[11] Peter Bull, Isaac Slavitt, and Greg Lipstein. Harnessing the power of the crowd to increase capacity for data science in the social sector. *arXiv preprint arXiv:1606.07781*, 2016.