Prediction of earthquake by machine learning models and neural network

Bohong Xu

School of Artificial Intelligence, South China Normal University, 510170 GuangZhou, China

Abstract. The occurrence of earthquakes has been examined from many aspects. Obviously, earthquakes occur without any warning and can damage entire cities in just a moment, causing enormous casualties and large amount of economic loss. In the last few years, people have given great effort to examine the earthquake from many aspects with various methods. The use of machine learning has solved seemingly intractable problems, inspiring the prospect of finding suitable solutions to dilemmas that are now considered unsolvable. Earthquake prediction, in this article, is executed with a neural network and machine learning models. After reading many articles, the author discovered that not many articles use a fully connected neural network to predict earthquake magnitude. In this article, the author first process the dataset sought from the internet, then predict the magnitude and the alert of earthquakes, with four machine learning models and a neural network. After that, the result of the accuracy are shown at the end of the article.

1 Introduction

Earthquake is a kind of natural phenomenon caused by the activities of continental crust. With serious destruction to the surface of the earth, many infrastructures are easy to be damaged. Some earthquakes that can release a lot of energy can even cause the explosion of volcanoes or tsunami. These natural disasters can cause a large quantity of loss of possessions and threaten people's lives. Lots of effort was spent on the Parkfield prediction experiment [1]. The results gained from researches made the scientific community wonder whether earthquakes could be predicted [2]. This question is still a in discussion among experts. Some scientists conclude that earthquake cannot be predicted [3]. Literature has a limited amount of studies that specifically compare the performance of various models [4]. But now, lots of studies are developed with tools of machine learning and deep learning. Deep learning is quickly becoming one of the most effective tools, not just in data science but also for solving complex and challenging problems in physics. This is backed up by the exceptional effectiveness of deep learning in uncovering complex patterns in numerous datasets. Prediction of earthquakes allows refugees to escape to safe areas in advance. With the help of warning, the rate of survival is likely to raise up. As it is widely used in the newspaper report, earthquake magnitude is a common measure to describe the level of the earthquake.

Corresponding author: 20222034042@m.scnu.edu.cn

Magnitude-frequency relationships are frequently studied in seismology, both through observation and theory. Their application is widespread in estimating the maximum magnitudes of future earthquakes, which is a crucial factor in earthquake prediction [5]. Magnitude is employed to measure the strength of the earthquake source through logarithmically calculations. And the public can know the strengthen quickly through the magnitude. Apart from the magnitude, the alert which is described in different kinds of color is also a convenient way for the refugees to know the strength of the earthquake. In this research, there are two methods to predict the alert and magnitude separately. One is the method of machine learning with four models. The other is building a neural network to predict the magnitude. Machine learning models consist of decision tree classifier, K-nearests classifier, random forest classifier, and gradient boosting classifier. All the algorithms have been applied to each dataset. To predict the magnitude and alert, predictive models, such as classification, decision trees, and Random Forests, are employed [6]. The research's objective is to forecast the magnitude value using an online dataset.

2 Methodology

2.1 Data description of the machine learning model

The data utilized in this investigation was extracted from Kaggle (https://www.kaggle.com/datasets/warcoder/earthquake-dataset) and can be accessed through the internet.

2.2 Data processing

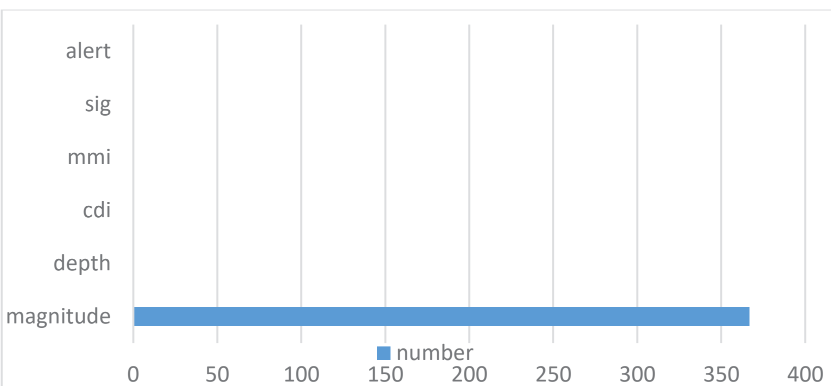

Fig. 1. Number of Null Values(Picture credit: Original)

After downloading the dataset, it is written into a data frame. The classifiers are trained on a dataset that contains 5 features, namely: ‘magnitude’, ‘depth’, ‘cdi’, ‘mmi’, ‘sig’, and ‘alert’ [7]. These 5 features are some fundamental data of earthquakes and they are more easily to be processed because they are in digit form which means many complex calculations can be executed directly and they are also typical features to be studied. The processed dataset contains 782 data for each feature. To insure there is no invalid data, calculating the number of null values is necessary. There are 367 null were found in the feature called “magnitude” as it is showed in Fig. 1. Then the null values are deleted in order not to cause error during the step of prediction. After deleting the null, a pie chart is drawn. This map (Fig. 2) shows the rate of different kinds of alerts.

 https://doi.org/10.1051/itmconf/20257004026

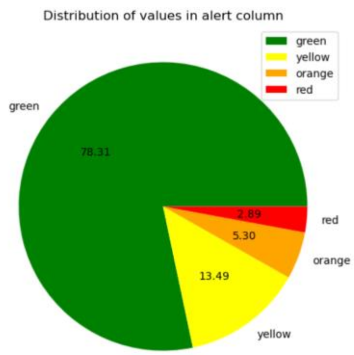

Fig. 2. Different Kinds of Alert(Picture credit: Original)

As Fig. 2 shows, the green feature is the most common one. And the unbalanced rate may cause a misunderstanding of minority class, which causes bad performance of the model. Thus, the minority class needs to be increased to a quantity that is nearly equal to the majority class. This process is over-sampling. There are many methods of over-sampling such as randomly over-sampling, synthetic minority oversampling technique, adaptive synthetic sampling and so on. Although random sampling is easier to be executed because it simply adds sample of the minority, over fitness may occur because no new information is added into the dataset. Thus, synthetic minority oversampling technique is selected because it can insert values into the minority to make new samples. After that, a new bar chart can be drawn. As the new chart (Fig. 3) shows, different kinds of colour bars have similar amount.

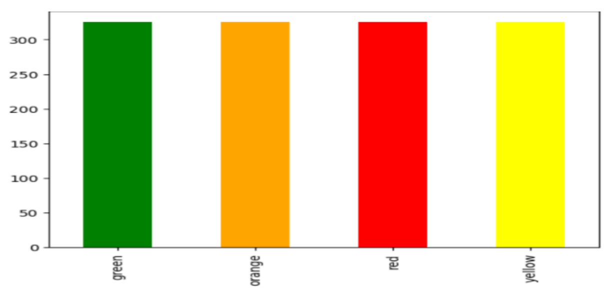

Fig. 3. Different Kinds of Alert (Picture credit: Original)

2.3 Data processing in the neural network model

In addition to the model that predicts alert level based on earthquake features, a regression model is also created to predict magnitude. This model is trained with tensorflow which is a good tool in the field of deep learning. While processing the data, the date data needs to be transformed to the format of second. Through this way, the time data can be calculated more conveniently. Then some features are chosen into the array. The features to be selected include: ‘date’, ‘latitude’, ‘longitude’, ‘depth’, and ‘magnitude’. And the target of the prediction is the feature ‘magnitude’. There are features in the X array that include 'date', 'latitude', 'longitude', and 'depth'. And the **Y** array includes the ‘magnitude’ feature. The **X** array undergoes the Min-Max normalization that can decrease the range of the sample into 0 to 1. Then the **X** array and **Y** array are reshaped to two-dimension arrays. After that, the dataset is split into a training and test set. There are 2000 samples for training and 2500 samples for validation set. While training the samples, some parameters of the function need to be set. The learning rate which is related to the progress of the learning is set to 0.001. The time of the training which is called "training_iteration" in the code is set to 1000. And every 200-time training the progress are showed on the console.

After that, a full connect neural network with 3 hidden layers is set. Two phases are involved in the operation of a neural network: learning and recall. When learning takes place, the outer product method stores information and determines the interconnection strengths between neurons. In the recall phase, the system's state changes as an external input is received and correlated with the stored information [8]. Every layer has 3 neurons. Each layer possesses a weight matrix and a bias. The bias term is got by the relu function. Then a forward propagation with 3 hidden layers and an output layer is set. In each layer, there is a matrix multiplication. This multiplication is governed by the X feature and the weight matrix of the first layer. Parameter X replaces the bias term of the preceding layer in the second and third layers. For the output layer, the parameters of the matrix multiplication is the output of the third layer's multiplication and the weight matrix of the output layer. After the calculation, the difference between the predicted value and the reality value is determined by using a loss function. There are also some steps to minimize the loss value. The first step in optimizing is to create an object that can update the weights and bias terms to reduce the loss. Then an object Saver is used to save the weights and bias terms during learning.

Following the processing of the data, a train set and a test set are separated. But before testing the model, the range of the data needs to be reduced so that the model of machine learning can understand the dataset more easily. The function "Standard Scaler" is applied to subtract the average value from the original feature value then the feature value is divided by the standard deviation. There is a potential relationship between earthquake depth and magnitude, as suggested by the correlation between depth and magnitude, which indicates that deeper seismic activity could affect the intensity of the event. These correlations provide a deeper understanding of the dataset, which can be utilized for further analysis and understanding of the seismic activity being studied [9].

3 Result

After fitting the dataset, a correlation matrix shows the relevance between each feature through a number. The correlation matrix is showed with a heat map as Fig. 4 shows.

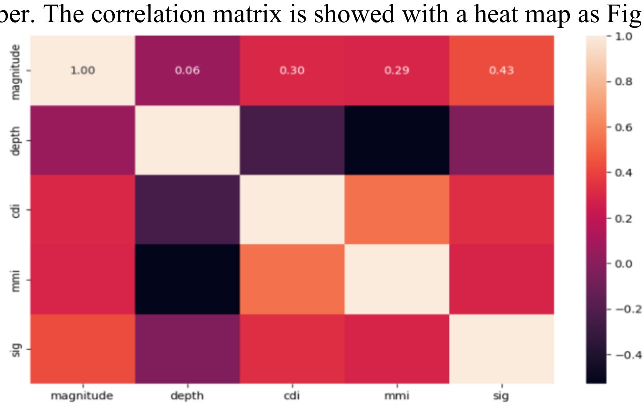

Fig. 4. Count of Different Kinds of Alert after Over Sampling(Picture credit: Original)

The first model is the "Decision Tree Classifier". The correlation map is showed in Fig. 5. The "Decision Tree Classifier" is based on the structure of trees. It can classify different kind of features by dividing different features into different layers of trees. The score of accuracy is 96.54%. The classification report is showed in Table 1.

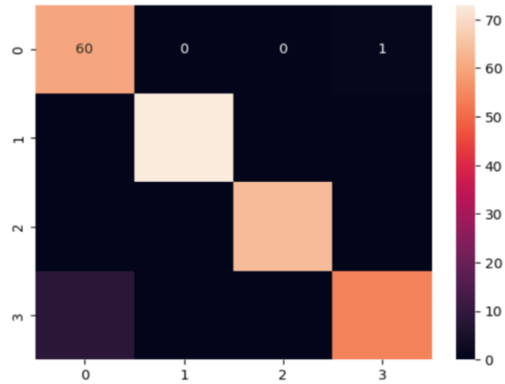

Fig. 5. Confusion Matrix of the Correct Classification of Decision Tree Classifier(Picture credit: Original)

Table 1. Report of the Accuracy of the Prediction Through Decision Tree Classifier

	positive predictive value	sensitivity	F-measure	class frequency
green	0.88	0.98	0.93	61
orange	1.00	1.00	1.00	73
red	1.00	1.00	1.00	64
yellow	0.98	0.87	0.92	62
classification success rate			0.97	260
class-wise average	0.97	0.96	0.96	260
weighted aggregation	0.97	0.97	0.97	260

The next model is the "K Nearest Neighbors" model which also called KNN. The correlation map is showed in Fig. 6. KNN is an algorithm of classification as a kind of supervised-learning. When a new sample is to be predicted, several samples which have the shortest distance are referred to make a new prediction. The score of accuracy is 92.69%. The classification report can be found in Table 2.

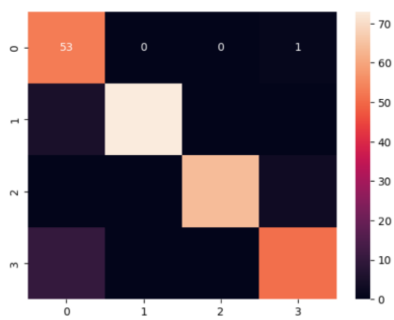

Fig. 6. Confusion Matrix of the Correct Classification of Decision Tree Classifier(Picture credit: Original)

Table 2. Report of the Accuracy of the Prediction Through K Nearest Neighbors

	positive predictive value	sensitivity	F-measure	class frequency
green	0.78	0.98	0.87	54
orange	1.00	0.94	0.97	78
red	1.00	0.96	0.98	67
yellow	0.93	0.84	0.88	61
classification success rate			0.93	260
class-wise average	0.93	0.93	0.92	260
weighted aggregation	0.94	0.93	0.93	260

After that, the model is the "Random Forest Classifier". The correlation map is showed in Fig. 7. In practice, decision tree-based supervised learning is a technique for constructing binary trees based on rules. When things are simple, decision trees generate multiple vertical and horizontal divisions within the data domain. Therefore, multiclass classification in this experiment is a particular good fit for them [10]. The score of accuracy is 96.15%. The classification report is showed in Table 3.

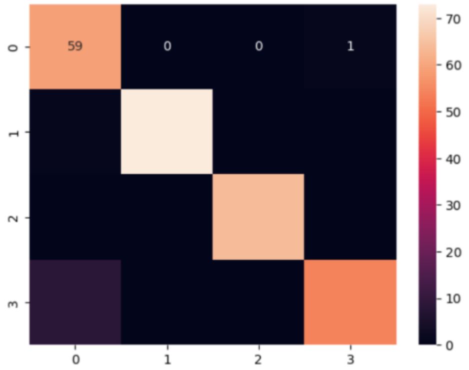

Fig. 7. Confusion Matrix of the Correct Classification Through Random Forest Classifier(Picture credit: Original)

Table 3. Report of the Accuracy of the Prediction Through Random Forest Classifier

	positive predictive value	sensitivity	F-measure	class frequency
green	0.87	0.98	0.92	60
orange	1.00	0.99	0.99	74
red	1.00	1.00	1.00	64
yellow	0.98	0.87	0.92	62
classification success rate			0.96	260
class-wise average	0.93	0.93	0.92	260
weighted aggregation	0.94	0.93	0.93	260

The last model is called "Gradient Boosting Classifier". The correlation map is showed in Fig. 8. The negative slope of the loss function can be used to approximate it. Then it uses a base learner to fit the residual. Therefore, the value of the loss function can be reduced. This algorithm can construct a model that is more accurate and stable buy combining many weak classifiers. The score of accuracy is 96.15%. The classification report is showed in Table 4.

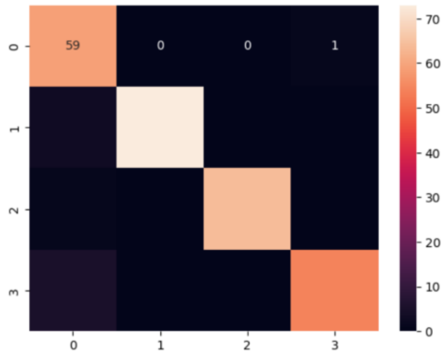

Fig. 8. Confusion Matrix of the Correct Classification Through Gradient Boosting Classifier(Picture credit: Original)

Table 4. Report of the Accuracy of the Prediction Through Gradient Boosting Classifier

	positive predictive value	sensitivity	F-measure	class frequency
green	0.87	0.98	0.92	60
orange	1.00	0.96	0.98	76
red	1.00	0.98	0.99	65
yellow	0.98	0.92	0.95	59
classification success rate			0.96	260
class-wise average	0.96	0.96	0.96	260
weighted aggregation	0.97	0.96	0.96	260

The neural network results have a mean squared error of 0.38 and a root squared error of 0.62.

4 Conclusion

In comparison to the other machine learning models, the Gradient Boosting Classifier is the most effective one. The grid search is also applied but its accuracy is not so high as the other models above by only 68%. Even the dataset has underwent the process of removing null and some data that is not good for the fit of the data. Since the application of the full connect network has not been exploited in this the prediction of magnitude, it is expected that the results obtained serve as a seed for further research.

References

1. S. N. A. Abdulrahman, S. N. A. Nassir, A. Saad, Earthquakes magnitude predication using artificial neural network in northern Red Sea area. Journal of King Saud University – Science. **24**, 301-313 (2012)

2. E. Florido, J. L. Aznarte, A. Morales-Esteban, and M. Á. Francisco, Earthquake magnitude prediction based on artificial neural networks. Croatian Operational Research Review. **7**, 159-169 (2016)
3. S. M. Mousavi, G. C. Beroza, A machine learning approach for earthquake magnitude estimation. Advancing Earth and Space Science research letter. 47, e2019GL085976 (2020)
4. A. Bhatia, S. Pasari, A. Mehta, Earthquake forecasting using artificial neural networks. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. **48**, 823-827 (2018)
5. M. Båth, Earthquake magnitude — recent research and current trends. Earth-Science Reviews. **17**, 315-398 (1981)
6. Y. Zhang, H. V. Burton, H. Sun, M. Shokrabadi, A machine learning framework for assessing post-earthquake structural safety. Structural Safety. **72**, 1–16 (2018)
7. K. Fauvel, D. Balouek-Thomert, D. Melgar, P. Silva, A. Simonet, G. Antoniu, A. Costan, V. Masson, M. Parashar, I. Rodero, A. Termier, A distributed multi-sensor machine learning approach to earthquake early-warning, The Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI-20), Association for the Advancement of Artificial Intelligence, (2020), 403-411
8. K. Y. Hsu, H. Y. Li, D. Psaltis, Holographic implementation of a fully connected neural network. Proceedings of the IEEE. **78**, 1637-1645 (1990)
9. A. Gupta, B. Sharma, P. Chingtham, Forecast of Earthquake Magnitude for North–West (NW) Indian Region Using Machine-Learning Techniques, International Conference on Entrepreneurship, Innovation, and Leadership, Singapore: Springer Nature Singapore, (2023) 361-376
10. S. Suthaharan, Machine Learning Models and Algorithms for Big Data Classification. Integrated Series in Information Systems. **36**, 1-12 (2016)