HIGH SCHOOL EDITION
Journal of Student Research
Volume 13 Issue 2 (2024)

Finding Earthquake Patterns with Time Series and Regression Models

Stavya Gaonkar[1], Aditi Ravindra[1], Apoorva Bathula[1], Rohan Kolala[1], Ansh Bhatia[1] and Sam Fendell[#]

[1]American High School, USA
[#]Advisor

ABSTRACT

Due to their unpredictability, earthquakes are certainly the most dangerous natural disasters. Scientists have long struggled to understand the movement of tectonic plates, as they occur at depths that cannot be directly seen and occur extremely slowly, at a rate that results in possibly one every thousand years between two adjacent plates. What if we could take a different approach, however? What if we could leverage the recent advances in machine learning to close this gap? That is exactly what we aimed to do in our research, with the question of whether various factors related to earthquakes, such as their magnitude and time of occurrence, could be determined through the analysis of years of historical data. This was conducted through the analysis of a dataset which contained records of all major earthquakes (magnitude 5.5 or higher) from 1900-2023. The library scikit-learn was used for various regression models, such as linear regression, random forest regression, and lasso regression, and the library FB prophet was used for time series forecasting, a method used to analyze patterns in previous earthquakes to predict new ones. Our conclusions were that, with some degree of error, the magnitude of an earthquake can be determined based on location/time and the time of occurrence can be pinpointed to specific days of the week as well as months.

Regression Models

Linear Regression

Linear regression is one of the most fundamental machine learning models: creating a line of best fit to model data. In this way, we can predict points that are not plotted on the graph. For example, in Figure 1, weight is plotted against height based on the points plotted in blue. With a linear regression model, which determines the equation for a line of best fit, the weight of someone who is 2 meters tall, which is not on the graph, can still be predicted.

The way this equation is created is by constantly altering the values of m and b in the equation $y = mx+b$ until the mean-squared error of the graph, or the average squared distance from each point on the graph from the line based on the equation, is minimized.

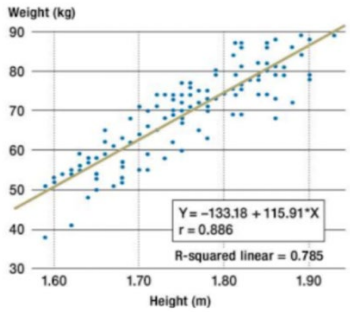

Figure 1. Example of Linear Regression using weight and height

When there is only one independent variable, such as the previous example, it is referred to as simple linear regression. For our research, we will be working with multiple linear regression, when there are multiple independent variables. In multiple linear regression, there is still a line of best fit created, but it is just plotted on a plane with more axes (Figure 2).

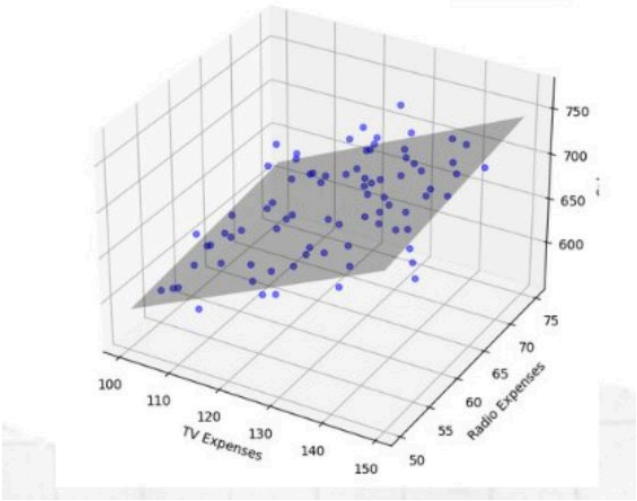

Figure 2. Example of multiple linear regression with two independent variables.

Lasso Regression

Lasso Regression effectively works the same way as linear regression except for one addition: the alpha parameter. The alpha parameter serves to reduce the amount overfitting of a model on a certain dataset. Overfitting refers to when a model tries to optimize all of its parameters so that it maximizes its performance on a specific set of data (Figure 3). The issue with this is that it is not generalizable; it is more important for a regression model to understand the bigger picture.

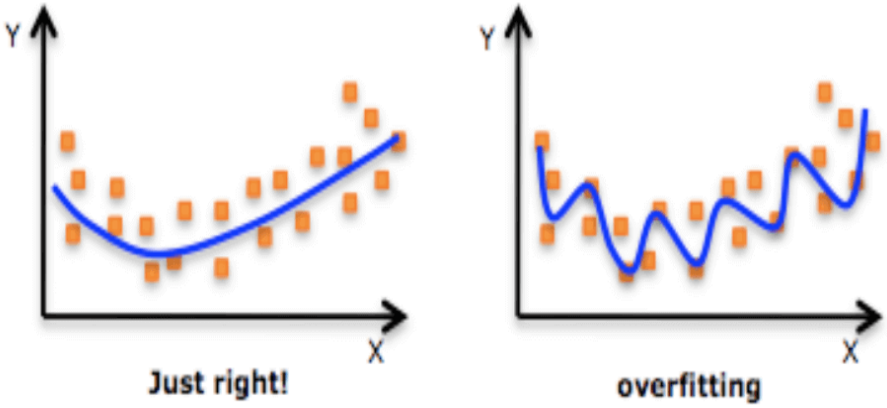

Figure 3. The problem of overfitting.

Therefore, the alpha parameter is extremely useful, as it helps regulate the linear regression model. The way it works is that alpha is like a multiplier to all of the parameters' coefficients on a regression model; therefore, the lesser the value of the alpha parameter, the lower the model's dependence on each individual parameter will be, reducing the overall amount of overfitting.

Random Forest Regression

Random forest regression is nothing like linear regression. Instead of a line of best fit to make predictions, it uses the concept of decision trees. Decision trees are usually used for classification, such as a "yes" or "no" output, as shown in Figure 4. However, it can also be used to make numerical predictions, in the form of random forest regression. This is done by adding or subtracting a numerical value every time a node is reached.

In the case of random forest regression, multiple decision trees are put together, determined by the parameter n_estimators, and the output from each of these decision trees are added together and divided by the number of decision trees: the outputs are effectively averaged (Figure 5).

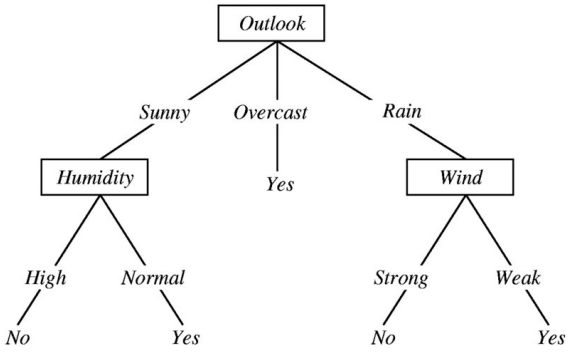

Figure 4. Example of a decision tree with the output "yes" or "no."

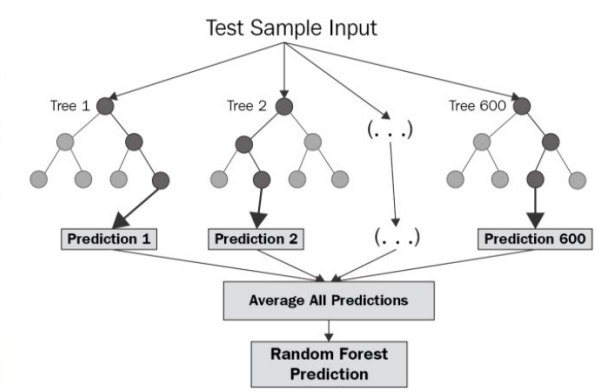

Figure 5. Represents a random forest regression model with 600 decision trees.

Time Series Analysis

In time series analysis, the value of a variable is predicted with respect to time based on previous values. A pattern is found in the variation of the variable with respect to time in the past and the time series model makes a prediction, in the form of a curve, of how it would look like in the future. Figure 6 shows a plot of how time series forecasting would be used to predict the number of occurrences of earthquakes per year with respect to time. The black dots represent the number of earthquakes that occurred in a given year and the dark blue line on the far right of the graph shows the predictions that the model is making for the future. For this example, the library FB Prophet was used.

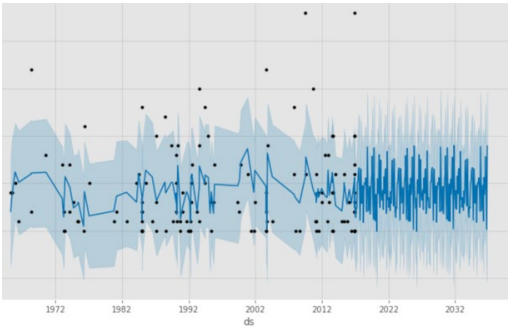

Figure 6. Graph of time series forecasting being used to predict the number of earthquakes per year.

Methods

Dataset

The dataset used for this research was the “Significant Earthquake Dataset 1900-2023” on Kaggle. This was helpful because there was a large amount of data, considering that all recorded earthquakes with magnitudes above 5.5 were recorded, but also because, unlike several other datasets we looked at, this one included various factors related to the earthquake, such as the specific coordinates of each earthquake, recorded to one decimal place, the magnitude, which we sought to investigate in this research, the depth, the gap between the tectonic plates, etc.

Jupyter Notebook

All code for this research was written on Jupyter Notebook, a coding environment that allows users to run specific blocks of code at a time and offers various programming languages. We used Python 3, a language commonly used for machine learning applications.

Regression Models

To implement these machine learning models in our code, we had to import libraries that execute functions which are not necessarily built-in to Python.

To begin with, to process the dataset, we imported Pandas. We downloaded the dataset from Kaggle as a csv file and had Pandas scan it into an accessible data frame; for data cleaning, we used the df.dropna() function in order to remove all rows of the dataset that contained N/A values.

For our first experiment, we wanted to see whether we would be able to predict the magnitude of an earthquake based on various factors, such as the location and the time of occurrence, hoping that there would be some pattern in the overall strength of an earthquake.

To do this, we implemented linear regression, random forest regression, and lasso regression, and we compared the relative errors for these regression models. Using train_test_split from scikit-learn and a test size of 0.2, we trained each model and then tested it. The library scikit-learn was used to develop each of the regression models; for lasso regression, we added an alpha value of 0.2, and for our random forest regressor, we added 1000 decision trees, or n_estimators, to optimize it on the training data without overfitting.

Code for Time Series Forecasting

The same dataset was used for the time series forecasting model that we developed. The library FB Prophet, created by developers from Facebook, was utilized for this implementation. For the time series forecasting, we plotted the number of earthquakes that occurred per year with respect to time. To do so, we created an array in which each index represented the number of earthquakes recorded in that year and iterated through the dataset, adding 1 to the value associated with the appropriate index. With the help of FB Prophet, we created the forecasting model and created graphs of its predictions using matplotlib.

Results

Correlation

	depth	gap	mag	dmin	rms	horizontalError	magError	depthError	latitude	longitude	magNst
depth	1.000000	0.019820	-0.131725	0.036721	0.000372	0.169074	-0.164053	0.303146	0.115954	0.067379	0.268505
gap	0.019820	1.000000	-0.583286	-0.018212	0.032035	-0.186082	-0.339636	-0.339038	0.041054	0.017595	0.163934
mag	-0.131725	-0.583286	1.000000	0.271421	-0.091635	0.420685	0.319012	0.355597	0.240605	0.378855	-0.164950
dmin	0.036721	-0.018212	0.271421	1.000000	-0.103323	0.494274	-0.070169	0.032681	0.408161	0.543854	0.157205
rms	0.000372	0.032035	-0.091635	-0.103323	1.000000	-0.119517	0.043108	-0.030571	-0.109787	-0.106985	-0.029938
horizontalError	0.169074	-0.186082	0.420685	0.494274	-0.119517	1.000000	0.066527	0.255864	0.154308	0.422239	0.016851
magError	-0.164053	-0.339636	0.319012	-0.070169	0.043108	0.066527	1.000000	0.039522	-0.180593	-0.126480	-0.537115
depthError	0.303146	-0.339038	0.355597	0.032681	-0.030571	0.255864	0.039522	1.000000	0.230213	0.203507	0.022647
latitude	0.115954	0.041054	0.240605	0.408161	-0.109787	0.154308	-0.180593	0.230213	1.000000	0.756466	0.223121
longitude	0.067379	0.017595	0.378855	0.543854	-0.106985	0.422239	-0.126480	0.203507	0.756466	1.000000	0.182806
magNst	0.268505	0.163934	-0.164950	0.157205	-0.029938	0.016851	-0.537115	0.022647	0.223121	0.182806	1.000000

Figure 7. Represents a correlation matrix, which represents the extent to which factors depend on each other.

We created a correlation matrix (Figure 7) for the various factors present on the dataset, and we noticed that the factors that had the greatest influence on the magnitude of an earthquake were in fact the location (longitude and latitude). Combined, the latitude and longitude were 62% responsible for an earthquake's magnitude. Therefore, our regression models, which predicted magnitude based on the location and time of occurrence, should have been accurate to some extent.

Regression Model Accuracy

Regression Model	Mean Squared Error
Linear	72.5%
Lasso	68.0%
Random Forest	34.3%

Figure 8. Shows the mean squared error of the predictions generated by the three regression models.

On the testing data, linear and lasso regression did not perform well at all, averaging a mean-squared distance of about 0.7 from the actual value. What this means is that, for example, if the magnitude of an earthquake was 5.0, then, on average, the linear and lasso regressions would predict a value whose distance from 5.0, squared, is 0.7. Random forest regression, on the other hand, performed extremely well, having a mean-squared error of just 0.34.

Time Series Forecasting

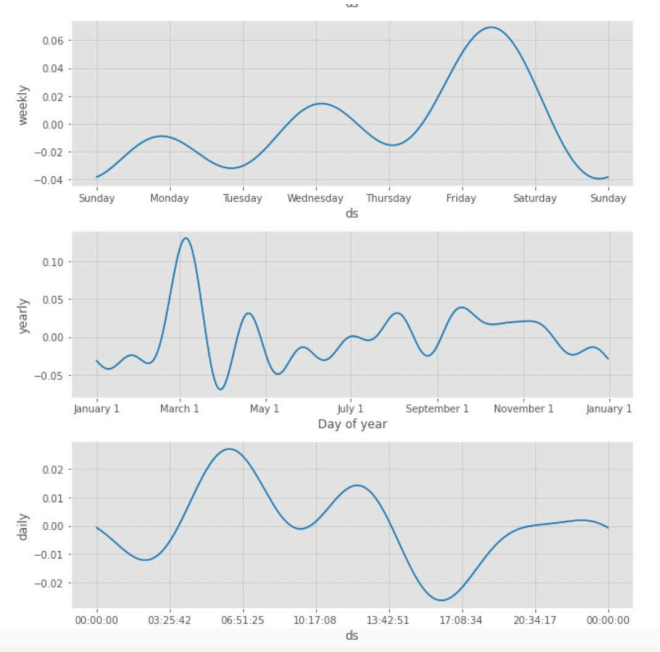

Figure 9. Analyzed global trends in the occurrence of earthquakes.

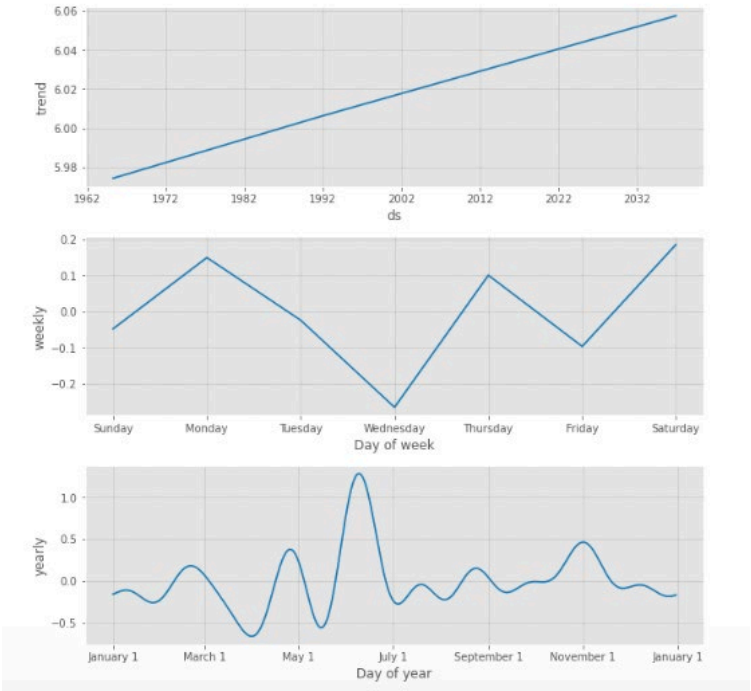

Figure 10. Analyzed trends from California.

These two figures (Figures 9 and 10) represent the predictions that our time series forecasting model made for the next year. Each of these figures contains three graphs: daily, weekly, and yearly predictions for earthquakes. In the discussion section, the accuracy of these results will be discussed.

Discussion

Our first important finding through our analysis of earthquakes was the correlation between the magnitude of an earthquake and the location in which it occurs (Figure 7). This points to how the strength of the earthquakes that occur in a certain area are extremely consistent; therefore, the magnitude of an earthquake at a location determined by its longitude and latitude can be roughly approximated with little error.

This conclusion was reinforced by our finding, through the random forest regression model we developed, that the magnitude of an earthquake can be predicted fairly well through just the latitude, longitude, and time of occurrence of an earthquake. The reason that the linear and lasso regression models were two times more inaccurate than the random forest regression model was because the relationship between magnitude and these three factors could not be predicted effectively through a linear relationship, although there was a clear correlation between them; lasso regression is essentially linear regression but with an optimized parameter alpha.

Lastly, through the time series forecasting that we performed, we made several additional findings. To begin with, we found that earthquakes occur most often during the dry season. In Figure 9, our model predicted that, in the next year, the most earthquakes would happen during May and June, a massive peak compared to the relatively low lines present in the rest of the graph. This is supported by scientific evidence, as when there is less groundwater, there is not as much material to push down on land; this causes land to rise, which makes earthquakes significantly more likely. In addition, we also found that earthquakes are no more likely to occur during the day or during the night, which disproves the common misconception that earthquakes occur at night. This can be seen in Figure 9, where there are no obvious peaks at night. This finding is also supported by studies showing that there is no connection between the position of the moon and the degree of seismicity that occurs.

Conclusion

Our research was successful in finding patterns in the occurrence of earthquakes through a thorough historical analysis, supported by regression and time forecasting models. We believe that our findings, that the strength of earthquakes in a certain location tends to be around the same value and that earthquakes tend to occur during the dry season, could be useful to seismologists who are looking for ways to ensure that we are better protected by earthquakes.

Limitations

One limitation was the historical data that we had access to. Although we were able to analyze a dataset that consisted of a century of earthquakes, the ones listed had magnitudes of 5.5 or higher. This could have potentially skewed our data; there may be something that all higher-magnitude earthquakes have in common.

In addition, another limitation could have been the fact that a lot of the entries in our dataset had null, or N/A, values. While using the dropna() function, we could have eliminated a significant amount of datapoints that could have been otherwise useful to our research. Using this function removed more than 27% of our data.

Acknowledgments

We would like to thank ASDRP for familiarizing us with the fundamentals of machine learning and our advisor Sam Fendell for providing us with this research topic, as well as key insights and recommendations over the course of our research. We extensively used Jupyter Notebook, which this research wouldn't have been possible

without. Finally, an additional thank you FB Prophet, a time forecasting library developed by Facebook, helped us a lot in the latter part of our research.

References

Islam, Jahaidul. “Significant Earthquake Dataset 1900-2023.” Kaggle, 18 Feb. 2023, www.kaggle.com/datasets/jahaidulislam/significant-earthquake-dataset-1900-2023.
“Is There Such a Thing as Earthquake Season?: Earth.” EarthSky, 22 Apr. 2021, earthsky.org/earth/earthquake-season-taiwan-dry-monsoon/.
“Can the Position of the Moon or the Planets Affect Seismicity? Are There More Earthquakes in the Morning/in the Evening/at a Certain Time of the Month?” Can the Position of the Moon or the Planets Affect Seismicity? Are There More Earthquakes in the Morning/in the Evening/at a Certain Time of the Month? | U.S. Geological Survey, www.usgs.gov/faqs/can-position-moon-or-planets-affect-seismicity-are-there-more-earthquakes-morningin-eveningat
Prabhakaran, Selva. “Time Series Analysis in Python - a Comprehensive Guide with Examples - Ml+.” Machine Learning Plus, Machine Learning Plus, 8 Sept. 2020, www.machinelearningplus.com/time-series/time-series-analysis-python/.
Brownlee, Jason. “How to Check If Time Series Data Is Stationary with Python.” Machine Learning Mastery, 14 Aug. 2020, machinelearningmastery.com/time-series-data-stationary-python/.
Erica. “Introduction to the Fundamentals of Time Series Data and Analysis.” Aptech, 12 May 2021, www.aptech.com/blog/introduction-to-the-fundamentals-of-time-series-data-and-analysis
“Forecasting at Scale.” Prophet, facebook.github.io/prophet/. Accessed 28 Feb. 2024.
Bakshi, Chaya. “Random Forest Regression.” Medium, Level Up Coding, 9 June 2020, levelup.gitconnected.com/random-forest-regression-209c0f354c84.