



OPEN

Improving earthquake prediction accuracy in Los Angeles with machine learning

Cemil Emre Yavas¹✉, Lei Chen¹, Christopher Kadlec¹ & Yiming Ji^{1,2}

This research breaks new ground in earthquake prediction for Los Angeles, California, by leveraging advanced machine learning and neural network models. We meticulously constructed a comprehensive feature matrix to maximize predictive accuracy. By synthesizing existing research and integrating novel predictive features, we developed a robust subset capable of estimating the maximum potential earthquake magnitude. Our standout achievement is the creation of a feature set that, when applied with the Random Forest machine learning model, achieves a high accuracy in predicting the maximum earthquake category within the next 30 days. Among sixteen evaluated machine learning algorithms, Random Forest proved to be the most effective. Our findings underscore the transformative potential of machine learning and neural networks in enhancing earthquake prediction accuracy, offering significant advancements in seismic risk management and preparedness for Los Angeles.

Accurately predicting earthquakes is crucial for mitigating risks and enhancing preparedness, especially in seismically active regions like Los Angeles. The ability to forecast seismic events with high accuracy can significantly impact disaster management strategies, reduce potential casualties, and minimize economic losses. In this context, our research contributes to the ongoing efforts to improve earthquake prediction using advanced machine learning and neural network techniques.

In a prior study, we developed a predictive pattern matrix for Los Angeles, achieving an accuracy rate of 69.14% in predicting the maximum magnitude earthquake within one of six categories¹. This initial success raised a critical question: could such predictive accuracies be replicated or even improved in other seismically active regions? To explore this, we extended our research to Istanbul, a city near the North Anatolian Fault, one of the most earthquake-prone regions globally, and achieved an accuracy of 91.65%². Building on these promising results, we further refined our approach and achieved an accuracy rate of 98.53% for San Diego³.

Encouraged by these successful outcomes in San Diego and Istanbul, we revisited Los Angeles to determine if we could surpass the previous accuracy of 69.14%. This research answers that question affirmatively; we successfully predicted earthquakes for Los Angeles with an accuracy of 97.97%. The results demonstrate the potential for significant advancements in earthquake prediction accuracy using machine learning techniques, contributing to more effective disaster preparedness and response strategies.

The importance of this research lies in its ability to harness the power of machine learning and neural networks to improve the predictive accuracy of seismic events, thereby offering valuable insights for disaster management authorities and policymakers. By enhancing our understanding of seismic patterns and advancing earthquake prediction methodologies, this study aims to reduce the adverse impacts of earthquakes in highly populated urban areas.

In this study, we applied a variety of machine learning and neural network techniques to predict seismic events in Los Angeles, utilizing a comprehensive dataset that includes all recorded earthquakes over the past 12 years. Through advanced feature engineering, we constructed a feature matrix incorporating critical predictive input variables informed by prior research. Previous studies have suggested various strategies to enhance earthquake prediction accuracy, such as identifying deep seismic patterns, testing different prediction models, and examining seismic frequency characteristics^{4–7}. Building upon these foundational works, we developed and evaluated sixteen different machine learning and neural network algorithms to determine the most effective model for predicting the highest magnitude of potential earthquakes within a 30-day period.

Our findings are significant: the Random Forest model emerged as the top performer, achieving an accuracy of 97.97%. This high level of accuracy represents a considerable improvement over previous efforts, confirming the potential of machine learning techniques to enhance earthquake prediction capabilities.

Our research builds on a diverse array of studies in earthquake prediction spanning from 1990 to 2024.

¹Department of Information Technology, Georgia Southern University, Statesboro, GA, USA. ²Yiming Ji contributed equally to this work. ✉email: cy02470@georgiasouthern.edu

Olsen et al. (1995)⁸ emphasized significant ground velocities expected during earthquakes, particularly near fault lines and in regions like the Los Angeles basin, where prolonged shaking can occur. This highlights the importance of understanding seismic dynamics in specific geographical locations.

In the realm of machine learning applications for earthquake prediction, Asim et al. (2018)⁹ explored the use of support vector regressors and hybrid neural networks to develop a predictive model. Their focus on seismic regions such as Hindukush, Chile, and Southern California aligns with our interest in predicting earthquakes in Los Angeles, underscoring the relevance of regional seismic activity in predictive modeling.

Zhang et al. (2019)¹⁰ proposed a precursory pattern-based feature extraction technique to enhance earthquake prediction performance. This approach highlights the importance of extracting meaningful patterns from seismic data, resonating with our objective of creating a predictive pattern matrix for earthquakes in Los Angeles.

Bao et al. (2021)¹¹ introduced an innovative method combining explicit and implicit features through deep learning-based electromagnetic signal analysis for earthquake magnitude prediction. This method offers valuable insights into improving prediction accuracy, aligning with our goal of enhancing forecasting precision.

Advanced neural network models, such as the graph convolutional neural network proposed by Bilal et al. (2022)¹², can significantly enhance earthquake prediction efficiency. Their focus on early earthquake detection using sophisticated neural network architectures underscores the potential of cutting-edge technologies to improve predictive capabilities.

The work of Hsu & Pratomo (2022)¹³ on early peak ground acceleration prediction using Long Short-Term Memory (LSTM) neural networks showcases the importance of models that capture order dependence in seismic waves. This aligns with our approach of utilizing machine learning algorithms to forecast earthquake occurrences within a specific timeframe.

By integrating findings from various studies on earthquake prediction, our research aims to enhance predictive modeling techniques specifically for the Los Angeles region. Through the integration of machine learning algorithms, feature extraction methods, and advanced neural network architectures, we strive to improve the accuracy and timeliness of earthquake forecasts, thereby enhancing disaster preparedness and response strategies.

The structure of this paper is as follows: in the next section, we do a background analysis, then we describe the dataset and the feature engineering process. Following that, we discuss the methodology employed in this study, including the various machine learning algorithms and neural network models evaluated. We then present our results and findings, highlighting the performance of the top-performing models. Finally, we conclude with a discussion of the implications of our findings and potential directions for future research.

Background

This section presents the scientific and technical foundations for forecasting earthquakes in Southern California. It synthesizes key research on seismic activity, fault systems, ground motion, and earthquake hazards, emphasizing the significance of site-specific data for accurate seismic hazard assessments. This knowledge underpins the use of machine learning and neural network algorithms for forecasting earthquake magnitudes, enhancing preparedness and mitigation strategies.

Olsen's study (2000)¹⁴ in the Los Angeles Basin demonstrates the impact of local geological conditions on ground motion during earthquakes. The three-dimensional modeling shows substantial ground motion variations due to the basin's subsurface structure, underscoring the need for detailed geological data in seismic hazard assessments.

Donnellan et al. (2015)¹⁵ examine the 2014 La Habra earthquake's implications for future seismic activity near Los Angeles. Their analysis of geodetic data and fault interactions highlights the role of interconnected fault systems in seismic events, stressing the importance of monitoring and advanced modeling.

Hauksson (1990)¹⁶ offers insights into the earthquake potential of thrust faults beneath the Los Angeles Basin, advocating for their inclusion in forecasting models to improve accuracy.

Shen et al. (1996)¹⁷ provide data on crustal deformation in the Los Angeles Basin, which aligns with existing earthquake probability models, refining forecasting algorithms.

Loveless & Meade (2011)¹⁸ explore stress modulation on the San Andreas fault, illustrating its influence on earthquake recurrence, crucial for refining forecasting models.

Romero et al. (2010)¹⁹ address the environmental implications of earthquakes, such as fault rupture and landslides, advocating for comprehensive forecasting models that account for diverse hazards.

Roten et al. (2014)²⁰ investigate ground motion reductions due to San Andreas fault zone plasticity, enhancing earthquake forecast accuracy through simulated scenarios.

Shaw & Suppe (1996)²¹ emphasize the significance of newly identified fault systems in seismic risk assessments, improving the precision of forecasts.

Zechar & Jordan (2008)²² evaluate alarm-based forecasting models, focusing on their accuracy and reliability, which is essential for effective earthquake forecasting.

Huang et al. (2020)²³ discuss the use of the Extreme Learning Machine (ELM) algorithm in forecasting earthquake casualties, highlighting the role of variables like intensity and building collapse rates in enhancing prediction accuracy.

Initiatives such as the Collaboratory for the Study of Earthquake Predictability (CSEP) and the Regional Earthquake Likelihood Models Experiment (RELM) by Schorlemmer et al. (2010)²⁴ have paved the way for prospective earthquake prediction efforts. Studies evaluating return periods and occurrence probabilities of maximum magnitude earthquakes by Al-Heety (2024)²⁵, and improved algorithms like Extreme Learning Machines (ELM) by Huang et al. (2020)²³ enhance earthquake casualty predictions. Insights from studies on fixed recurrence and slip models by Rubinstein et al. (2012)²⁶ and self-organized criticality by Yang et al.

(2004)²⁷ provide valuable perspectives on earthquake behavior prediction and the challenges posed by complex seismic dynamics. Innovative approaches, such as the use of deep learning neural networks by Huang et al. (2018)²⁸ and attention mechanisms in earthquake prediction models by Kavianpour et al. (2021)²⁹, offer further advancements.

Geller et al. (1997)³⁰ express skepticism about earthquake predictability, yet ongoing research addresses these challenges, improving reliability.

Eberhard et al. (2012)³¹ highlight the importance of continuous experiments to refine forecasting models, particularly in seismically active regions.

Rubinstein et al. (2012)²⁶ underscore the importance of understanding stress accumulation and release on faults, contributing to improved forecasting algorithms.

Tehseen et al.'s study (2020)³² on earthquake forecasting using expert systems underscores the importance of long-term forecasts regarding the timing, intensity, and location of future earthquakes. Utilizing expert systems and comprehensive data analysis, researchers can develop more robust models tailored to specific regions like Southern California.

Ogata (2013)³³ advocates for statistical models of seismicity to accurately evaluate predictive performance. Assessing these models' efficacy in earthquake forecasting enhances reliability and contributes to more effective disaster mitigation.

Banna et al. (2021)³⁴ discuss the use of attention-based Bi-Directional Long Short-Term Memory (LSTM) networks, highlighting the potential of advanced machine learning techniques in seismic forecasting. Leveraging deep learning models like LSTM networks can improve forecast accuracy and enhance preparedness in regions such as Southern California.

Kagan (1997)³⁵ emphasizes real-time seismology's role in aiding relief efforts and issuing warnings before severe shaking occurs. Understanding the feasibility of earthquake forecasting through real-time monitoring is vital for improving preparedness strategies.

Ma et al. (2021)³⁶ evaluate the largest possible earthquake magnitudes in mainland China using extreme value theory, underscoring the role of ground-based observations and statistical analyses. Insights from such studies help refine predictive models and improve forecast accuracy in regions with high seismic activity.

Herrera et al. (2022)³⁷ focus on long-term forecasting of strong earthquakes using machine learning techniques to cluster earthquakes based on historical intervals. This approach enhances predictive capabilities for models in regions such as North and South America.

Michael (1997)³⁸ explores statistical techniques to evaluate earthquake forecasting methods, comparing observed outcomes with random chance. Such comparisons help improve model efficacy in seismic forecasting.

Kodera et al. (2016)³⁹ evaluate earthquake early warning systems for the 2016 Kumamoto earthquake, providing insights into system performance under heavy loading conditions. These evaluations guide the development of similar systems in earthquake-prone regions like Southern California.

Yuan et al. (2023)⁴⁰ examine the SARIMA model for forecasting earthquakes in the Longmenshan Fault Zone, offering a practical approach to predicting earthquake occurrence times and improving disaster preparedness.

Hajikhodaverdikhan et al. (2018)⁴¹ highlight the potential of integrating meteorological data into earthquake forecasting models. Using intelligent analysis of historical meteorological datasets, researchers can enhance forecast precision and early warning systems.

Astuti et al. (2013)⁴² investigate geoelectric field signals before earthquakes using adaptive STFT techniques, emphasizing the importance of signal analysis as input parameters for refining prediction models.

Nishikawa (2023)⁴³ compares statistical models for low-frequency earthquake activity, highlighting their importance in understanding slow earthquake activity patterns that may precede major events. This knowledge can improve forecast accuracy in regions like Southern California.

Nimmagadda & Dreher (2007)⁴⁴ emphasize the efficacy of ontology-based data warehousing in earthquake forecasting, enhancing model efficiency and accuracy.

Prasad et al. (2013)⁴⁵ analyze earthquake magnitude detection using primary and secondary waves, stressing the role of wave analysis for robust forecasts in regions like Southern California.

Yang et al. (2022)⁴⁶ streamline data processing and model development through automated regression pipelines, improving the efficiency of earthquake forecasting models.

Zheng and Tao (2023)⁴⁷ underscore the importance of accurate geophysical data in forecasting, while Hussain et al.⁴⁸ discuss the relationship between b-values and seismic stress levels, aiding high-magnitude earthquake predictions.

Research on diverse data sources, such as GPS, ionospheric data, and outgoing longwave radiation, has enhanced earthquake forecasting models. Gitis et al. (2021)⁴⁹ highlight the importance of seismological data, while Zhai et al. (2020)⁵⁰ investigate thermal anomalies using non-seismic time series data, demonstrating the multidisciplinary nature of earthquake research.

Investigations like Woith et al. (2018)⁵¹ explore the potential of animal behavior as a precursor to seismic events, reflecting the interdisciplinary approach in earthquake forecasting research.

Existing research provides extensive insights into seismic activity and fault dynamics in Southern California. Studies by Olsen¹⁴, Donnellan et al.¹⁵, and Hauksson¹⁶ have enhanced our understanding of seismic hazards, while further research on geodetic data¹⁷ and historical earthquake analysis⁵² has shaped our approach. Despite advancements, gaps remain in accurately forecasting earthquake magnitudes with sufficient lead time for mitigation. Our research aims to address this by using advanced machine learning and neural networks, building on previous studies^{14–17,52} to improve forecast accuracy and response efforts in Southern California.

Dataset

We utilized earthquake data from the Southern California Earthquake Data Center (SCEDC), managed by the California Institute of Technology⁵³.

For the Los Angeles region, using data from⁵⁴, we filtered events by latitude 34.0522, longitude -118.2437 within a 100 km radius, focusing on earthquake events from January 1, 2012, 00:00:00 to September 01, 2024, 00:00:00. All magnitude types and depths were included. We selected the XML output format and subsequently converted the dataset to CSV. The dataset includes 4 different magnitude types: M_l , M_h , M_w , M_{lr} . The counts for each type are as follows: 23,284 events for M_l , 52 for M_h , 52 for M_w , and 2 for M_{lr} .

According to the Southern California Earthquake Data Center (SCEDC) maintained by the California Institute of Technology⁵⁵, starting at the end of December 2015, SCSN began calculating an additional magnitude type, labeled Revised Local Magnitude (M_{lr}). M_{lr} magnitudes are only calculated for events with M_L between 3.0 and 6.0, and are obtained by applying a linear adjustment to the M_L value. The adjustment is designed to bring initial magnitude values derived from M_L into closer agreement with Moment Magnitude (M_w) values, because M_w is expected to be the preferred magnitude type for events above magnitude 3⁵⁵.

For most areas in Southern California, M_L is systematically larger than M_w for magnitudes greater than 3.5; consequently, the M_{lr} adjustment is a reduction of the M_L value of up to 0.5 units (larger adjustment for larger events). M_{lr} is calculated using the following formula⁵⁵:

$$M_{lr} = M_L \times 0.853 + 0.40125 \quad (1)$$

Solving for M_L , the formula becomes:

$$M_L = \frac{M_{lr} - 0.40125}{0.853} \quad (2)$$

For most earthquakes, M_L is the preferred magnitude for events smaller than 3.5, while M_w is preferred for events greater than 3.5⁵⁵.

For our dataset from the California Earthquake Data Center (SCEDC), we converted every magnitude type to M_L using SCEDC's formulas to ensure consistency and facilitate comparative analysis.

We removed the M_h values from our dataset because their conversion to M_L is not clearly defined. The M_h magnitude type is considered a non-standard method, typically used in situations where standard magnitude calculation methods are not applicable or when a magnitude is assigned temporarily until a more accurate value is determined⁵⁶. Since M_h values constitute only 0.22% of the total events in our dataset, their exclusion does not significantly impact our analysis.

Conversion of earthquake magnitudes to local magnitude (M_L)

To ensure consistency across all seismic data, we converted all earthquake magnitudes in our dataset to the local magnitude (M_L). The dataset obtained from the Southern California Earthquake Data Center (SCEDC) included various magnitude types: Local Magnitude (M_L), Revised Local Magnitude (M_{lr}), Moment Magnitude (M_w), and a non-standard magnitude type (M_h). This subsection outlines the methodology and formulas used for these conversions and provides examples to clarify the process.

Conversion of revised local magnitude (M_{lr}) to local magnitude (M_L)

The Revised Local Magnitude (M_{lr}) is an adjusted form of M_L , intended to align more closely with the Moment Magnitude (M_w) for events where M_L is systematically larger than M_w , particularly for magnitudes greater than 3.5. The formula provided by SCEDC for converting M_{lr} to M_L is as follows⁵⁵:

$$M_{lr} = M_L \times 0.853 + 0.40125 \quad (3)$$

Rearranging to solve for M_L :

$$M_L = \frac{M_{lr} - 0.40125}{0.853} \quad (4)$$

For example, consider the event recorded on May 9, 2021, with a magnitude type of M_{lr} and a value of 3.45:

$$M_L = \frac{3.45 - 0.40125}{0.853} \approx 3.57 \quad (5)$$

Thus, the local magnitude (M_L) for this event is approximately 3.57.

Conversion of moment magnitude (M_w) to local magnitude (M_L)

The Moment Magnitude (M_w) is often preferred for events greater than magnitude 3.5. SCEDC provides the same adjustment formula to convert M_w to M_L , given that M_{lr} is already an estimation aligned with M_w . Therefore, we use the following formula for the conversion:

$$M_L = \frac{M_w - 0.40125}{0.853} \quad (6)$$

Consider the event recorded on September 13, 2021, with a magnitude type of M_w and a value of 3.62:

$$M_L = \frac{3.62 - 0.40125}{0.853} \approx 3.77 \quad (7)$$

Thus, the local magnitude (M_L) for this event is approximately 3.77.

Exclusion of non-standard magnitude (M_h)

The dataset also included a small percentage (0.22%) of events with a non-standard magnitude type (M_h), defined as a temporary designation used when standard methods are not applicable or when the magnitude is not yet finalized³⁶. Since there is no clear conversion method from M_h to M_L , we excluded these events from our analysis to maintain consistency and accuracy.

The exclusion of M_h events is justified by their negligible proportion in the dataset, ensuring that the removal does not affect the overall dataset's integrity.

Justification for the conversion approach

The conversion of all magnitudes to the local magnitude (M_L) ensures a uniform dataset that facilitates comparative analysis and improves the reliability of our earthquake prediction models. By using the established formulas from SCEDC, we minimize errors associated with mixed magnitude types and ensure that all data align with the preferred magnitude type for the region. This approach allows for consistent modeling and a robust understanding of seismic activity in the Los Angeles area.

Criteria for dataset screening and selection of the study parameters

To accurately predict earthquake occurrences in the Los Angeles area, we screened the dataset obtained from the Southern California Earthquake Data Center (SCEDC) based on two main criteria: spatial extent and temporal range. This section clarifies the reasons for selecting a 100 km radius around Los Angeles and explains why we focused on earthquake events from January 1, 2012, to September 1, 2024.

Selection of the 100 km radius

A 100 km radius was chosen to encompass a broad area around Los Angeles that is highly relevant for earthquake forecasting. This distance is appropriate for several reasons:

- **Seismic relevance:** Los Angeles is located near multiple active fault lines, including the San Andreas Fault, the Newport-Inglewood Fault, and the San Jacinto Fault. These faults are known to produce significant seismic activity that could affect the city and its surrounding areas. A 100 km radius captures seismic events originating from these faults, providing a comprehensive dataset to analyze patterns and predict future earthquakes that might impact the region.
- **Urban and infrastructure impact:** A radius of 100 km ensures that the dataset includes all earthquakes that could potentially impact the densely populated urban center of Los Angeles and its critical infrastructure. Studies have shown that even moderate earthquakes within this distance can cause substantial damage due to the proximity of fault lines to the city, the nature of the underlying geological structures, and the complex interplay between seismic waves and urban environments.
- **Data sufficiency and model accuracy:** Using a radius smaller than 100 km could exclude significant seismic events that contribute to the overall understanding of earthquake patterns in the region. Conversely, a radius much larger than 100 km could introduce noise by including data from areas with different seismic characteristics, potentially reducing the predictive accuracy of our models. Therefore, a 100 km radius provides an optimal balance, ensuring sufficient data without compromising the model's relevance and accuracy.

Temporal range: data from 2012 onwards

We chose to focus on earthquake data from January 1, 2012, to September 1, 2024, for several reasons:

- **Computational efficiency:** Analyzing data over an extended period can increase the computational burden significantly. The selected timeframe balances the need for a comprehensive dataset with the practical considerations of computational efficiency. It includes 23,284 recorded events, which is a substantial sample size for training and validating machine learning models while avoiding excessive computational demands.
- **Consistency in magnitude types:** From 2012 onwards, the SCEDC dataset primarily uses a consistent magnitude type, specifically the local magnitude (M_L). Before this period, there were more varied magnitude types, such as duration magnitude (M_d) and network magnitude (M_n), for which conversions to M_L are not clearly defined. Focusing on data from 2012 onwards ensures uniformity in magnitude types, reducing potential errors or inconsistencies that could arise from conversions and thereby improving the reliability of the model.
- **Sufficient data volume:** The period from 2012 to 2024 provides a large enough dataset (23,284 events) to capture a wide range of seismic activities, from minor tremors to significant earthquakes. This timeframe encompasses a diverse set of seismic events, including aftershocks and foreshocks, allowing for a comprehensive analysis and the development of robust predictive models. The selected period is adequate to establish meaningful patterns and trends in earthquake activity for the Los Angeles area. By focusing on a 100 km radius and

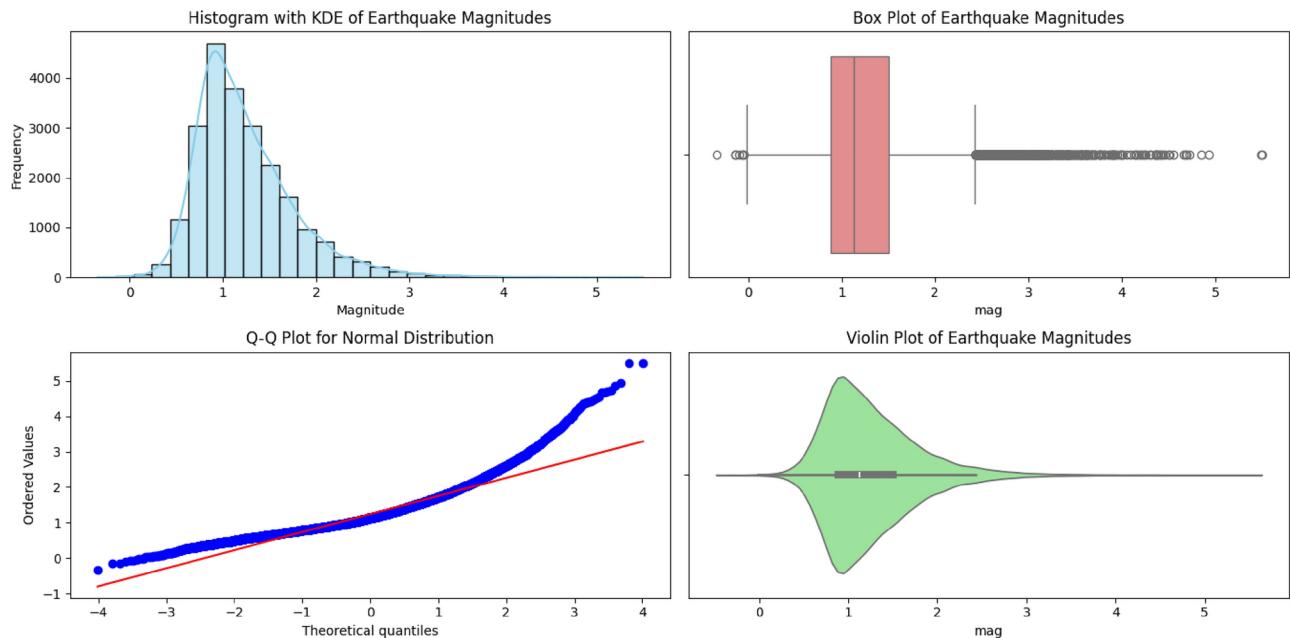


Fig. 1. Visual analysis of earthquake magnitudes: histogram with KDE, Box Plot, Q-Q Plot, and violin plot

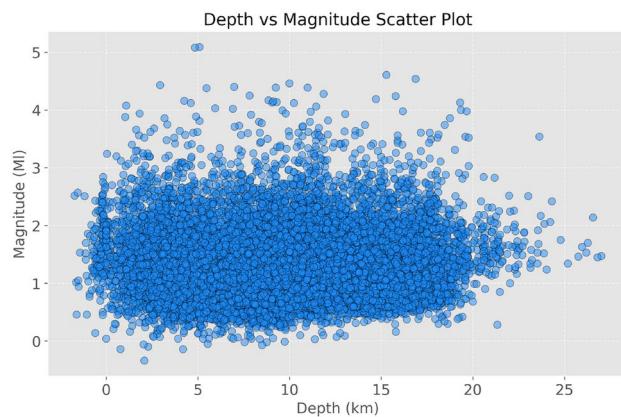


Fig. 2. Depth vs magnitude

selecting a temporal range from 2012 onwards, we aimed to strike a balance between data relevance, consistency, and computational manageability. This approach allows us to build accurate and efficient earthquake prediction models tailored specifically to the seismic characteristics of the Los Angeles region.

Statistical analysis of earthquake magnitudes

To gain deeper insights into the distribution of earthquake magnitudes in the Los Angeles region, we conducted a comprehensive statistical analysis of the dataset. This analysis involved calculating various descriptive statistics and performing statistical tests to determine the underlying distribution of the data.

Descriptive statistics

The dataset contains earthquake events recorded between January 1, 2012, and September 1, 2024, with magnitudes ranging from 0 to 5.50. The mean magnitude of these events is 1.24, and the median magnitude is 1.13, indicating a slight positive skew in the data. The standard deviation is 0.53, reflecting some variability in the magnitudes of the earthquakes. The range of the dataset is 5.84, and the interquartile range (IQR) is 0.62, which shows that the majority of the earthquake magnitudes are concentrated within a relatively narrow band.

The skewness of 1.41 and kurtosis of 3.66 suggest that the distribution is moderately right-skewed with heavier tails than a normal distribution. This is visually supported by the Histogram with Kernel Density Estimate (KDE) and the Q-Q plot in Figure 1.

Statistical distribution tests

We conducted a D'Agostino and Pearson's test for normality to assess whether the earthquake magnitudes follow a normal distribution. The test statistic is 6337.48 with a p-value of 0.0000, indicating that the null hypothesis of normality is rejected. Additionally, we performed a Kolmogorov-Smirnov test for an exponential distribution, which yielded a test statistic of 0.5905 and a p-value of 0.0000. These results suggest that the data does not follow a normal or exponential distribution, further confirming the right-skewed nature observed in the descriptive statistics.

Visual analysis

Figure 1 provides several visual representations of the earthquake magnitude data. The histogram with KDE shows a right-skewed distribution, where most earthquake magnitudes are clustered between 0 and 2. The Q-Q plot for normality further illustrates deviations from a normal distribution, particularly in the tails, where observed values exceed the theoretical quantiles. The box plot highlights the presence of several outliers, and the violin plot confirms the concentration of values around the lower magnitudes, reinforcing the observations from the descriptive statistics and statistical tests.

Conclusion

The statistical analysis reveals that the earthquake magnitudes in our dataset do not follow a normal or exponential distribution but are moderately right-skewed with heavy tails. This information is critical for selecting appropriate models and statistical methods for further analysis, ensuring the robustness and validity of any predictive models developed from this data.

Exploratory data analysis

To gain insights into the characteristics and patterns of earthquake data in Los Angeles, we conducted an exploratory data analysis (EDA). We generated and analyzed the following visualizations:

- Depth vs magnitude scatter plot:** We generated a scatter plot to illustrate the relationship between earthquake depth and magnitude, as depicted in Fig. 2. This plot provides a visual representation of the data points, where each point corresponds to an earthquake event with a specific depth and magnitude. The scatter plot demonstrates that while there is a concentration of events with lower magnitudes across various depths, there are also noticeable occurrences of higher magnitudes at different depths. The importance of depth as a predictive feature is further highlighted by our feature selection analysis, where the rolling mean of depth (\bar{d}_i) emerged as the second most influential variable according to the Information Gain method.
- Earthquake count over time:** We created a line plot to show the number of earthquakes over time, aggregated monthly, aiding us in identifying trends, seasonality, or any unusual activity over the analyzed period, as depicted in Fig. 3.
- Geographic distribution of earthquakes:** We produced a scatter plot to map the geographical distribution of earthquakes, with magnitude represented by color, helping us identify locations with higher seismic activity, as illustrated in Fig. 4 and in Fig. 5.

Analysis of small and negative magnitude values

Upon reviewing the dataset obtained from the Southern California Earthquake Data Center (SCEDC), we identified a total of 10 earthquake events with negative magnitude values and 8,736 events with magnitudes less than 1.0. These values are derived directly from the SCEDC dataset, which includes comprehensive earthquake records within a 100 km radius of Los Angeles from January 1, 2012, to September 1, 2024.

Explanation for negative magnitudes: The presence of negative magnitudes in the dataset is not an error introduced during data processing or magnitude conversion. Negative magnitudes can occur in seismic datasets for various reasons. Typically, these values are associated with very small seismic events, such as microearthquakes or noise events, which are detected but have such low energy release that they fall below

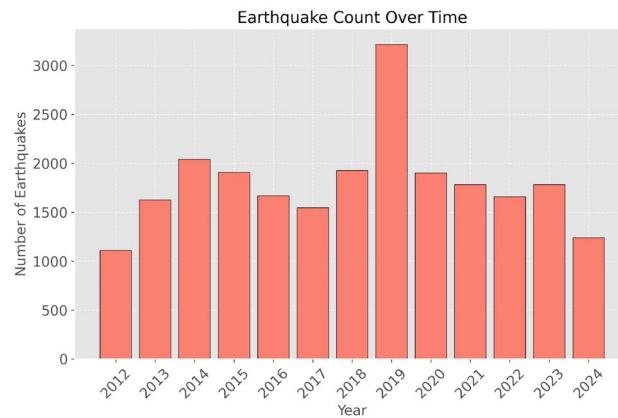


Fig. 3. Earthquake count over time

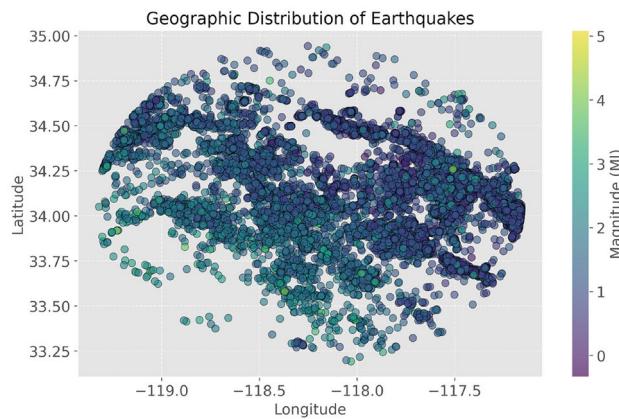


Fig. 4. Geographic distribution of earthquakes

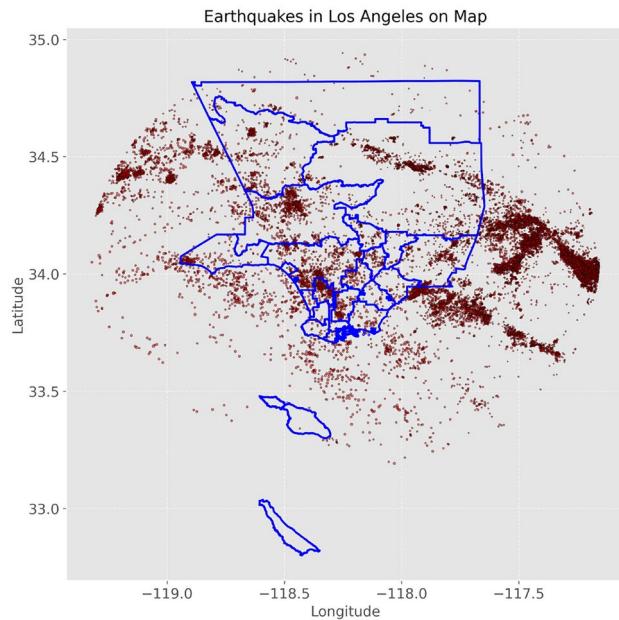


Fig. 5. Map of earthquakes in Los Angeles created using Python 3.9, with the libraries: pandas 1.3.5 (<https://pandas.pydata.org>), matplotlib 3.5.1 (<https://matplotlib.org>), seaborn 0.11.2 (<https://seaborn.pydata.org>), and geopandas 0.10.2 (<https://geopandas.org>). The shapefile for Los Angeles city boundaries was generated using QGIS 3.34.11-Prizren for macOS (<https://qgis.org>). The figure shows earthquake events filtered by latitude 34.0522, longitude -118.2437 within a 100 km radius, focusing on events from January 1, 2012, 00:00:00 to September 01, 2024, 00:00:00.

the standard zero-point reference for local magnitude calculations. Additionally, negative magnitudes may be a result of data adjustments or corrections applied by the data source, in this case, the SCEDC, to account for specific local conditions or calibration issues.

Explanation for small magnitudes: A significant portion of the dataset comprises events with magnitudes less than 1.0 (8,736 events), which is expected given the comprehensive nature of the SCEDC's earthquake catalog. Small magnitude earthquakes, often referred to as microearthquakes, are crucial for understanding seismic activity in a region. They provide insights into the background seismicity, allowing researchers to study fault activity, stress accumulation, and other geophysical processes that may not be apparent from larger events alone. Including small magnitude events in our analysis enables a more granular and detailed assessment of seismic activity patterns in the Los Angeles region.

Distribution of negative magnitudes: The negative magnitude events are sparsely distributed over the time period of the dataset. The occurrences span from October 9, 2015, to September 4, 2023, with only one instance per specific date. This distribution suggests that negative magnitudes are isolated cases and do not represent a systematic issue within the dataset.

Justification for inclusion: The decision to retain negative and small magnitude values in our dataset is based on the importance of these values for a comprehensive understanding of seismic activity. Removing these events would eliminate valuable data points that contribute to the overall analysis of seismic patterns in the Los Angeles region. Moreover, since these values are directly sourced from SCEDC, they adhere to the center's established data collection and reporting standards.

In summary, while the dataset includes some negative and small magnitude values, these records are both expected and scientifically meaningful. They provide a complete picture of the seismic activity in the Los Angeles region and help enhance the robustness of earthquake prediction models by including a full range of observed events.

Justification for choosing a 30-Day prediction period

The selection of a 30-day prediction period in our study was driven by a strategic decision to balance the need for timely alerts with the practical considerations of preparedness in densely populated urban areas. While many existing studies focus on shorter prediction periods, such as 7 days, we aimed to explore a longer timeframe that could offer significant benefits in the context of disaster management and public safety.

A 30-day prediction window provides a critical advantage for large cities, where the complexity of infrastructure, population density, and logistical challenges require more extended preparation periods for effective response. By providing forecasts over a 30-day horizon, our study seeks to enhance the lead time available to authorities and residents, thereby enabling more comprehensive planning, coordination, and mitigation strategies.

This approach is particularly relevant in the context of catastrophic earthquakes, such as the 1999 Izmit Earthquake, which struck northwestern Turkey, resulting in approximately 17,000 fatalities and leaving over 250,000 people homeless. The earthquake's impact was exacerbated by inadequate building codes and enforcement, leading to widespread structural failures⁵⁷. Similarly, the 2005 Kashmir Earthquake in Pakistan caused around 86,000 deaths and injured over 69,000 individuals. The region's mountainous terrain and the pre-existing poverty level significantly hampered rescue and recovery efforts, contributing to the high casualty rate⁵⁸.

The 2008 Sichuan Earthquake in China was one of the deadliest in recent history, with official reports indicating approximately 87,000 deaths and over 370,000 injuries⁵⁹. The earthquake's magnitude (Ms 8.0) and the high population density in affected areas played crucial roles in the casualty figures. Moreover, the collapse of poorly constructed buildings accounted for a significant portion of the fatalities, highlighting the importance of building standards in earthquake-prone regions⁶⁰. The response to the Sichuan earthquake was notable for its rapid mobilization of resources, which, despite the initial chaos, ultimately helped mitigate further casualties⁶¹.

The 2010 Haiti Earthquake, with a magnitude of 7.0, resulted in catastrophic consequences, with estimates of over 220,000 deaths and more than 300,000 injuries. The earthquake struck a densely populated area with many informal settlements, where buildings were not designed to withstand seismic activity. The lack of infrastructure and emergency services severely hampered rescue efforts, leading to a humanitarian crisis⁶². The international response was significant but faced challenges due to the scale of destruction and logistical difficulties in delivering aid⁶².

The 2004 Indian Ocean Earthquake and tsunami, which had a magnitude of 9.1, caused approximately 230,000 to 280,000 deaths across multiple countries, including Indonesia, Sri Lanka, India, and Thailand. The tsunami's rapid onset and the vast geographical area affected contributed to the high casualty figures. The disaster highlighted the need for improved early warning systems and disaster preparedness in coastal regions⁵⁸.

By focusing on a 30-day prediction period, our study contributes to the scientific community by exploring an alternative timeframe that could provide a meaningful extension of preparation time for such high-risk scenarios.

Data and model reproducibility

The dataset generated during the current study and analyzed to achieve our findings is publicly available on Zenodo⁶³.

Dataset

Obtain the dataset from the provided Zenodo link⁶³.

Machine learning model

Use the Random Forest algorithm, which is implemented in many popular machine learning libraries such as scikit-learn in Python.

Model parameters

- Number of Estimators: 100
- Random State: 15
- Test Size: 20% of the data should be used for testing.
- Data Scaling: Apply StandardScaler to standardize the features.

Feature sets

- All Features: Use all 19 in Table 4 features listed below to achieve an accuracy of 0.9769.
- Best 15-Variable Subset: Use the top 15 features in Table 9 to achieve an accuracy of 0.9797.

Defined target variable

For each earthquake event, we calculated the maximum magnitude of the subsequent seismic event occurring within the next thirty days. To transform this problem into a classification task, we categorized these magnitudes. Previous studies have demonstrated that an imbalanced dataset, caused by improper classification of the target variable, can significantly reduce the performance of machine learning models in earthquake prediction^{64–66}. To address this issue, we used the frequency distribution of the target variable to define intervals that ensure a balanced dataset. Our target variable for machine learning (ML) and neural network (NN) methods is the *maximum magnitude for the next 30 days*.

To determine the class boundaries, we employed the Jenks Natural Breaks optimization method, also known as the Natural Breaks classification method. This technique is specifically designed to identify natural groupings in data by minimizing variance within classes and maximizing variance between classes^{67–70}. By arranging values into distinct classes based on natural breaks in the data distribution, the method ensures that intervals accurately reflect the underlying data, leading to meaningful and interpretable classes for the target variable^{71–74}. This classification approach optimizes the classification thresholds to capture the inherent variability in the dataset, enhancing the interpretability and reliability of the results^{65,66}.

The Jenks optimization algorithm works by iteratively adjusting class boundaries to minimize the total sum of squared deviations (TSSD) within each class⁷⁵. Let n be the number of data points, k be the number of classes, and x_i be the data points where $i = 1, 2, \dots, n$. The steps of the algorithm are as follows:

1. Sort the data points x_i in ascending order.
2. Divide the sorted data into k initial classes.
3. Calculate the sum of squared deviations (SSD) from the class mean for each class:

$$SSD_j = \sum_{x_i \in C_j} (x_i - \mu_j)^2 \quad (8)$$

where C_j is the j -th class and μ_j is the mean of the j -th class.

4. Compute the total sum of squared deviations (TSSD) for all classes:

$$TSSD = \sum_{j=1}^k SSD_j \quad (9)$$

5. Adjust the class boundaries iteratively to minimize TSSD. By employing this method, we ensured that our target variable classes are well-defined and reflect the natural groupings in the earthquake data, thereby improving the performance and reliability of our predictive models. The results of this classification are presented in Table 1.

Evaluation of target variable classification

To address the concerns regarding the definition and balancing of the target variable, we employed the Jenks Natural Breaks classification method to categorize the maximum magnitudes of subsequent seismic events within the next thirty days. This method optimizes the class boundaries by minimizing variance within each class and maximizing variance between classes, thereby ensuring a more balanced distribution of the target variable.

To statistically evaluate the effectiveness of the Jenks Natural Breaks method, we compared it against two alternative classification methods: the Equal Interval and Quantile methods. The Equal Interval method divides the range of data into equal-sized intervals, while the Quantile method creates intervals based on data quantiles, ensuring each class contains an approximately equal number of data points.

Class	Boundaries	Number of Events
Class 1	0.65 - 2.82	4280
Class 2	2.82 - 3.25	6843
Class 3	3.25 - 3.67	5034
Class 4	3.67 - 4.13	3799
Class 5	4.13 - 4.72	2808
Class 6	4.72 - 5.50	574

Table 1. Natural Breaks Classification for Maximum Magnitude of Next Seismic Event in the Next 30 Days

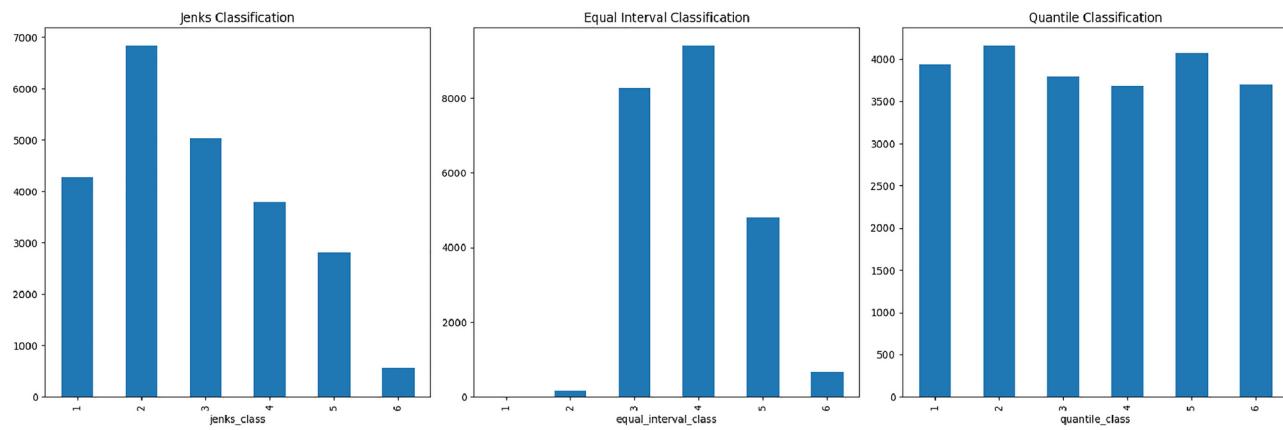


Fig. 6. Distribution of events across classes for Jenks Natural Breaks, Equal Interval, and Quantile classification methods. The Jenks method shows a more balanced distribution compared to Equal Interval while capturing the natural groupings more effectively than the Quantile method.

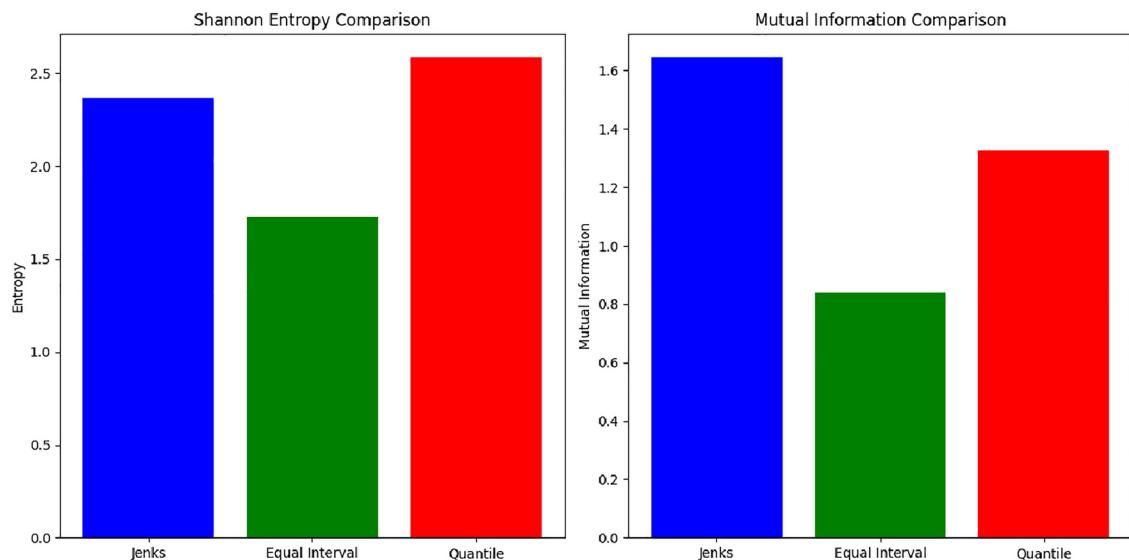


Fig. 7. Comparison of Shannon entropy and mutual information scores for different classification methods. The Jenks Natural Breaks method demonstrates superior performance in mutual information, indicating a better reflection of the inherent data structure.

Figure 6 illustrates the distribution of events across the six classes for each classification method. The Jenks Natural Breaks method shows a more balanced distribution of events compared to the Equal Interval method, which exhibits a highly skewed distribution, particularly in the lower and middle magnitude ranges. In contrast, the Quantile method results in an evenly distributed number of events per class, which might suggest a balanced dataset but does not take into account the natural groupings in the data.

To quantify the improvement in data balance, we calculated the Shannon entropy and mutual information for each classification method. Shannon entropy measures the uncertainty or randomness in the distribution of events across classes, where higher values indicate a more uniform distribution. Mutual information quantifies the amount of information gained about the original data distribution through the classification, with higher values indicating that the classification better reflects the natural patterns in the data.

The results, presented in Figure 7, show that the Jenks Natural Breaks method achieves a Shannon entropy value of 2.37, which is higher than the entropy for the Equal Interval method (1.73) but slightly lower than that for the Quantile method (2.58). However, the mutual information score for the Jenks Natural Breaks method (1.64) is significantly higher than that for the Equal Interval method (0.84) and also surpasses that for the Quantile method (1.33). These results indicate that while the Quantile method achieves a slightly more uniform distribution, the Jenks Natural Breaks method provides a better balance that reflects the inherent structure of the dataset, enhancing the interpretability and reliability of the classification.

By employing the Jenks Natural Breaks method, we ensured that the target variable classification is both meaningful and representative of the underlying data distribution, thereby improving the performance of our machine learning models in predicting seismic events.

Verification of target variable balancing using statistical tests

To further validate the balance of the target variable after applying the Jenks Natural Breaks optimization method, we conducted several additional statistical tests to assess the effectiveness of our classification approach.

1. Chi-square test for goodness of fit: We performed a Chi-Square test for goodness of fit to determine whether the observed frequencies of events in each class significantly differ from the expected frequencies under a uniform distribution. The results of the Chi-Square test yielded a Chi² statistic of 3458.13 with a p-value of 0.0000. As the p-value is less than 0.05, this indicates that there is a statistically significant difference between the observed and expected frequencies, confirming that the distribution of the target variable across classes is not uniform (see Figure 8).

2. Kolmogorov-Smirnov test: We also conducted a Kolmogorov-Smirnov (KS) test to compare the empirical distribution of the target variable with a theoretical normal distribution. The KS test resulted in a D statistic of 0.0881 with a p-value of 0.0000. The low p-value indicates that the empirical distribution of the target variable differs significantly from the theoretical normal distribution (see Figure 9). This supports the effectiveness of the Jenks Natural Breaks method in identifying natural groupings within the data rather than conforming to a normal distribution.

3. Gini coefficient: To measure inequality within the distribution of the classified target variable, we calculated the Gini coefficient. The resulting Gini coefficient is 0.2736, indicating a relatively low level of inequality in the distribution. This suggests that the dataset is fairly balanced across the different classes.

Conclusion: These statistical tests provide further evidence that the classification of the target variable using the Jenks Natural Breaks method captures the inherent structure and natural groupings within the data. The significant results from the Chi-Square and Kolmogorov-Smirnov tests highlight that the observed distribution is meaningfully different from a uniform or normal distribution, respectively, while the Gini coefficient value confirms a balanced yet representative classification.

Feature engineered input variables

We engineered new features to be used as input variables for our machine learning (ML) and neural network (NN) models. All the input variables, existing ones and engineered ones, can be seen in Table 4.

Rolling mean of depth (\bar{d}_i)

The rolling mean of the depth of earthquakes provides an average depth of seismic events over a specified period⁴. In this context, we calculated the rolling mean of the depth for the last 30 days to capture the average depth of recent seismic activity.

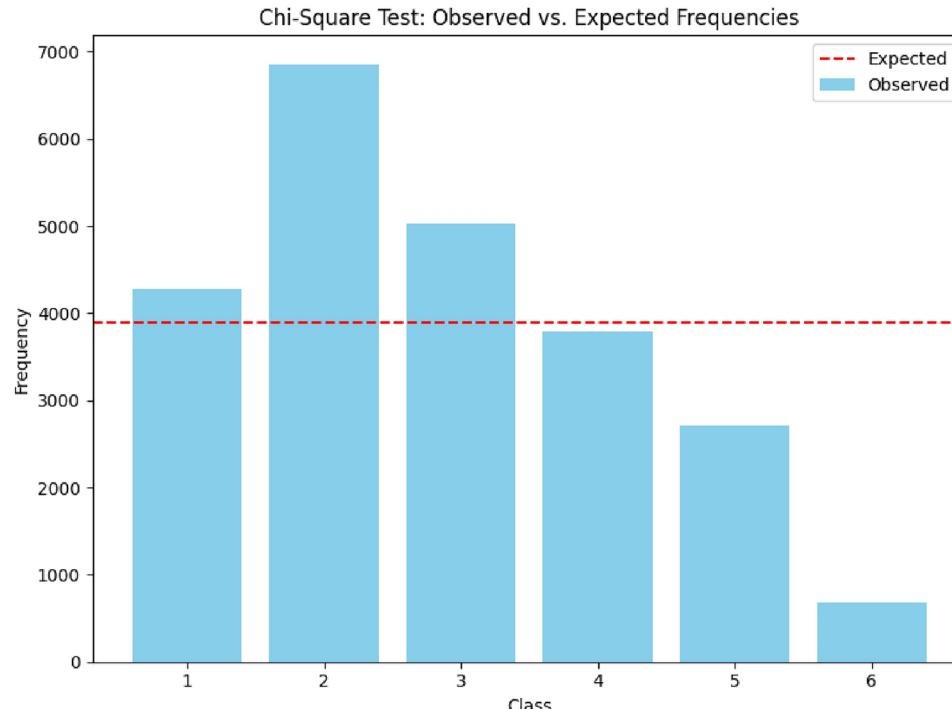


Fig. 8. Chi-square test: observed vs. expected frequencies. The significant deviation from the expected uniform distribution confirms the unique characteristics captured by the Jenks classification method.

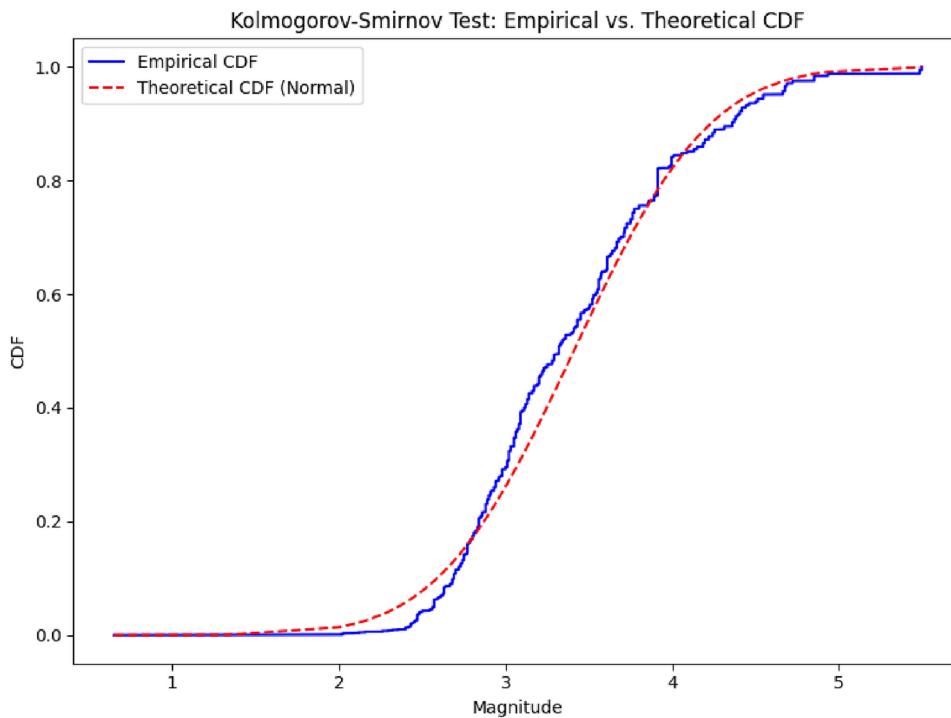


Fig. 9. Kolmogorov-Smirnov test: empirical vs. theoretical CDF. The empirical cumulative distribution function (CDF) significantly deviates from the theoretical normal CDF, highlighting the effectiveness of the Jenks Natural Breaks method in identifying natural groupings.

We computed the rolling mean of the depth for each earthquake event considering the last 30 days up to that event. Let d_i represent the depth of the i -th earthquake event.

The rolling mean depth \bar{d}_i for the last 30 days up to event i was calculated as follows:

$$\bar{d}_i = \frac{1}{N_i} \sum_j d_j \quad (10)$$

where N_i is the number of events in the last 30 days up to event i .

Time since last earthquake ($T_{\text{since last}}$)

We calculated the time elapsed since the last earthquake for each event. This feature helps to understand the intervals between successive earthquakes and provides insights into the frequency and recurrence patterns of seismic activity in Los Angeles. The dataset was sorted chronologically, and the time difference between each earthquake and its preceding event was computed¹⁰.

Gutenberg-richter b -value

We calculated the Gutenberg-Richter b -value for each event based on the previous 50 events, which is critical for understanding the distribution of earthquake magnitudes^{76,77}.

Gutenberg-richter law

The Gutenberg-Richter law states that the number of earthquakes $N(M)$ with magnitudes greater than or equal to M is given by:

$$N(M) = 10^{a-bM} \quad (11)$$

Logarithmic form

The logarithmic form is derived from the Gutenberg-Richter law (Equation 11), which describes the relationship between the number of earthquakes $N(M)$ with magnitudes greater than or equal to M .

Taking the logarithm (base 10) of both sides of Equation 11 gives:

$$\log_{10} N(M) = a - bM \quad (12)$$

This logarithmic form highlights the linear relationship between the logarithm of the number of earthquakes and the magnitude M , with the parameters a and b representing the seismicity rate and the magnitude-frequency scaling factor, respectively.

Cumulative distribution function (CDF) derivation

The cumulative distribution function (CDF) $F(M)$ represents the probability that an earthquake has a magnitude greater than or equal to M . It is derived from the Gutenberg-Richter law as follows:

1. The number of earthquakes $N(M)$ with magnitudes greater than or equal to M is given by:

$$N(M) = 10^{a-bM} \quad (13)$$

2. The total number of earthquakes N_{total} with magnitudes greater than or equal to the minimum magnitude M_{\min} is:

$$N_{\text{total}} = 10^{a-bM_{\min}} \quad (14)$$

3. The CDF $F(M)$ is the ratio of the number of earthquakes with magnitudes greater than or equal to M to the total number of earthquakes:

$$F(M) = \frac{N(M)}{N_{\text{total}}} \quad (15)$$

4. Substituting $N(M)$ and N_{total} into the equation and simplifying, we obtain:

$$F(M) = \frac{10^{a-bM}}{10^{a-bM_{\min}}} = 10^{-b(M-M_{\min})} \quad (16)$$

Thus, the cumulative distribution function (CDF) $F(M)$ is given by:

$$F(M) = 10^{-b(M-M_{\min})} \quad (17)$$

Probability density function (PDF)

Differentiating the CDF gives the probability density function (PDF):

$$f(M) = b \cdot 10^{-b(M-M_{\min})} \cdot \ln(10) \quad (18)$$

Derivation of the PDF from the CDF

The cumulative distribution function (CDF) of a random variable M , denoted $F(M)$, represents the probability that M takes on a value less than or equal to M :

$$F(M) = P(M \leq m) \quad (19)$$

The probability density function (PDF), denoted $f(M)$, is the derivative of the CDF:

$$f(M) = \frac{d}{dM} F(M) \quad (20)$$

For the Gutenberg-Richter law, the CDF $F(M)$ is given by:

$$F(M) = 10^{-b(M-M_{\min})} \quad (21)$$

To find the PDF $f(M)$, we differentiate $F(M)$ with respect to M :

1. Differentiate the exponential function:

$$F(M) = 10^{-b(M-M_{\min})} \quad (22)$$

Let $u = -b(M - M_{\min})$. Then $F(M) = 10^u$.

2. Apply the chain rule:

$$\frac{d}{dM} 10^u = 10^u \ln(10) \cdot \frac{du}{dM} \quad (23)$$

Since $u = -b(M - M_{\min})$, we have $\frac{du}{dM} = -b$.

3. Combine the results:

$$f(M) = 10^{-b(M-M_{\min})} \ln(10) \cdot (-b) \quad (24)$$

Simplifying, we get:

$$f(M) = b \cdot 10^{-b(M-M_{\min})} \cdot \ln(10) \quad (25)$$

Thus, the PDF is:

$$f(M) = b \cdot 10^{-b(M-M_{\min})} \cdot \ln(10) \quad (26)$$

Likelihood function

Given N earthquake magnitudes $\{M_1, M_2, \dots, M_N\}$, the likelihood function is the product of individual probabilities:

$$L(b) = \prod_{i=1}^N f(M_i) \quad (27)$$

Taking the natural logarithm of the likelihood function:

$$\ln L(b) = \sum_{i=1}^N \ln f(M_i) \quad (28)$$

Substituting the PDF into the log-likelihood function:

$$\ln L(b) = \sum_{i=1}^N (\ln b + \ln \ln(10) - b(M_i - M_{\min}) \ln(10)) \quad (29)$$

Simplifying:

$$\ln L(b) = N \ln b + N \ln \ln(10) - b \ln(10) \sum_{i=1}^N (M_i - M_{\min}) \quad (30)$$

Maximizing log-likelihood

Maximum Likelihood Estimation (MLE) is a statistical technique that provides a robust and accurate method for estimating the Gutenberg-Richter b-value, particularly in scenarios with limited data points or rare, large-magnitude earthquakes⁷⁸. Differentiating with respect to b and setting the derivative to zero:

$$\frac{d \ln L(b)}{db} = \frac{N}{b} - \ln(10) \sum_{i=1}^N (M_i - M_{\min}) = 0 \quad (31)$$

Solving for b :

$$b = \frac{N}{\ln(10) \sum_{i=1}^N (M_i - M_{\min})} \quad (32)$$

Mean magnitude

Since \bar{M} is the mean magnitude:

$$\bar{M} = \frac{1}{N} \sum_{i=1}^N M_i \quad (33)$$

The summation can be rewritten as:

$$\sum_{i=1}^N (M_i - M_{\min}) = N(\bar{M} - M_{\min}) \quad (34)$$

Substituting back

Substituting this back, we get:

$$b = \frac{1}{\ln(10)(\bar{M} - M_{\min})} \quad (35)$$

Final formula

Therefore, the b -value calculation formula is:

$$b = \frac{\log_{10} e}{\bar{M} - M_{\min}} \quad (36)$$

We used Equation 36 to calculate the b -value for each event.

Calculation of incremental b -values (Δb)

We first calculated the b -values using the fifty events that occurred prior to each event, as detailed in previous studies^{79,80}. This methodology allows us to track changes in seismicity over time, serving as predictive features for seismic activity analysis. After obtaining the b -values, we calculated the incremental b -values by determining the differences between b -values over various time windows, specifically between events i and $i - 2$, $i - 2$ and $i - 4$, $i - 4$ and $i - 6$, $i - 6$ and $i - 8$, and $i - 8$ and $i - 10$.

The study by Volant et al.⁷⁹, explores the relationship between seismic activity, aseismic deformation, and brittle failure within a geological structure subjected to fluid extraction. This study investigates the induced seismic activity and aseismic displacements resulting from gas extraction in an area previously devoid of displacement, shedding light on the impact of fluid extraction on seismicity and deformation processes⁷⁹. The study offers valuable insights that could inform the development of predictive models incorporating incremental b -values derived from seismic data analysis over time.

The study by Yousefzadeh et al.⁸⁰ investigates the effect of spatial parameters on the performance of machine learning algorithms for predicting the magnitude of future earthquakes in Iran. This study compares the performance of conventional methods such as Support Vector Machine (SVM), Decision Tree (DT), and Shallow Neural Network (SNN) with a contemporary Deep Neural Network (DNN) method. One of the key parameters introduced in this study is the Fault Density (FD), which, along with incremental b -values, enhances the accuracy of the earthquake prediction models. The results showed that incremental b -values, which measure the change in seismicity over time, significantly contribute to the prediction accuracy of earthquakes. The study highlights the importance of using both temporal and spatial parameters, including incremental b -values, in developing robust predictive models for seismic activity⁸⁰.

We detailed the methodology for calculating the incremental b -values, which served as predictive features for seismic activity analysis. We derived the incremental b -values from the differences in b -values calculated over various time windows.

We calculated incremental b -values for each event^{81,82}.

 b -value increments between events i and $i - 2$

We calculated the b -value increment between events i and $i - 2$ as follows:

$$\Delta b_{i,i-2} = b_i - b_{i-2} \quad (37)$$

where b_i is the b -value at event i and b_{i-2} is the b -value at event $i - 2$.

 b -value increments between events $i - 2$ and $i - 4$

We calculated the b -value increment between events $i - 2$ and $i - 4$ as follows:

$$\Delta b_{i-2,i-4} = b_{i-2} - b_{i-4} \quad (38)$$

where b_{i-2} is the b -value at event $i - 2$ and b_{i-4} is the b -value at event $i - 4$.

b-value increments between events $i - 4$ and $i - 6$

We calculated the b -value increment between events $i - 4$ and $i - 6$ as follows:

$$\Delta b_{i-4,i-6} = b_{i-4} - b_{i-6} \quad (39)$$

where b_{i-4} is the b -value at event $i - 4$ and b_{i-6} is the b -value at event $i - 6$.

b-value increments between events $i - 6$ and $i - 8$

We calculated the b -value increment between events $i - 6$ and $i - 8$ as follows:

$$\Delta b_{i-6,i-8} = b_{i-6} - b_{i-8} \quad (40)$$

where b_{i-6} is the b -value at event $i - 6$ and b_{i-8} is the b -value at event $i - 8$.

b-value increments between events $i - 8$ and $i - 10$

We calculated the b -value increment between events $i - 8$ and $i - 10$ as follows:

$$\Delta b_{i-8,i-10} = b_{i-8} - b_{i-10} \quad (41)$$

where b_{i-8} is the b -value at event $i - 8$ and b_{i-10} is the b -value at event $i - 10$.

Maximum magnitude recorded during the previous week ($M_{\max}^{\text{last week}}$)

The maximum magnitude recorded during the previous week is a crucial feature for assessing recent seismic activity and the potential for future earthquakes. Martinsson and Törnman⁸³ provide insights into the relationship between induced seismic activity and production rates, depth, and size within a mining context. Their study highlights that high seismic activity in a given week can increase the likelihood of elevated seismicity in the subsequent week, emphasizing the importance of monitoring and analyzing seismic events over short time intervals to assess evolving seismic activity patterns.

Bohnhoff et al.⁸⁴ discuss seismicity patterns following the Gutenberg-Richter law, indicating that a high-magnitude seismic event can be preceded by foreshocks. Monitoring the maximum magnitude recorded in a given period can offer valuable insights into the potential for larger seismic events.

Zhang et al.¹⁰ focus on feature extraction techniques for earthquake prediction, emphasizing the importance of identifying precursory patterns in seismic data. This study highlights the significance of monitoring the maximum magnitude as a key feature for predicting seismic events and understanding seismic activity trends.

Asim et al.⁹ explore earthquake prediction models using support vector regressors and hybrid neural networks, showcasing the capability of these methodologies in forecasting seismic events of specific magnitudes. Their study underscores the importance of advanced prediction techniques in assessing seismic hazards and the potential impact of earthquakes based on their magnitudes.

We used Maximum Magnitude Recorded During the Previous Week as an input feature for earthquake prediction^{85,86}. To determine the maximum magnitude recorded during the previous week, we implemented the following steps:

1. **Defining the time window:** We established a sliding window of 7 days (one week) for each event in the dataset.
2. **Identifying relevant events:** For each event i , we identified all seismic events occurring within the 7 days prior to the event i .
3. **Determining the maximum magnitude:** We then calculated the maximum magnitude among the identified events. This can be mathematically expressed as:

$$M_{\max}^{\text{last week}} = \max\{M_j \mid t_i - 7 \text{ days} \leq t_j < t_i\} \quad (42)$$

where $M_{\max}^{\text{last week}}$ represents the maximum magnitude recorded during the previous week, M_j is the magnitude of event j , t_i is the time of event i , and t_j is the time of event j .

Clustering coefficient (C_{clust})

The Clustering Coefficient, denoted as C_{clust} , is a measure of the degree to which points in a dataset tend to cluster together. In the context of our earthquake dataset, it can be used to quantify the spatial clustering of earthquake events. This can be particularly useful for understanding the spatial distribution and potential patterns of seismic activity^{85,87,88}.

Pairwise distance calculation

We computed the pairwise distances between all earthquake events using the Haversine formula. The Haversine formula is used to calculate the shortest distance between two points on the surface of a sphere, which is particularly useful for calculating distances on Earth.

The Haversine formula is derived from spherical trigonometry. Given two points on the Earth's surface, specified by their latitude and longitude (ϕ_i, λ_i) and (ϕ_j, λ_j) , the Haversine formula calculates the distance d_{ij} between these points as follows:

$$d_{ij} = 2R \arctan 2 \left(\sqrt{a}, \sqrt{1-a} \right) \quad (43)$$

where

$$a = \sin^2 \left(\frac{\Delta\phi}{2} \right) + \cos(\phi_i) \cos(\phi_j) \sin^2 \left(\frac{\Delta\lambda}{2} \right) \quad (44)$$

In these equations: - $\Delta\phi = \phi_j - \phi_i$ is the difference in latitude. - $\Delta\lambda = \lambda_j - \lambda_i$ is the difference in longitude. - ϕ and λ are the latitude and longitude of the events. - R is the Earth's radius (approximately 6371 km).

The Haversine formula accounts for the spherical shape of the Earth, providing a more accurate distance measurement compared to simple Euclidean distance calculations.

Adjacency matrix construction

We constructed an adjacency matrix A based on a distance threshold ϵ . If the distance between two events was less than ϵ , they were considered neighbors.

The adjacency matrix A is defined as:

$$A_{ij} = \begin{cases} 1 & \text{if } d_{ij} < \epsilon \\ 0 & \text{otherwise} \end{cases} \quad (45)$$

This matrix helps in identifying which earthquake events are neighbors based on the specified distance threshold ϵ .

Local clustering coefficient calculation

The Local Clustering Coefficient $C_{\text{clust},i}$ measures the extent to which the neighbors of a given node (earthquake event i) are themselves neighbors. It provides insight into the local density of connections around each earthquake event.

The Local Clustering Coefficient for an earthquake event i is calculated as:

$$C_{\text{clust},i} = \frac{2E_i}{k_i(k_i - 1)} \quad (46)$$

where:

- E_i is the number of edges (connections) between the neighbors of i .
- k_i is the number of neighbors of i (degree of i). The coefficient is normalized by the maximum possible number of edges between the neighbors, given by $k_i(k_i - 1)/2$ for an undirected graph. The factor of 2 in the numerator accounts for each edge being counted twice in an undirected graph. This formula gives a value between 0 and 1, where 1 indicates that all neighbors of i are directly connected (forming a complete subgraph), and 0 indicates no direct connections between neighbors.

A detailed example of the calculation for a specific earthquake event is provided in Appendix A.1 and in Appendix A.2.

Global clustering coefficient calculation

The Global Clustering Coefficient C_{clust} is the average of the Local Clustering Coefficients of all earthquake events up to that event, considering events within the last 30 days for each event.

The Global Clustering Coefficient C_{clust} is calculated as:

$$C_{\text{clust}} = \frac{1}{N_i} \sum_{i=1}^{N_i} C_{\text{clust},i} \quad (47)$$

where N_i is the number of events in the last 30 days up to event i .

This coefficient provides an overall measure of the clustering tendency of earthquake events within a specified time frame.

Standard deviation of magnitude for the last 30 days ($\sigma_{\text{mag},30}$)

The standard deviation of the magnitude of earthquakes provides a measure of the variability in earthquake strengths over a specified period^{89,90}. In this context, we calculated the standard deviation of the magnitude for the last 30 days to capture recent seismic activity variability.

We computed the standard deviation of the magnitude for each earthquake event considering the last 30 days up to that event. Let m_i represent the magnitude of the i -th earthquake event.

The mean magnitude $\mu_{\text{mag},30}$ for the last 30 days up to event i was calculated as follows:

$$\mu_{\text{mag},30} = \frac{1}{N_i} \sum_j m_j \quad (48)$$

where N_i is the number of events in the last 30 days up to event i .

The standard deviation $\sigma_{\text{mag},30}$ of the magnitude for the last 30 days up to event i was then calculated using the mean magnitude:

$$\sigma_{\text{mag},30} = \sqrt{\frac{1}{N_i} \sum_j (m_j - \mu_{\text{mag},30})^2} \quad (49)$$

Sum of the mean square deviation (η) from the regression line based on the Gutenberg-Richter law

We calculated the mean square deviation from the regression line to measure how well the observed data fit the Gutenberg-Richter (GR) law^{80,91}.

The sum of the mean square deviation (η) from the regression line based on the Gutenberg-Richter (GR) law is an essential metric for evaluating how well the observed earthquake data fits the GR model. This metric quantifies the variability and reliability of seismic activity predictions. Recent studies have shown that η provides valuable insights into seismic predictions⁹².

The sum of the mean square deviation (η) from the regression line based on the Gutenberg-Richter law also serves as an important predictor of future seismic events. By analyzing the deviations of observed seismicity data from the regression line defined by the Gutenberg-Richter law, researchers can gain insights into the consistency of seismic activity patterns and the likelihood of future earthquakes. The study by Fahandezhsadi and Sadi⁹³, titled “Earthquake Magnitude Prediction using Probabilistic Classifiers,” explores the use of the sum of the mean square deviation about the regression line as a feature for earthquake magnitude prediction, emphasizing its significance in assessing seismic activity trends and forecasting future events.

For a set of observed magnitudes $\{M_i\}$ and their corresponding cumulative counts $\{N_i\}$, the calculations are shown below:

Deviation from the regression line

The Gutenberg-Richter law states that the number of earthquakes $N(M)$ with magnitudes greater than or equal to M is given by:

$$N(M) = 10^{a-bM} \quad (50)$$

Taking the logarithm (base 10) of both sides, we obtain the linear form:

$$\log_{10} N(M) = a - bM \quad (51)$$

For a set of observed magnitudes $\{M_i\}$ and their corresponding cumulative counts $\{N_i\}$:

Observed Value: The observed number of earthquakes with magnitude M_i or greater is N_i . The logarithmic form of the observed count is:

$$\log_{10} N_i \quad (52)$$

Expected value: The expected (theoretical) number of earthquakes N_i^{theory} for a magnitude M_i , according to the Gutenberg-Richter law, is:

$$N_i^{\text{theory}} = 10^{a-bM_i} \quad (53)$$

Taking the logarithm (base 10) of the expected count:

$$\log_{10} N_i^{\text{theory}} = a - bM_i \quad (54)$$

Deviation calculation

The deviation d_i for each observed magnitude M_i is the difference between the observed logarithmic count $\log_{10} N_i$ and the expected logarithmic count $\log_{10} N_i^{\text{theory}}$.

Deviation formula:

$$d_i = \log_{10} N_i - \log_{10} N_i^{\text{theory}} \quad (55)$$

Substitute expected value: Substituting the theoretical value $\log_{10} N_i^{\text{theory}} = a - bM_i$ into the deviation formula:

$$d_i = \log_{10} N_i - (a - bM_i) \quad (56)$$

This deviation d_i represents how much the observed data deviates from the expected data according to the Gutenberg-Richter law.

Mean square deviation

The mean square deviation η is the average of the squares of these deviations. It is calculated as:

$$\eta = \frac{1}{n} \sum_{i=1}^n d_i^2 \quad (57)$$

Substituting the expression for d_i :

$$\eta = \frac{1}{n} \sum_{i=1}^n (\log_{10} N_i - (a - bM_i))^2 \quad (58)$$

where n is the number of data points.

Calculating the a -value

Given the total number of earthquakes N_{total} above the minimum magnitude M_{\min} , the a -value can be derived as follows:

- From the Gutenberg-Richter law:

$$N_{\text{total}} = 10^{a-bM_{\min}} \quad (59)$$

- Taking the logarithm of both sides:

$$\log_{10} N_{\text{total}} = a - bM_{\min} \quad (60)$$

- Solving for a :

$$a = \log_{10} N_{\text{total}} + bM_{\min} \quad (61)$$

- Thus, the a -value calculation formula is:

$$a = \log_{10} N_{\text{total}} + bM_{\min} \quad (62)$$

Difference between the largest observed magnitude and largest expected magnitude based on the Gutenberg-Richter law (ΔM)

Saichev and Sornette's study examines Båth's law, which empirically shows an average magnitude difference of 1.2, regardless of the mainshock magnitude⁹⁴. This reference highlights the importance of evaluating the difference between observed and expected magnitudes to accurately understand seismic activity and predict future events.

The difference between the largest observed magnitude and the largest expected magnitude based on the Gutenberg-Richter (GR) law, denoted as ΔM , is a crucial metric for evaluating seismic hazard. This metric helps identify regions where the observed seismicity deviates from the expected patterns, which is essential for assessing the potential for large, unexpected earthquakes. Recent studies have demonstrated that ΔM can provide significant insights into seismic hazard assessments⁹⁵.

We calculated the largest expected magnitude (M_{expected}) using the Gutenberg-Richter (GR) law⁹⁶. The Gutenberg-Richter law is expressed as:

$$\log_{10} N(M) = a - bM \quad (63)$$

To find the largest expected magnitude, we consider the equation when $N(M) = 1$ (i.e., the magnitude at which we expect to see one event, meaning the cumulative number of earthquakes with a magnitude greater than or equal to the greatest magnitude is one):

$$\log_{10}(1) = a - bM_{\text{expected}} \quad (64)$$

Since $\log_{10}(1) = 0$, the equation simplifies to:

$$0 = a - bM_{\text{expected}} \quad (65)$$

Solving for M_{expected} , we get:

$$M_{\text{expected}} = \frac{a}{b} \quad (66)$$

The difference ΔM between the largest observed magnitude (M_{observed}) and the largest expected magnitude (M_{expected}) is then calculated as:

$$\Delta M = M_{\text{observed}} - M_{\text{expected}} \quad (67)$$

Elapsed time between the last n Events (T)

Research by Faro et al.⁹⁷, investigates the influence of causal relationships on time perception and judgments of elapsed time between events. This study emphasizes the role of causal associations in shaping temporal judgments, providing insights into the predictive value of elapsed time between seismic events.

Furthermore, incorporating the elapsed time between events as a predictive feature in machine learning algorithms can enhance the models' ability to capture temporal dependencies in seismic activity. The study by Nguyen et al.⁹⁸ underscores the informative nature of elapsed time between events, suggesting its relevance in predictive modeling and decision-making processes.

We calculated the elapsed time (T) as the total time interval between the first and the last event within a specified window of n events^{99,100}. For a given set of events, let t_1 be the time of the first event and t_n be the time of the n -th event. The elapsed time T is determined as follows:

$$T = t_n - t_1 \quad (68)$$

where:

- t_1 is the time of the first event in the window,
- t_n is the time of the n -th event in the window.

Mean time between events (μ)

The study by Salam et al.¹⁰¹, includes the average time between events (μ) as one of the indicators for earthquake prediction. This reference underscores the importance of temporal features in predictive modeling and suggests that the mean time between events can be a valuable predictor for forecasting seismic events.

We calculated the mean time between events (μ) as the average time interval between consecutive earthquake events. For a given set of n events, let t_i be the time of the i -th event. The time interval between consecutive events is defined as:

$$\Delta t_i = t_{i+1} - t_i \quad (69)$$

The mean time between events is then calculated as the average of these time intervals:

$$\mu = \frac{1}{n-1} \sum_{i=1}^{n-1} \Delta t_i \quad (70)$$

where:

- t_i is the time of the i -th event,
- Δt_i is the time interval between the i -th event and the $(i + 1)$ -th event,
- n is the total number of events.

Coefficient of variation (CV)

The study by Rosenau and Oncken¹⁰², discusses the relationship between the coefficient of variation of recurrence intervals and seismic activity patterns in subduction zones. This reference emphasizes the importance of understanding the variability in recurrence intervals for predicting seismic events and assessing the frequency-size distribution of earthquakes in different geological settings.

We calculated the coefficient of variation (C) as the ratio of the standard deviation (σ) to the mean (μ) of the inter-event times^{82,103}. For a given set of n events, let Δt_i be the time interval between consecutive events. The mean (μ) and standard deviation (σ) of these time intervals are given by:

$$\mu = \frac{1}{n-1} \sum_{i=1}^{n-1} \Delta t_i \quad (71)$$

$$\sigma = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n-1} (\Delta t_i - \mu)^2} \quad (72)$$

The coefficient of variation (C) is then calculated as:

$$C = \frac{\sigma}{\mu} \quad (73)$$

where:

- μ is the mean time between events,
- σ is the standard deviation of the time intervals,
- n is the total number of events.

The square root of the cumulative seismic energy ($\sqrt{E_\Sigma}$)

Salam et al.¹⁰¹ conducted a study, which highlights the significance of energy-related features in earthquake prediction models. The study includes the square root of the released energy during a specific time as one of the indicators for earthquake prediction, supporting the notion that the rate of seismic energy release can be a valuable predictor for forecasting seismic events.

The rate of the square root of seismic energy is a metric used to quantify the energy released by seismic events. It provides a normalized measure of seismic activity by considering the energy release rate, which is important for understanding the dynamics of earthquake processes and assessing seismic hazards^{80,104}.

The seismic energy (E) released by an earthquake^{5,105} can be estimated using its magnitude (M) through the following relationship:

$$E = 10^{1.5M+4.8} \quad (74)$$

where:

- E is the seismic energy in joules,
- M is the magnitude of the earthquake. To calculate the square root of the cumulative seismic energy ($\sqrt{E_\Sigma}$), we first compute the seismic energy for each event using the formula $10^{1.5M_i+4.8}$. Then, we sum these values over a specified window of $n = 50$ events and take the square root of the sum. The expression is given by:

$$\sqrt{E_\Sigma} = \sqrt{\sum_{i=1}^{50} 10^{1.5M_i+4.8}} \quad (75)$$

where: - M_i is the magnitude of the i -th event in the window.

Sure, here is the subsection for the Number of Earthquakes in the Last 30 Days for each event:

Number of earthquakes in the last 30 days

The number of earthquakes in the last 30 days provides a measure of the frequency of seismic events over a specified period^{6,7}. In this context, we calculated the number of earthquakes for the last 30 days to capture the recent activity in seismic events.

We computed the number of earthquakes for each earthquake event considering the last 30 days up to that event. Let $N_{eq,30}$ represent the number of earthquake events in the last 30 days up to event i .

The number of earthquakes $N_{eq,30}$ for the last 30 days up to event i was calculated as follows:

$$N_{eq,30} = \sum_{j=i-N+1}^i I_{[t_j > t_i - 30 \text{ days}]} \quad (76)$$

where $I_{[t_j > t_i - 30 \text{ days}]}$ is an indicator function that equals 1 if the event j occurred within the last 30 days from event i , and 0 otherwise.

Explanation of predictor selection

The selection of predictor variables for our machine learning (ML) and neural network (NN) models was based on a comprehensive analysis of factors that contribute to earthquake occurrence and magnitude. Our goal was to construct a predictive model that captures spatial, temporal, and seismic characteristics, while ensuring that each variable contributes unique information relevant to earthquake forecasting.

Initially, we included a broad set of predictors (as shown in Table 2), encompassing variables related to the location and magnitude of earthquakes, temporal patterns, seismic energy, and statistical properties derived from earthquake catalogs. These predictors were carefully chosen to balance the representation of different domains that potentially influence earthquake behavior:

- **Spatial variables:** Latitude and Longitude were included to capture the geographic distribution of earthquakes.
- **Magnitude and temporal variables:** Magnitude, $M_{\max}^{\text{last week}}$ (maximum magnitude recorded in the last week), T (elapsed time between the last n events), and CV (coefficient of variation of inter-event times) were selected to account for both the strength and timing of past events.
- **Seismic characteristics:** Variables such as b -value, incremental b -values ($\Delta b_{i,i-2}$, $\Delta b_{i-2,i-4}$, $\Delta b_{i-4,i-6}$, $\Delta b_{i-6,i-8}$, and $\Delta b_{i-8,i-10}$), and C_{clust} (clustering coefficient) were included to reflect the seismicity patterns and clustering behavior observed in the earthquake catalog.
- **Statistical properties:** $\sigma_{\text{mag},30}$ (standard deviation of magnitude over 30 days), \bar{d}_i (rolling mean of depth), and $\sqrt{E_{\Sigma}}$ (square root of cumulative seismic energy) were incorporated to provide statistical measures of seismic activity over time.

Multicollinearity analysis and mitigation

To assess multicollinearity among the predictors, we conducted a Variance Inflation Factor (VIF) analysis. The VIF quantifies how much a variable is correlated with the other variables in the model. A high VIF indicates high correlation, suggesting that the predictor might be redundant.

Table 3 shows the initial VIF values for all predictor variables. The time-related variables $T_{\text{since last}}$, T , and μ exhibited infinite VIF values, indicating perfect correlation. To address this, we removed $T_{\text{since last}}$ (time since the last earthquake) and μ (mean time between events), retaining only T (elapsed time between the last n events) to preserve a meaningful temporal feature without introducing perfect multicollinearity.

No.	Variable	Explanation
1	Latitude	Latitude of the earthquake epicenter
2	Longitude	Longitude of the earthquake epicenter
3	\bar{d}_i	Rolling Mean of Depth
4	Magnitude	Magnitude of the earthquake
5	$T_{\text{since last}}$	Time elapsed since the last earthquake
6	b -value	Gutenberg-Richter b -value for the moment magnitude
7	$\Delta b_{i,i-2}$	Incremental b -value between events i and $i - 2$
8	$\Delta b_{i-2,i-4}$	Incremental b -value between events $i - 2$ and $i - 4$
9	$\Delta b_{i-4,i-6}$	Incremental b -value between events $i - 4$ and $i - 6$
10	$\Delta b_{i-6,i-8}$	Incremental b -value between events $i - 6$ and $i - 8$
11	$\Delta b_{i-8,i-10}$	Incremental b -value between events $i - 8$ and $i - 10$
12	$M_{\max}^{\text{last week}}$	Maximum magnitude recorded during the last week
13	C_{clust}	Clustering Coefficient
14	$\sigma_{\text{mag},30}$	Standard Deviation of Magnitude
15	η	Sum of the mean square deviation from the regression line
16	ΔM	Difference between the largest observed and expected magnitudes
17	T	Elapsed time between the last n events
18	μ	Mean time between events
19	CV	Coefficient of variation of the inter-event times
20	$\sqrt{E_{\Sigma}}$	Square root of the cumulative seismic energy
21	$N_{\text{eq},30}$	Number of Earthquakes in the Last 30 Days

Table 2. Our input variables before multicollinearity reduction. We removed $T_{\text{since last}}$ (time since the last earthquake) and μ (mean time between events), both shown in bold, retaining only T (elapsed time between the last n events) to preserve a meaningful temporal feature without introducing perfect multicollinearity.

Reduction of multicollinearity

To mitigate multicollinearity, we removed the time-related variables $T_{\text{since last}}$ and μ as their infinite VIF values indicated perfect correlation with other variables. Following this reduction, we recalculated the VIF values for the remaining predictor variables, as shown in Fig. 10.

The VIF analysis post-reduction shows that all remaining variables have VIF values well below the threshold of 10, indicating that multicollinearity has been effectively mitigated. The updated correlation matrix of predictor variables, as depicted in Figure 11, further confirms the reduced multicollinearity by illustrating a balanced and varied correlation pattern among the remaining predictors.

Impact on predictor selection

The removal of time-related variables ($T_{\text{since last}}$ and μ) addresses the issue of multicollinearity and enhances the robustness of the model. The retained predictor set now includes diverse variables that capture spatial, temporal, and seismic characteristics without redundant information. This approach improves model stability, interpretability, and overall performance.

The multicollinearity check also indicated that other variables such as latitude, longitude, and CV were not highly collinear with the rest, ensuring that all essential features remain in the model. These variables are crucial for understanding the geographical distribution of earthquakes, measuring the variability in seismic activity, and capturing the dynamics of seismic patterns over different scales.

By refining the predictor set through multicollinearity analysis, we enhance the predictive power of our ML and NN models while ensuring that each variable contributes unique information relevant to earthquake forecasting.

Methodology

This study focuses on evaluating a variety of machine learning (ML) algorithms and neural network (NN) models to predict the classification of earthquake magnitudes within the next 30 days. The models were trained and tested on a dataset of seismic events, with features scaled to optimize performance. Our input variables included a range of features designed to enhance predictive accuracy, as summarized in Table 4. By integrating these features, we aimed to develop robust machine learning models capable of forecasting the class of future seismic events in the Los Angeles area.

To ensure repeatability, we set a random state of 15 for all methodologies. The random state establishes the seed for the random number generator used in the algorithms, ensuring consistent results. Additionally, our test sample constituted 20% of the dataset throughout the project.

The machine learning algorithms we employed included Logistic Regression, Decision Trees, Random Forest, Gradient Boosting Machines (GBM), Support Vector Machines (SVM), k-Nearest Neighbors (k-NN), Naive Bayes, AdaBoost, XGBoost, and LightGBM.

Variable	VIF
Latitude	1.222523
Longitude	1.450631
Magnitude	1.236321
$T_{\text{since last}}$	∞
C_{clust}	3.231743
$\sigma_{\text{mag},30}$	2.551717
\bar{d}_i	2.363855
$N_{\text{eq},30}$	2.967707
b -value	3.419005
$\Delta b_{i,i-2}$	1.986764
$\Delta b_{i-2,i-4}$	3.944611
$\Delta b_{i-4,i-6}$	5.880192
$\Delta b_{i-6,i-8}$	7.788847
$\Delta b_{i-8,i-10}$	5.002879
$M_{\text{last week}}^{\text{max}}$	1.228443
η	3.041347
ΔM	4.144046
T	∞
μ	∞
CV	1.371371
$\sqrt{E_{\Sigma}}$	1.847086

Table 3. Initial VIF values for all predictor variables.

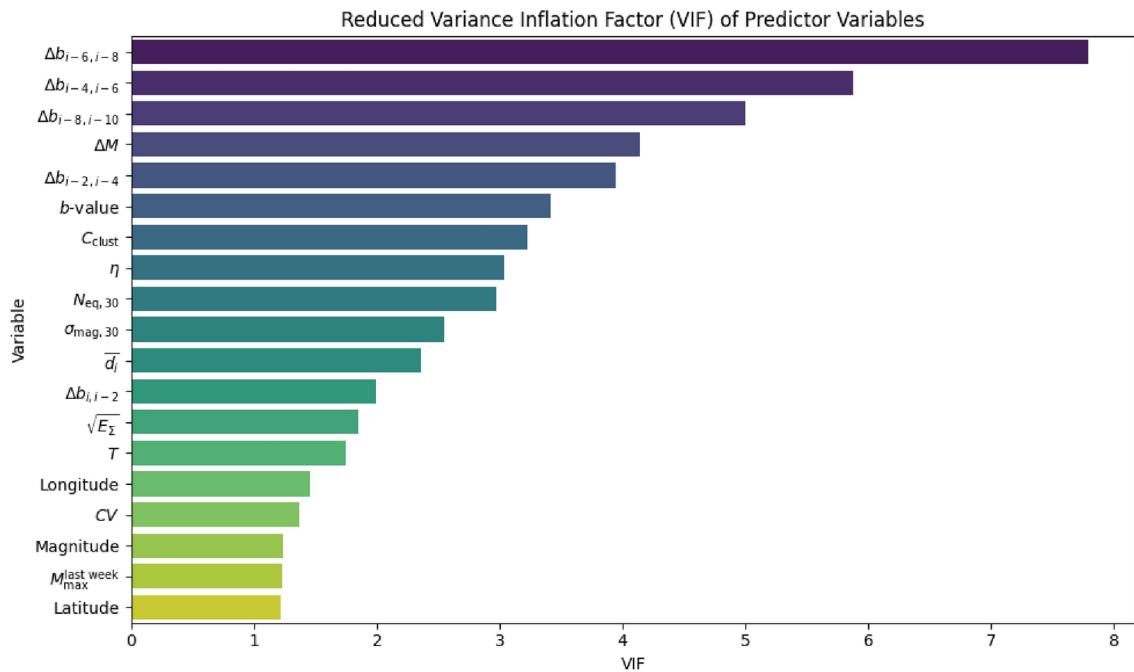


Fig. 10. Reduced variance inflation factor (VIF) of predictor variables. After removing the time-related variables with infinite VIF values, the remaining variables have VIF values below 10, indicating no significant multicollinearity.

Our neural network models encompassed Multilayer Perceptron (MLP), Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Long Short-Term Memory Networks (LSTM), Gated Recurrent Units (GRU), and Transformer Models. Our methodology diagram can be seen in Fig. 12³.

Exploratory data analysis for training and testing sets

An exploratory data analysis was conducted on both the training and testing sets to examine the spatial distribution of seismic events, the distribution of earthquake magnitudes, and the relationship between earthquake depth and magnitude.

Figure 13 and Figure 14 present the geographic distribution of seismic events for the training and testing sets, respectively. Each point on the scatter plots represents a seismic event, with color intensity corresponding to the earthquake's magnitude. These plots demonstrate the clustering of seismic activities in certain regions, emphasizing areas with higher seismic frequencies and magnitudes.

The distribution of earthquake magnitudes is illustrated in Figure 15 for the training set and Figure 16 for the testing set. Both plots show a left-skewed distribution, indicating a higher frequency of smaller-magnitude earthquakes and a lower frequency of larger-magnitude events.

To explore the relationship between earthquake depth and magnitude, depth versus magnitude scatter plots were created for both the training (Figure 17) and testing sets (Figure 18). These plots reveal that most earthquakes occurred at shallow depths, with magnitudes generally ranging from 0 to 3, while a few events with higher magnitudes were observed across various depths.

Overall, this analysis provides valuable insights into the characteristics and patterns of seismic events within the dataset, which is crucial for developing robust machine learning models.

Selection criteria for machine learning algorithms

The selection of the 16 algorithms for this study was guided by several key considerations to ensure a comprehensive evaluation of their performance in classifying the maximum earthquake magnitude within the next 30 days.

Diversity of models: The chosen algorithms represent a diverse range of machine learning techniques, encompassing both traditional machine learning models and neural network architectures. This diversity allows for a thorough exploration of different algorithmic approaches and their effectiveness in handling the specific characteristics of the earthquake dataset, such as non-linearity, temporal dependencies, and high dimensionality.

Relevance to the problem domain: Each of the selected algorithms was chosen for its relevance to specific aspects of the earthquake magnitude classification task:

- *Traditional machine learning models:* Algorithms such as Logistic Regression, Decision Trees, Random Forest, Gradient Boosting Machines (GBM), AdaBoost, XGBoost, and LightGBM were included for their robust performance in various classification tasks. These models are particularly effective at handling structured data and can capture complex patterns in the input features. Ensemble methods like Random Forest, GBM,

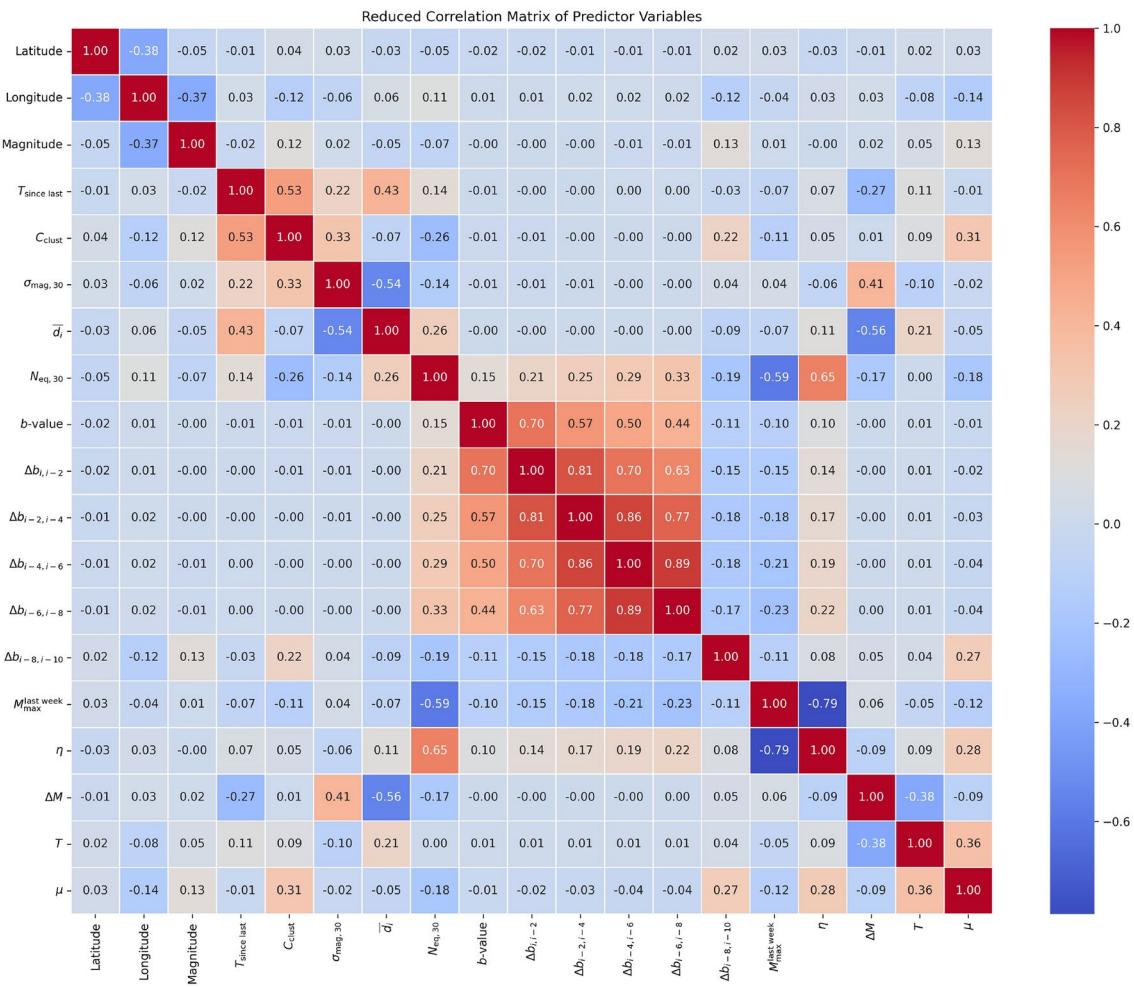


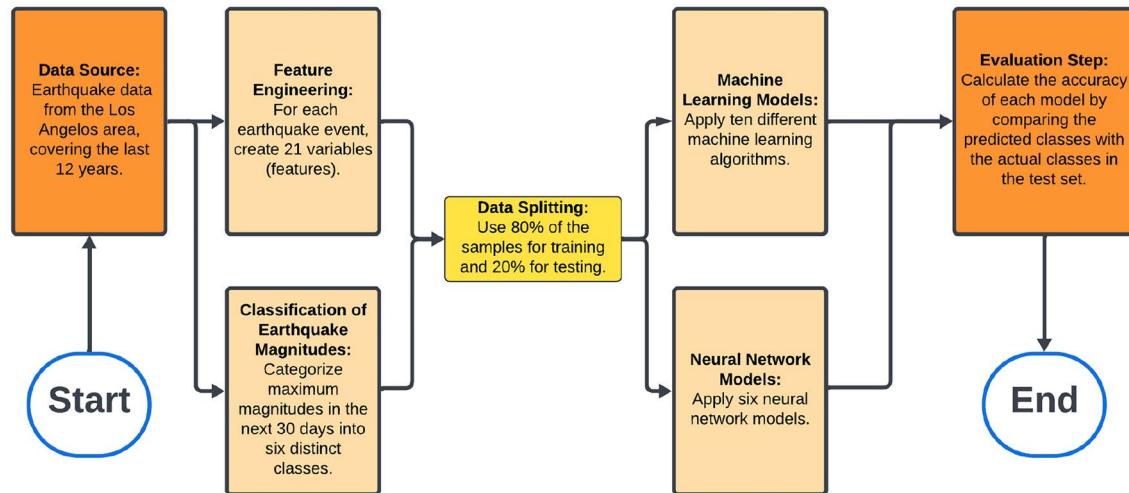
Fig. 11. Reduced correlation matrix of predictor variables. The matrix shows reduced correlations among the remaining predictor variables, confirming the effectiveness of the multicollinearity mitigation efforts.

XGBoost, and LightGBM offer additional advantages through their ability to combine multiple weak learners to create a more accurate and robust predictive model.

- **Neural network models:** The use of neural network architectures, including Multilayer Perceptrons (MLP), Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Long Short-Term Memory Networks (LSTM), Gated Recurrent Units (GRU), and Transformer models, is motivated by the complex and potentially non-linear nature of seismic data. For example, MLPs provide a baseline for deep learning approaches, while CNNs can capture spatial relationships in input data that may correspond to geographic features. RNNs, LSTMs, and GRUs are particularly relevant for their ability to model sequential and temporal dependencies inherent in time-series data like earthquake events, where past occurrences may influence future activity. LSTMs and GRUs, in particular, are designed to address the vanishing gradient problem found in traditional RNNs, allowing them to learn long-term dependencies that are crucial in understanding patterns over time in seismic data.
- **Incorporation of modern techniques:** In addition to traditional models, we incorporated modern techniques such as Transformer models. Transformers have demonstrated state-of-the-art performance in a wide range of sequence-based tasks due to their attention mechanisms, which allow them to focus on different parts of the input sequence. Unlike traditional RNNs, Transformers do not rely on sequential data processing, which makes them highly efficient in terms of parallel computation. This efficiency is particularly useful when dealing with large datasets, such as those in earthquake research. Their inclusion enables us to investigate their potential in capturing complex dependencies across different temporal and spatial dimensions in seismic data.

Comparative analysis: The inclusion of this wide range of models enables a comparative analysis to identify the most effective algorithm(s) for our specific problem. Each model's performance was evaluated on the same dataset with consistent preprocessing steps to ensure a fair comparison. The results of these comparisons are detailed in Table 5, where the accuracies of all selected models are presented.

No.	Variable	Explanation
1	Latitude	Latitude of the earthquake epicenter
2	Longitude	Longitude of the earthquake epicenter
3	\bar{d}_i	Rolling Mean of Depth
4	Magnitude	Magnitude of the earthquake
5	b -value	Gutenberg-Richter b -value for the moment magnitude
6	$\Delta b_{i,i-2}$	Incremental b -value between events i and $i - 2$
7	$\Delta b_{i-2,i-4}$	Incremental b -value between events $i - 2$ and $i - 4$
8	$\Delta b_{i-4,i-6}$	Incremental b -value between events $i - 4$ and $i - 6$
9	$\Delta b_{i-6,i-8}$	Incremental b -value between events $i - 6$ and $i - 8$
10	$\Delta b_{i-8,i-10}$	Incremental b -value between events $i - 8$ and $i - 10$
11	$M_{\text{max}}^{\text{last week}}$	Maximum magnitude recorded during the last week
12	C_{clust}	Clustering Coefficient
13	$\sigma_{\text{mag},30}$	Standard Deviation of Magnitude
14	η	Sum of the mean square deviation from the regression line
15	ΔM	Difference between the largest observed and expected magnitudes
16	T	Elapsed time between the last n events
17	CV	Coefficient of variation of the inter-event times
18	$\sqrt{E_{\Sigma}}$	Square root of the cumulative seismic energy
19	$N_{\text{eq},30}$	Number of Earthquakes in the Last 30 Days

Table 4. Our input variables for ML and NN models after multicollinearity reduction.**Fig. 12.** Our methodology.

By employing this diverse set of algorithms, we aim to determine the most suitable approach for classifying earthquake magnitudes, taking into account the unique characteristics of seismic data and the nature of the classification task.

Research methodology, model evaluation approach, and hyperparameter optimization

In our research, the primary objective was to obtain an initial overview of the performance of various machine learning (ML) and neural network (NN) models using general, commonly accepted parameter settings. The intent behind this approach was not to achieve the absolute best performance for each model but to evaluate their baseline capabilities under standard conditions. This strategy allows us to identify which models hold the most promise before committing to the more time-intensive process of fine-tuning.

By adopting this methodology, we aimed to understand the comparative strengths and weaknesses of each algorithm when applied to our specific dataset. This preliminary phase is critical for selecting the most suitable model(s) for more detailed investigation. Based on the initial evaluations, the Random Forest model emerged as the best-performing model, demonstrating superior accuracy and robustness compared to other algorithms.

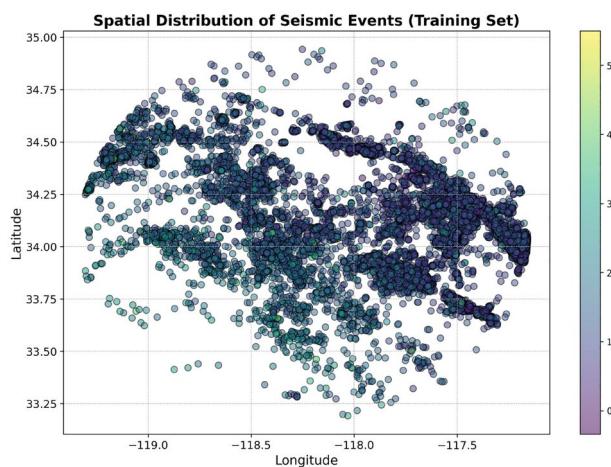


Fig. 13. Spatial distribution of seismic events (training set).

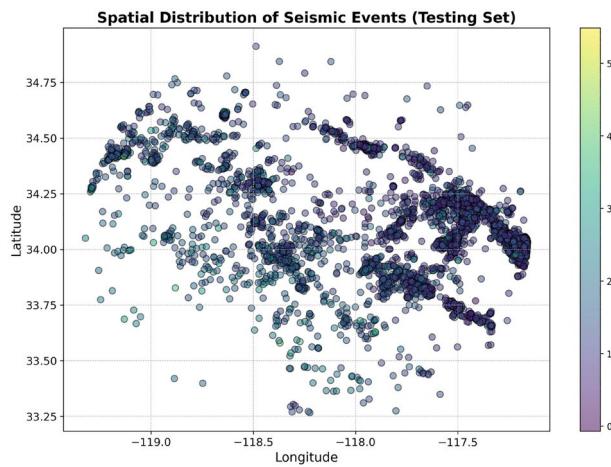


Fig. 14. Spatial distribution of seismic events (testing set).

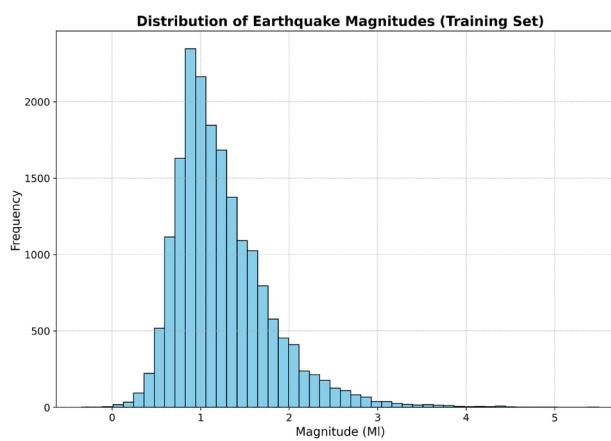


Fig. 15. Distribution of earthquake magnitudes (training set).

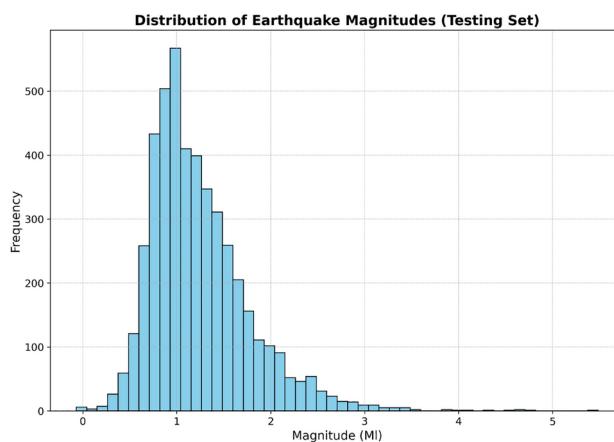


Fig. 16. Distribution of earthquake magnitudes (testing set).

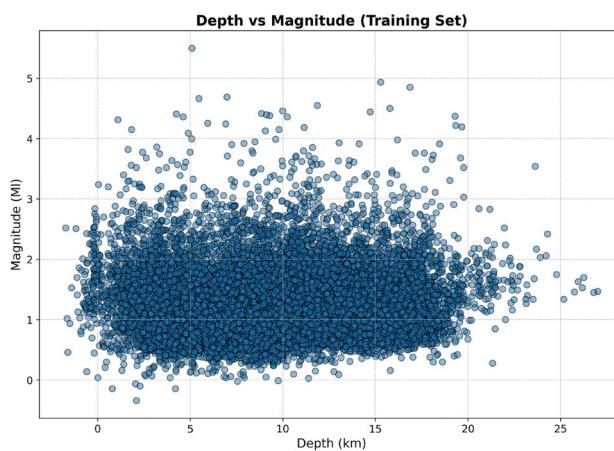


Fig. 17. Depth vs magnitude scatter plot (training set).

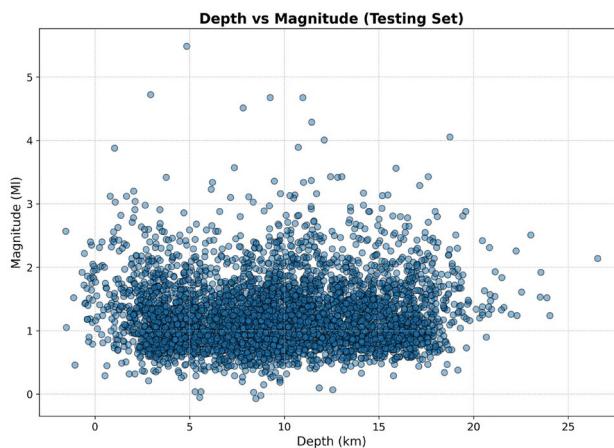


Fig. 18. Depth vs magnitude scatter plot (testing set).

#	Model	Accuracy
1	Random Forest	0.9769
2	XGBoost	0.9675
3	LightGBM	0.9336
4	Multilayer Perceptron	0.9260
5	Decision Trees	0.8967
6	Recurrent Neural Networks	0.7681
7	Gradient Boosting Machines	0.7009
8	k-Nearest Neighbors	0.6384
9	Convolutional Neural Networks	0.6262
10	Gated Recurrent Units	0.5360
11	Support Vector Machines	0.5207
12	Transformer Models	0.4596
13	Long Short-Term Memory Networks	0.4528
14	AdaBoost	0.3533
15	Logistic Regression	0.3264
16	Naive Bayes	0.2620

Table 5. Accuracy comparison of different models on the test dataset.

Following the identification of Random Forest as the most promising model, we proceeded to fine-tune its hyperparameters for a 15-variable subset. This fine-tuning process involved optimizing key parameters such as the number of trees (`n_estimators`), maximum depth (`max_depth`), and the minimum number of samples required to split a node (`min_samples_split`) and to be at a leaf node (`min_samples_leaf`). By refining these parameters, we aimed to enhance the model's predictive performance and achieve the highest possible accuracy for our specific application. The results and the detailed analysis of the fine-tuning process are discussed in the “Analysis of Fine-Tuned Random Forest Model” section.

Hyperparameter selection and optimization

The performance of machine learning models and neural networks is highly influenced by the choice of hyperparameters. In this study, several models were utilized, each with carefully chosen hyperparameters to optimize performance in predicting the maximum earthquake class in the coming 30 days. The following subsections provide a detailed description of the hyperparameters used for each model, along with the strategies employed for their optimization.

- Logistic regression:**Hyperparameters:**`solver='lbfgs'`, `max_iter=1000`, `random_state=15`.
-
- **Optimization:** The solver `lbfgs` was chosen for its efficiency in handling multi-class classification problems. The number of iterations was set to 1000 to ensure convergence. Decision tree classifier:**Hyperparameters:**`criterion='gini'`, `max_depth=None`, `random_state=15`.
-
- **Optimization:** The Gini impurity criterion was selected to measure the quality of splits. No maximum depth was specified to allow the tree to expand until all leaves are pure or contain fewer than the minimum samples required to split. Random forest classifier:**Hyperparameters:**`n_estimators=100`, `criterion='gini'`, `max_features='sqrt'`, `random_state=15`.
-
- **Optimization:** The number of trees (`n_estimators`) was set to 100 to balance computational cost and performance. The square root of the number of features was used to determine the maximum number of features considered for splitting. Gradient boosting machines (GBM):**Hyperparameters:**`learning_rate=0.1`, `n_estimators=100`, `subsample=1.0`, `random_state=15`.
-
- **Optimization:** The learning rate was set to 0.1, balancing convergence speed and model accuracy. The number of boosting stages (`n_estimators`) was set to 100. Support vector machines (SVM):**Hyperparameters:**`kernel='rbf'`, `C=1.0`, `gamma='scale'`, `random_state=15`.
-
- **Optimization:** The Radial Basis Function (RBF) kernel was selected due to its capability to handle non-linear relationships. The penalty parameter (`C`) was set to 1.0, and gamma was set to 'scale' to maintain balanced sensitivity. k-nearest neighbors (k-NN):**Hyperparameters:**`n_neighbors=5`, `weights='uniform'`, `metric='minkowski'`.
-
- **Optimization:** The number of neighbors (`n_neighbors`) was set to 5. A uniform weight was chosen, and the Minkowski distance was employed for distance computation. Naive bayes (GaussianNB):**Hyperparameters:** No hyperparameters require adjustment.
-

- **Optimization:** The Gaussian Naive Bayes method was used without further parameter tuning, as it does not have hyperparameters that significantly impact the model's performance. AdaBoost classifier:**Hyperparameters:**`n_estimators=50, algorithm='SAMME', random_state=15`.
-
- **Optimization:** The number of boosting rounds was set to 50. The SAMME algorithm was chosen for its ability to handle multi-class classification. XGBoost:**Hyperparameters:**`n_estimators=100, learning_rate=0.1, max_depth=6, random_state=15`.
-
- **Optimization:** The learning rate was set to 0.1, with 100 estimators to ensure a balance between speed and performance. The maximum depth of 6 was used to prevent overfitting. LightGBM:**Hyperparameters:**`num_leaves=64, max_depth=6, min_split_gain=0.5, min_child_samples=20, random_state=15`.
-
- **Optimization:** LightGBM hyperparameters were adjusted to avoid warnings and improve performance by balancing model complexity and overfitting risk. Multilayer perceptron (MLP):**Hyperparameters:**`hidden_layer_sizes=(100, 100), learning_rate_init=0.001, max_iter=1000, random_state=15`.
- **Optimization:** A multi-layer structure with two hidden layers, each containing 100 neurons, was used. The initial learning rate was set to 0.001 for gradual weight updates.

Neural network models

- Convolutional neural networks (CNN):**Hyperparameters:** Layers: Input, Conv1D (32 filters, kernel size 3), Flatten, Dense (64 units, ReLU), Dense (output layer, softmax).
-
- **Optimization:** Model architecture optimized through empirical testing to balance depth and computational efficiency. Recurrent neural networks (RNN):**Hyperparameters:** Layers: Input, LSTM (64 units, return sequences), Flatten, Dense (64 units, ReLU), Dense (output layer, softmax).
-
- **Optimization:** LSTM units were used to capture temporal dependencies, and additional layers were added to refine the feature representation. Long short-term memory networks (LSTM):**Hyperparameters:** Layers: Input, LSTM (64 units), Dense (64 units, ReLU), Dense (output layer, softmax).
-
- **Optimization:** Configured with an appropriate number of units and layers to optimize sequential pattern learning and minimize overfitting. Gated recurrent units (GRU):**Hyperparameters:** Layers: Input, GRU (64 units), Dense (64 units, ReLU), Dense (output layer, softmax).
-
- **Optimization:** GRU layers were chosen for their efficiency compared to LSTM, with configurations aimed at reducing computational complexity while retaining the ability to model long-term dependencies. Transformer models:**Hyperparameters:** Layers: Input, Dense (64 units, ReLU), Dense (output layer, softmax).
- **Optimization:** A simplified transformer architecture was employed to assess its effectiveness for this particular dataset, focusing on handling non-sequential data relationships. The model's structure was designed to optimize learning while maintaining computational efficiency. This combination of general parameter settings and strategic fine-tuning allowed us to achieve a comprehensive understanding of the performance potential of various machine learning and neural network models. Through this process, we identified the most promising models for further investigation and optimized the Random Forest model to achieve superior predictive accuracy for our specific application.

Random forest

We utilized a Random Forest model with 100 estimators to predict earthquake categories, and the resulting accuracy is presented in Table 5. Random Forest is a machine learning algorithm that has been successfully applied in seismology for earthquake prediction and analysis. Researchers have utilized Random Forest to develop models that can predict earthquakes by analyzing seismic data and identifying patterns that precede seismic activity¹⁰⁶. This approach represents a significant advancement in earthquake prediction methodologies, emphasizing data-driven techniques to enhance the reliability of seismic forecasts.

In seismology, Random Forest has been employed to distinguish seismic waveforms, enabling researchers to effectively differentiate between earthquake signals and background noise¹⁰⁷. By training Random Forest classifiers with a substantial dataset of earthquake and noise waveforms, researchers have created models capable of automatically extracting features and classifying seismic events with high accuracy¹⁰⁷. This application of Random Forest underscores its potential in improving earthquake early warning systems by facilitating rapid and precise identification of seismic events.

Random Forest has also been utilized to detect and classify seismic signals related to various geological phenomena, such as landslides and glacial earthquakes^{106,108}. By employing Random Forest classifiers, researchers have automated the process of identifying and categorizing seismic events, leading to more efficient monitoring and analysis of geological activities^{106,108}. This automated approach not only enhances the speed of event recognition but also improves the overall understanding of seismic processes in geologically active regions.

In the field of earthquake forecasting, Random Forest has shown promise in predicting the magnitude and occurrence of seismic events¹⁰⁸. Studies have indicated that Random Forest models can effectively forecast earthquake magnitudes in specific regions, providing valuable insights for disaster preparedness and risk mitigation strategies¹⁰⁸. This predictive capability highlights the potential of Random Forest in supporting decision-making processes related to earthquake response and mitigation efforts.

XGBoost

We applied XGBoost to predict earthquake categories, and the resulting accuracy is shown in Table 5. XGBoost, which stands for Extreme Gradient Boosting, is a machine learning algorithm renowned for its efficiency and effectiveness in handling complex datasets. In the field of seismology, XGBoost has shown significant promise in earthquake prediction and analysis. Seismologists have been exploring the application of machine learning techniques like XGBoost to enhance their ability to predict seismic events accurately and efficiently¹⁰⁹. By utilizing XGBoost, researchers have developed models capable of predicting earthquakes by analyzing various features and patterns in seismic data¹⁰⁹. This approach signifies a shift towards more advanced and data-driven methodologies in seismology, aiming to improve the accuracy and timeliness of earthquake predictions.

In the realm of earthquake prediction, XGBoost has been used alongside other artificial intelligence models to evaluate earthquake spatial probability, particularly in regions like the Arabian Peninsula¹¹⁰. The integration of XGBoost with explainable artificial intelligence (XAI) models has shown promising results, emphasizing the importance of including additional factors such as seismic gaps and tectonic contacts to enhance prediction accuracy¹¹⁰. This fusion of advanced machine learning techniques with traditional seismic analysis methods demonstrates a multidimensional approach to earthquake forecasting, highlighting the significance of comprehensive data analysis in seismology.

XGBoost has also been integrated into earthquake monitoring and early warning systems to provide real-time alerts before significant ground shaking occurs¹¹¹. By employing broadband P waveform data and XGBoost algorithms, researchers have developed systems capable of issuing earthquake warnings several seconds prior to the onset of a seismic event¹¹¹. This proactive approach to earthquake prediction underscores the potential of machine learning algorithms like XGBoost in improving seismic monitoring and disaster mitigation efforts.

Moreover, XGBoost has been incorporated into comprehensive earthquake prediction models, combining neural networks and other machine learning classifiers to analyze seismic data and forecast earthquake impacts¹¹².

LightGBM

We applied LightGBM to predict earthquake categories. The resulting accuracy is shown in Table 5.

LightGBM, a tree-based boosting algorithm, has been utilized in earthquake prediction within seismology. Researchers have successfully employed LightGBM to develop models capable of predicting earthquake magnitudes and mapping seismic vulnerability by leveraging artificial intelligence techniques¹¹³. By utilizing historical strong motion data from databases such as NGA-west2, LightGBM models have demonstrated the ability to swiftly and accurately replicate the distribution of strong motion near earthquake epicenters¹¹³. This application of LightGBM represents a significant advancement in earthquake prediction methodologies, showcasing the algorithm's efficiency in handling seismic data and enhancing predictive capabilities in seismology.

LightGBM has been acknowledged for its efficiency in data processing and memory consumption reduction, making it a valuable tool for analyzing seismic data in large sample applications¹¹⁴. The algorithm's capacity to enhance processing speed while maintaining accuracy is particularly advantageous in seismology, where timely analysis of seismic events is critical for effective earthquake prediction and risk assessment. By leveraging LightGBM's capabilities, researchers can streamline data processing tasks and improve the efficiency of earthquake prediction models.

In the realm of earthquake forecasting, LightGBM has played a crucial role in predicting seismic events and evaluating seismic vulnerability in earthquake-prone regions. Through the integration of LightGBM into predictive models, researchers have been able to analyze seismological parameters and forecast the areas impacted by earthquake-induced landslides using sophisticated data processing techniques¹¹⁵. This approach underscores the algorithm's versatility in handling complex seismic datasets and providing valuable insights into earthquake impacts, thereby aiding in disaster preparedness and risk mitigation efforts.

The incorporation of LightGBM into earthquake prediction models has enabled researchers to enhance the accuracy of seismic forecasts and deepen the understanding of seismic processes. By integrating LightGBM into comprehensive earthquake prediction frameworks, seismologists can leverage the algorithm's capabilities to analyze seismic data, identify seismic patterns, and forecast earthquake magnitudes with greater precision⁹.

Multilayer perceptron (MLP)

We employed a Multilayer Perceptron (MLP) to predict earthquake categories, and the resulting accuracy is detailed in Table 5. In seismology, the Multilayer Perceptron (MLP) neural network model has been utilized to predict earthquake magnitudes and assess seismic events accurately. Researchers have employed MLP to develop models capable of forecasting the magnitude of earthquakes, providing valuable insights into seismic activity¹¹⁶. By leveraging the capabilities of MLP, seismologists can analyze seismic data and predict earthquake magnitudes with enhanced precision, contributing to more effective disaster preparedness and risk mitigation strategies in earthquake-prone regions.

The application of MLP in seismology has enabled researchers to predict the magnitude of earthquakes using neural network models with multiple hidden layers¹¹⁶. By training MLP models with seismic data, researchers can extract patterns and features that aid in forecasting earthquake magnitudes, thereby improving the accuracy of seismic event predictions. This approach highlights the effectiveness of MLP in handling complex seismic datasets and enhancing the understanding of seismic processes in seismology.

MLP has been utilized to predict the occurrence of seismic events and assess earthquake magnitudes based on historical seismic data¹¹⁷. By employing MLP neural networks with backpropagation learning algorithms, researchers can analyze seismic patterns and predict the magnitude of earthquakes accurately. This utilization of

MLP in earthquake prediction models demonstrates the algorithm's effectiveness in handling seismic data and enhancing the reliability of seismic forecasts in seismology.

MLP has been applied in seismology to create earthquake prediction models that utilize artificial neural networks to forecast seismic events¹¹⁸.

Decision trees

We employed Decision Trees to predict earthquake categories, and the resulting accuracy is detailed in Table 5.

Decision Trees are a widely used machine learning algorithm in seismology for earthquake prediction and analysis. They are structured as tree-like models where each internal node represents a feature or attribute, each branch signifies a decision rule, and each leaf node indicates the outcome or prediction¹¹⁹. In seismology, Decision Trees have been effectively utilized to analyze seismic data, forecast earthquake magnitudes, evaluate seismic vulnerability, and categorize seismic events based on various parameters.

Researchers have applied Decision Trees in seismology to predict earthquake magnitudes and assess seismic vulnerability by creating models and predicting seismic events using a tree structure¹¹⁹. By developing Decision Trees based on seismic data, researchers can identify patterns and relationships that assist in predicting earthquake magnitudes and comprehending seismic processes. This methodology showcases the efficacy of Decision Trees in managing intricate seismic datasets and enhancing the precision of earthquake predictions in seismology.

Decision Trees have been utilized to assess parameters influencing earthquake damage and simulate earthquake damage distributions in seismically active regions¹²⁰. Through the application of Decision Tree techniques, researchers can holistically evaluate earthquake damages, considering both structural and non-structural factors, to accurately predict and model earthquake damage distributions. This use of Decision Trees underscores their adaptability in analyzing seismic data and forecasting the impact of seismic events on structures and infrastructure.

In the realm of earthquake prediction, Decision Trees have been employed to classify seismic events, differentiate between various types of seismic signals, and forecast the likelihood of earthquakes based on historical seismic data¹²¹. By leveraging Decision Trees, researchers can establish models that aid in decision-making during seismic events, enhance earthquake emergency response strategies, and refine earthquake forecasting methodologies. This utilization of Decision Trees demonstrates their effectiveness in analyzing seismic data and supporting decision-making processes in seismology.

Decision Trees have been integrated with other machine learning algorithms to predict earthquake occurrences, evaluate the seismic performance of structures, and enhance disaster planning and response strategies¹²².

Recurrent neural networks (RNN)

We employed Recurrent Neural Networks (RNN) to predict earthquake categories, and the accuracy results are shown in Table 5. Recurrent Neural Networks (RNN) have become a valuable tool in seismology for earthquake prediction and analysis. RNNs, a type of neural network that incorporates feedback loops, are well-suited for handling seismic time-series data due to their ability to capture temporal dependencies in sequential data¹²³. Researchers have successfully applied RNNs in seismology to model postseismic deformation, classify seismic events, and predict earthquake occurrences with improved accuracy, contributing to more effective disaster management strategies and risk mitigation efforts in earthquake-prone regions.

A study introduced a machine learning approach using RNNs to characterize the postseismic deformation of the 2011 Tohoku-Oki Earthquake based on time-series data, demonstrating the effectiveness of RNNs in accurately modeling observed seismic phenomena¹²³. By leveraging the capabilities of RNNs, researchers can analyze seismic data over time and predict the evolution of seismic events, providing valuable insights into the dynamics of seismic processes in seismology.

RNNs have been utilized in earthquake detection systems to analyze seismic array data and detect seismic events efficiently. A study focused on developing a graph-partitioning based CNN for earthquake detection using a seismic array, showcasing the effectiveness of RNNs in processing large-scale seismic network data sets and improving earthquake detection techniques¹²⁴. This application of RNNs highlights their ability to handle complex spatiotemporal data and enhance earthquake detection capabilities in seismology.

RNNs have also been integrated into seismic event classification models to analyze seismic waveforms and classify seismic events accurately. By leveraging RNNs for seismic event classification, researchers can extract features from seismic signals and categorize seismic events based on their characteristics, leading to more precise earthquake predictions and assessments in seismology¹²⁵.

Clarification of input dimensions for RNN

The architecture of our Recurrent Neural Network (RNN) was designed to process earthquake data where each feature represents a distinct time step in a sequence. The input to the RNN was structured as a 3D tensor with dimensions (number of samples, sequence length, number of features). Specifically, the dimension (number of features, 1) refers to the input shape for each individual sequence processed by the RNN.

In this context, number of features corresponds to the total number of attributes describing each earthquake event in our dataset, and the second dimension of 1 represents a single time step. The structure ensures that each feature is treated as part of a temporal sequence, even if the sequence length is one. This approach allows the RNN to learn temporal dependencies across the features, leveraging the recurrent connections to capture patterns that may be indicative of earthquake categories.

The choice of sequence length 1 was made because each feature in the dataset corresponds to a single time point, rather than a series of time points. Therefore, while the RNN typically processes longer sequences, in this case, the model learns from the inter-feature relationships within each event.

Clarification of LSTM layer usage in RNN architecture

In this study, the Recurrent Neural Network (RNN) architecture employed to forecast earthquake categories incorporated an LSTM (Long Short-Term Memory) layer. The LSTM layer is a specialized form of RNN that is particularly well-suited for learning long-term dependencies in sequential data. This choice was made to enhance the model's ability to capture temporal patterns within the earthquake data, thereby improving its predictive performance.

It is important to distinguish between the general term "RNN" and the specific implementation of an RNN using LSTM layers. While traditional RNNs are effective for processing sequences, they often struggle with learning long-range dependencies due to issues such as vanishing gradients. LSTM networks address this limitation by introducing memory cells that can maintain information over longer periods, making them more effective for tasks involving temporal sequences.

In our implementation, the architecture labeled as an "RNN" utilizes an LSTM layer to leverage these advanced capabilities. The model comprises an input layer with the shape defined by the number of features and a single time step, followed by an LSTM layer with 64 units configured to return sequences, a Flatten layer, a Dense layer with 64 units using the ReLU activation function, and a final Dense layer with softmax activation to produce the output probabilities for earthquake categories.

The distinction between the RNN and LSTM models in our study is as follows:

- The RNN model specifically refers to the architecture described above, where an LSTM layer is employed to capture long-term dependencies in the sequential data.
- The LSTM model mentioned separately in the manuscript refers to another architecture that also utilizes an LSTM layer but is designed with different configurations, such as not returning sequences after the LSTM layer, which results in variations in how the temporal information is processed. Both models were evaluated to understand their effectiveness in forecasting earthquake categories, with the LSTM-based RNN model demonstrating strong performance due to its ability to model complex temporal patterns in the dataset.

Gradient boosting machines (GBM)

We applied Gradient Boosting Machines (GBM) to predict earthquake categories, and the resulting accuracy is shown in Table 5. Gradient Boosting Machines (GBM) are an ensemble learning technique developed by Jerome Friedman. GBM consists of weak learners, typically regression trees, that are sequentially added using a functional gradient descent to minimize the loss function of the entire ensemble¹²⁶. In seismology, GBM has been utilized to enhance earthquake prediction models by optimizing the loss function and improving the accuracy of seismic forecasts.

Convolutional neural networks (CNN)

We utilized Convolutional Neural Networks (CNN) to predict earthquake categories, and the accuracy results are displayed in Table 5. In the field of seismology, Convolutional Neural Networks (CNN) have proven to be a valuable tool for earthquake prediction and analysis. CNNs, a type of deep neural network that incorporates convolution calculations and has a deep structure, are well-suited for handling seismic data and predicting seismic events¹²⁷. Researchers have successfully utilized CNNs in seismology to analyze seismic patterns, classify seismic events, and forecast earthquake occurrences with increased accuracy, contributing to more effective disaster management strategies and risk mitigation efforts in earthquake-prone regions.

A study demonstrated the development of a CNN model capable of detecting and classifying seismic body wave phases across various circumstances, highlighting the effectiveness of CNNs in seismic phase detection¹²⁸. Through the application of CNNs, researchers can automate the process of identifying seismic phases, leading to improved seismic event classification and analysis. This utilization of CNNs emphasizes their value in enhancing seismic data processing and interpretation in seismology.

CNNs have been integrated into earthquake prediction models to assess seismic vulnerability and forecast seismic ground motions. By incorporating CNNs into seismic vulnerability assessment frameworks, researchers can enhance the accuracy of seismic impact predictions on structures and infrastructure. This integration showcases the effectiveness of CNNs in analyzing seismic data and improving the prediction of earthquake impacts, thereby supporting disaster preparedness and risk mitigation strategies in seismology.

CNNs have been employed in the classification of seismic events based on waveform data, demonstrating their ability to process complex seismic signals and accurately classify seismic events¹²⁹.

Long short-term memory networks (LSTM)

We used Long Short-Term Memory Networks (LSTM) to predict earthquake categories, and the accuracy results are presented in Table 5.

Recurrent Long Short-Term Memory Networks (LSTM) are a type of neural network architecture particularly well-suited for sequential data analysis due to their ability to retain information over long periods. In the context of seismology, LSTM networks have been increasingly utilized for earthquake prediction. These networks excel in capturing the temporal dependencies present in seismic data, making them valuable tools for forecasting seismic events. Studies such as those by Hsu et al.¹³⁰, Cao et al.¹³¹, and Abri and Artuner¹³² have demonstrated the effectiveness of LSTM networks in predicting various seismic parameters like peak ground acceleration (PGA) and earthquake occurrences.

In seismology, the prediction of earthquakes has long been a challenging and critical endeavor due to its implications for public safety and disaster mitigation. Researchers have explored various approaches to improve earthquake prediction accuracy, with a focus on leveraging advanced technologies like deep learning and neural networks. The study by Dias and Papa¹¹⁸ highlights the application of neural networks, specifically multilayer perceptron models, for probabilistic earthquake forecasting, showcasing the potential of machine learning techniques in seismic event prediction.

The integration of attention mechanisms with LSTM networks, as demonstrated in the work by Banna et al.³⁴, has shown promising results in enhancing earthquake prediction accuracy. By incorporating attention mechanisms, which allow the model to focus on relevant parts of the input sequence, the LSTM network can better capture subtle patterns in seismic data, leading to improved forecasting capabilities.

k-nearest neighbors (k-NN)

We applied k-Nearest Neighbors (k-NN) to predict earthquake categories, and the accuracy results are shown in Table 5. In seismology, the k-Nearest Neighbors (k-NN) algorithm has been utilized as a valuable tool for earthquake prediction and analysis. The k-NN algorithm is a popular non-parametric method used for classification and regression, making it suitable for handling seismic data and predicting seismic events¹³³. By leveraging the k-NN algorithm, seismologists can analyze seismic patterns, classify seismic events, and forecast earthquake occurrences with enhanced accuracy, contributing to more effective disaster management strategies and risk mitigation efforts in earthquake-prone regions.

Researchers have employed the k-NN algorithm in seismology to classify seismic signals, differentiate between various types of seismic events, and predict the likelihood of earthquakes based on historical seismic data¹³⁴. By applying the k-NN algorithm, researchers can develop models that aid in decision-making during seismic events, improve earthquake emergency response strategies, and refine earthquake forecasting methodologies. This utilization of the k-NN algorithm demonstrates its effectiveness in analyzing seismic data and supporting decision-making processes in seismology¹³⁵.

Gated recurrent units (GRU)

We implemented Gated Recurrent Units (GRU) to predict earthquake categories, and the accuracy results are detailed in Table 5. Gated Recurrent Units (GRU) are a type of neural network architecture designed to efficiently model sequential data, similar to LSTM networks. In seismology, GRUs have become valuable for earthquake prediction due to their streamlined architecture with fewer parameters, making them computationally efficient for certain applications in earthquake forecasting. Studies by Dias and Papa¹¹⁸ and Wang et al.¹³⁶ have explored the use of neural networks, including GRUs, in earthquake prediction, demonstrating the potential of these models in capturing complex temporal patterns in seismic data.

Seismologists are increasingly utilizing advanced machine learning techniques, such as GRUs, to enhance the accuracy and reliability of earthquake prediction models. By employing GRU networks, researchers can analyze seismic data sequences effectively and extract meaningful patterns for more precise seismic event forecasts. Akter¹³⁷ utilized an Evidential Reasoning Approach to predict earthquakes based on specific signs and patterns, showcasing the versatility of neural network models like GRUs in seismic hazard assessment.

Integrating GRUs with additional data sources, such as GPS data and outgoing longwave radiation, has shown promise in improving earthquake prediction accuracy. While Gitis et al.⁴⁹ stress the importance of using artificial neural networks for earthquake prediction, studies like that of Zhai et al.⁵⁰ demonstrate the effectiveness of combining GRU models with time series forecasting techniques to detect thermal anomalies in earthquake processes, highlighting the interdisciplinary approach required in modern seismology research.

In earthquake forecasting, evaluating seismic parameters and their spatial variations is crucial for developing robust prediction models. Research by Hussain et al.⁴⁸ on the spatial variation of b-values and their relationship with fault blocks suggests the potential of using such parameters alongside GRU networks to predict high-magnitude earthquakes. Additionally, studies like that of Marc et al.¹¹⁵ focus on predicting the area affected by earthquake-induced landslides based on seismological parameters, illustrating the practical applications of integrating GRU models with geophysical data for hazard assessment.

Analyzing earthquake catalogs and historical seismicity patterns provides valuable insights for refining earthquake prediction models. Investigations such as those by Chouliaras¹³⁸ on the earthquake catalog of the National Observatory of Athens and Alabi et al.¹³⁹ on seismicity patterns in Southern Africa emphasize the importance of leveraging historical seismic data to enhance the performance of GRU-based forecasting models.

Support vector machines (SVM)

We used Support Vector Machines (SVM) to predict earthquake categories, and the accuracy results are shown in Table 5.

Support Vector Machines (SVM) have been utilized in seismology for earthquake prediction and analysis. Researchers have employed SVM as a machine learning tool to enhance earthquake forecasting models and improve the accuracy of seismic event predictions¹⁴⁰. By leveraging the capabilities of SVM, seismologists can analyze seismic data, classify seismic events, and predict earthquake occurrences with greater precision, contributing to more effective disaster management strategies and risk mitigation efforts in earthquake-prone regions.

In the context of seismology, SVM has been used to classify seismic signals, differentiate between various types of seismic events, and forecast the likelihood of earthquakes based on historical seismic data¹⁴⁰. By utilizing SVM algorithms, researchers can develop models that aid in decision-making during seismic events, enhance earthquake emergency response strategies, and refine earthquake forecasting methodologies. This application

of SVM highlights its effectiveness in analyzing seismic data and supporting decision-making processes in seismology.

SVM has also been combined with other machine learning algorithms to forecast earthquake occurrences, assess seismic vulnerability, and improve disaster planning and response strategies¹⁴¹.

Transformer models

We utilized Transformer Models to predict earthquake categories, and the accuracy results are presented in Table 5. Transformer models have become a valuable tool in various seismological applications, including earthquake prediction and seismic event analysis. In the context of seismology, transformer models have been utilized for tasks such as earthquake detection, phase picking, earthquake source characterization, and early warning systems¹⁴². These models have demonstrated their effectiveness in efficiently processing large volumes of seismic data and capturing complex temporal patterns present in seismic signals.

Seismologists have increasingly turned to machine learning techniques, including transformer models, to enhance earthquake prediction accuracy and improve seismic event forecasting. The ability of transformer models to handle sequential data and learn dependencies across different time steps makes them well-suited for analyzing seismic signals and extracting meaningful features for earthquake prediction. The application of transformer models in seismology has shown promising results in enhancing the understanding of seismic events and improving the reliability of earthquake forecasts.

Transformer models have been instrumental in separating earthquake signals from ambient noise in seismograms, contributing to more accurate earthquake detection and analysis¹⁴².

AdaBoost

We utilized AdaBoost with the SAMME algorithm to predict earthquake categories, and the accuracy results are displayed in Table 5. In the field of seismology, the AdaBoost machine learning algorithm has been employed to enhance earthquake prediction models and improve the accuracy of seismic event forecasts. AdaBoost, which stands for Adaptive Boosting, is a boosting algorithm that combines multiple weak learners to create a strong predictive model. Researchers have utilized AdaBoost in seismology to analyze seismic data, predict earthquake occurrences, and assess seismic vulnerability effectively¹⁴³.

A study introduced a novel earthquake prediction framework based on the classical AdaBoost machine learning algorithm, incorporating satellite remote sensing products like infrared and hyperspectral gases to detect earthquake perturbations¹⁴⁴. By integrating AdaBoost within the framework of inverse boosting pruning trees (IBPT), the researchers achieved promising forecasting results in the retrospective validation of global earthquake cases, demonstrating the algorithm's effectiveness in earthquake prediction¹⁴⁴.

AdaBoost has been integrated into earthquake prediction models to evaluate seismic vulnerability and forecast seismic ground motions. The seismic vulnerability of Reinforced Concrete (RC) structures under single and multiple seismic events was predicted using various machine learning algorithms, including the AdaBoost Regressor¹⁴⁵. This incorporation of AdaBoost into seismic vulnerability assessment models underscores its usefulness in analyzing seismic data and improving the prediction of earthquake impacts on structures.

AdaBoost has also been applied in earthquake prediction systems that merge earthquake indicators with genetic programming to enhance prediction accuracy. An earthquake prediction system utilizing AdaBoost alongside earthquake prediction indicators has led to improved results in earthquake forecasting¹⁴⁶.

Logistic regression

We applied Logistic Regression to predict earthquake categories, and the accuracy results are displayed in Table 5.

Logistic regression is a statistical method commonly used in various fields, including seismology, to analyze the relationship between a binary outcome and one or more predictor variables. In the context of seismology, logistic regression has been applied to predict and assess different aspects related to earthquakes. For instance, Jesse et al.¹⁴⁷ developed a global empirical model for assessing seismically induced landslides using logistic regression to understand the distribution of earthquake-triggered landslides based on factors like ground shaking, topographic slope, and land cover type. This study highlights the utility of logistic regression in modeling the impact of earthquakes on the occurrence of landslides.

Logistic regression has been utilized in earthquake prediction studies, although traditional models based on physical principles and statistical seismology laws have limitations in predicting large earthquakes¹⁴⁸. While logistic regression has been used in earthquake prediction models, it is essential to acknowledge the challenges in accurately forecasting significant seismic events solely based on empirical laws and physical principles.

In the specific context of seismically induced damage patterns, Rawat et al.¹⁴⁹ employed logistic regression to investigate seismic hazard assessment by considering site-specific parameters such as lithology, proximity to fault lines, soil texture, and groundwater. This application demonstrates how logistic regression can be used to understand the factors influencing seismic damage patterns and assess earthquake risks in different geological settings.

Logistic regression has also been applied in studies focusing on earthquake-induced landslides. Vilder et al.¹⁵⁰ used a logistic regression model to correlate earthquake-induced landslide inventories with various topographic, geological, and seismological parameters to determine the factors contributing to coseismic landslides.

Naive bayes

We used Naive Bayes to predict earthquake categories, and the accuracy results are shown in Table 5. One study by Fahandezhsadi and Sadi⁹³ focused on earthquake magnitude prediction using probabilistic classifiers, including Naive Bayes. The research aimed to enhance the accuracy of Naive Bayes by relaxing its strong

conditional independence assumption, indicating an interest in exploring the potential of Naive Bayes in seismic event forecasting. This study suggests that Naive Bayes, when adapted and optimized for seismic data, could potentially contribute to earthquake prediction efforts.

In a broader context of seismic event discrimination, a study by Elkhouly¹⁵¹ employed multiple machine learning techniques, including Naive Bayes, to distinguish between nuclear explosions and natural earthquakes. While the primary focus was on seismic discrimination, the inclusion of Naive Bayes in the machine learning models underscores its versatility and potential applicability in seismic data analysis. This research highlights the adaptability of Naive Bayes in complex seismic event classification tasks.

A study by Murwantara et al.¹⁵² comparing machine learning algorithms for earthquake prediction in Indonesia evaluated Naive Bayes alongside other methods like multinomial logistic regression and support vector machine. The research aimed to assess the performance of these algorithms in medium-to-long-term earthquake prediction using historical data, indicating the consideration of Naive Bayes as a potential tool for seismic forecasting. This study suggests that Naive Bayes can be part of a comprehensive approach to earthquake prediction when combined with other predictive models.

Model comparison

To evaluate the performance of different machine learning (ML) and neural network (NN) models, we calculated their accuracies on the test dataset. The results are presented in Table 5. All models except Naive Bayes achieved statistically significant accuracies. Notably, the Random Forest model achieved the highest accuracy, exceeding our initial expectations.

Baseline accuracy

The baseline accuracy in machine learning refers to the accuracy achieved by always predicting the most frequent class in the dataset. It serves as a reference point to determine if a model performs better than random guessing or a simple heuristic. In our dataset, the baseline accuracy was calculated as follows:

$$\text{Baseline Accuracy} = \frac{\text{Number of instances in the most frequent class}}{\text{Total number of instances}} \quad (77)$$

By dividing the number of instances in the most frequent class by the total number of instances, we obtained a baseline accuracy of:

$$\text{Baseline Accuracy} = 27.6999\% \quad (78)$$

This value represents the minimum accuracy any model should exceed to be considered better than a naive prediction strategy.

Confidence interval calculation

To assess whether the models' accuracies were statistically significant compared to the baseline accuracy, we calculated the 95% confidence interval (CI) for the baseline accuracy. The CI provides a range within which the true baseline accuracy is likely to fall with a 95% confidence level. The standard error (SE) of the baseline accuracy was calculated using the formula:

$$SE = \sqrt{\frac{\text{Baseline Accuracy} \times (1 - \text{Baseline Accuracy})}{n}} \quad (79)$$

where n represents the number of observations in the test dataset. The standard error quantifies the uncertainty in the baseline accuracy estimate due to the sample size. The 95% confidence interval was computed using the Z-score corresponding to a 95% confidence level:

$$Z = \text{CDF}^{-1}(0.975) \approx 1.96 \quad (80)$$

The confidence interval is then calculated as:

$$CI = \text{Baseline Accuracy} \pm Z \times SE \quad (81)$$

Substituting the values, we obtained:

$$CI \approx (26.4038\%, 28.9960\%) \quad (82)$$

This confidence interval indicates that any model accuracy significantly above 28.9960% would be considered statistically significant compared to the baseline accuracy at a 95% confidence level.

Critical accuracy

The critical accuracy is defined as the upper bound of the confidence interval, which in our case is 28.9960%. Any model with an accuracy greater than this threshold is deemed statistically significant in outperforming the

baseline. Fifteen out of sixteen ML and NN models demonstrated statistically significant results, as shown in Table 5.

Purpose of Z-score and SE

The Z-score, a statistical measure, quantifies the number of standard deviations a data point is from the mean of a distribution. For our 95% confidence interval, a Z-score of approximately 1.96 is used to cover the central 95% of the normal distribution.

The standard error (SE) represents the standard deviation of the sampling distribution of a statistic, in this case, the baseline accuracy. The SE helps to quantify the uncertainty around the estimated baseline accuracy, enabling us to calculate the confidence interval. The lower the SE, the more precise our estimate of the baseline accuracy.

By understanding the Z-score and SE, we can more accurately interpret the confidence interval and determine which models demonstrate statistically significant performance improvements over the baseline.

Evaluation of model performance using multiple metrics

To comprehensively evaluate the performance of each machine learning and neural network model, we used multiple metrics: Accuracy, Precision, Recall, F1-Score, and ROC-AUC. The results of these metrics are summarized in Table 6.

Accuracy

Accuracy measures the proportion of correctly predicted instances among the total instances. It is calculated as:

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total Instances}} \quad (83)$$

Models like Random Forest, XGBoost, and LightGBM achieved the highest accuracy scores, reflecting their strong performance in correctly classifying the earthquake data.

Precision

Precision represents the proportion of true positive predictions among all positive predictions. It is given by:

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \quad (84)$$

High precision values for models like Random Forest, XGBoost, and LightGBM indicate that these models have a low rate of false positives, making them reliable for predicting specific earthquake classes.

Recall

Recall measures the proportion of true positive predictions among all actual positives and is calculated as:

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \quad (85)$$

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
Logistic Regression	0.3264	0.3448	0.3264	0.3001	0.6255
Decision Trees	0.8967	0.8969	0.8967	0.8967	0.9347
Random Forest	0.9769	0.9769	0.9769	0.9768	0.9989
Gradient Boosting Machines	0.7009	0.7166	0.7009	0.7013	0.9196
Support Vector Machines	0.5207	0.5835	0.5207	0.5071	0.8118
k-Nearest Neighbors	0.6384	0.6483	0.6384	0.6372	0.8697
Naive Bayes	0.2620	0.2954	0.2620	0.2420	0.5742
AdaBoost	0.3533	0.4418	0.3533	0.3229	0.6481
XGBoost	0.9675	0.9675	0.9675	0.9674	0.9975
LightGBM	0.9336	0.9341	0.9336	0.9336	0.9936
Multilayer Perceptron	0.9260	0.9263	0.9260	0.9260	0.9901
Convolutional Neural Networks	0.6262	0.6561	0.6262	0.6454	0.8959
Recurrent Neural Networks	0.7681	0.7651	0.7681	0.7619	0.9485
Long Short-Term Memory Networks	0.4528	0.4782	0.4528	0.4437	0.7528
Gated Recurrent Units	0.5360	0.5444	0.5360	0.5336	0.8279
Transformer Models	0.4596	0.4892	0.4596	0.4525	0.7618

Table 6. Comparison of model performance using multiple metrics.

Models such as Random Forest, XGBoost, and LightGBM exhibit high recall scores, demonstrating their effectiveness in correctly identifying most of the actual earthquake events.

F1-Score

The F1-Score is the harmonic mean of Precision and Recall, balancing both metrics. It is computed as:

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (86)$$

A high F1-Score, particularly for the Random Forest and XGBoost models, indicates a balanced performance in terms of both correctly predicting earthquake events and minimizing false positives.

ROC-AUC

The ROC-AUC (Receiver Operating Characteristic - Area Under Curve) measures the ability of a model to distinguish between classes. A higher ROC-AUC value indicates better model performance in separating the classes. It is particularly useful for evaluating models where class imbalance is present.

$$\text{ROC-AUC} = \text{Area under the ROC curve} \quad (87)$$

Models like Random Forest, XGBoost, and LightGBM have ROC-AUC values close to 1, suggesting excellent discriminative ability.

Conclusion

The comparative analysis using multiple metrics shows that Random Forest, XGBoost, and LightGBM models consistently outperform other models across all evaluation metrics. These models demonstrate high accuracy, precision, recall, F1-Score, and ROC-AUC values, making them the most reliable for earthquake prediction in this study. Conversely, models such as Naive Bayes, AdaBoost, and Logistic Regression exhibit lower performance across all metrics, indicating limited effectiveness for this specific task.

Analysis of model accuracies in predicting earthquake classes

In this section, we analyze the accuracy of each machine learning (ML) and neural network (NN) model in predicting different earthquake classes, particularly focusing on their performance in correctly predicting class 6, which represents strong earthquakes.

Logistic regression

Logistic Regression demonstrated relatively low accuracy across all classes, with accuracies ranging from 0.000 for class 6 to 0.505 for class 4. Its accuracy in predicting class 6 (strong earthquakes) was 0.000, indicating limited capability in correctly identifying more severe earthquakes. The accuracies for each class are as follows:

Class 1: 0.294, Class 2: 0.318, Class 3: 0.292, Class 4: 0.505, Class 5: 0.465, Class 6: 0.000.

Decision trees

The Decision Tree classifier showed a significant improvement over Logistic Regression, with accuracies exceeding 0.87 across most classes. It achieved an accuracy of 0.886 in predicting class 5 and 0.906 in predicting class 4. However, the model's accuracy in predicting class 6 was 0.868, demonstrating a reasonably good performance in recognizing strong earthquakes. The accuracies for each class are:

Class 1: 0.905, Class 2: 0.904, Class 3: 0.883, Class 4: 0.906, Class 5: 0.886, Class 6: 0.868.

Random forest

The Random Forest classifier emerged as the top-performing model, achieving consistently high accuracy across all classes, with values above 0.96. Notably, it achieved an accuracy of 0.979 for class 5 and 0.976 for class 4. For class 6, it maintained an accuracy of 0.982, indicating a strong ability to correctly predict severe earthquakes. The accuracies are as follows: **Class 1: 0.978, Class 2: 0.976, Class 3: 0.976, Class 4: 0.976, Class 5: 0.979, Class 6: 0.982.**

Gradient boosting machines (GBM)

Gradient Boosting Machines showed good performance, particularly for class 4 (accuracy of 0.896) and class 6 (accuracy of 0.913). However, its accuracy was lower for class 5 (0.707), suggesting some limitations in handling moderately severe earthquakes compared to Random Forest. The detailed accuracies are: **Class 1: 0.717, Class 2: 0.631, Class 3: 0.683, Class 4: 0.896, Class 5: 0.707, Class 6: 0.913.**

Support vector machines (SVM)

Support Vector Machines demonstrated moderate accuracy across all classes, with a peak accuracy of 0.861 for class 4. However, its performance in predicting class 6 was lower, with an accuracy of 0.815, indicating a moderate ability to correctly classify strong earthquakes. The accuracies for each class are: **Class 1: 0.473, Class 2: 0.447, Class 3: 0.565, Class 4: 0.861, Class 5: 0.735, Class 6: 0.815.**

k-nearest neighbors (k-NN)

The k-Nearest Neighbors model achieved varied accuracy across classes, with its highest accuracy being 0.767 for class 4. However, its performance in predicting class 6 was relatively poor, with an accuracy of 0.746, suggesting limitations in its ability to handle severe earthquakes. The accuracies are: **Class 1: 0.546, Class 2: 0.653, Class 3: 0.641, Class 4: 0.767, Class 5: 0.656, Class 6: 0.746.**

Naive Bayes

The Naive Bayes classifier showed low accuracy across most classes, particularly for classes 1, 3, and 5, where the accuracy was below 0.25. For class 6, the model achieved an accuracy of 0.047, indicating a limited capability in predicting strong earthquakes. The accuracies for each class are: **Class 1: 0.250, Class 2: 0.396, Class 3: 0.222, Class 4: 0.418, Class 5: 0.163, Class 6: 0.047.**

AdaBoost

AdaBoost displayed good performance in predicting class 5, with perfect accuracy (1.000). However, its accuracy for class 6 was only 0.000, showing that while AdaBoost can handle some classes well, its performance in predicting strong earthquakes is less reliable. The accuracies are: **Class 1: 0.278, Class 2: 0.348, Class 3: 0.359, Class 4: 0.606, Class 5: 1.000, Class 6: 0.000.**

XGBoost

XGBoost performed exceptionally well, achieving high accuracies across all classes, with values above 0.96. For class 6, it achieved an accuracy of 0.982, indicating its strong capability to predict strong earthquakes. The accuracies for each class are: **Class 1: 0.966, Class 2: 0.969, Class 3: 0.962, Class 4: 0.977, Class 5: 0.960, Class 6: 0.982.**

LightGBM

LightGBM also demonstrated robust performance, with accuracies generally exceeding 0.91 across all classes. The accuracy for class 6 was 0.972, showing it is effective in identifying strong earthquakes, though slightly less so than XGBoost and Random Forest. The accuracies are: **Class 1: 0.948, Class 2: 0.917, Class 3: 0.932, Class 4: 0.962, Class 5: 0.909, Class 6: 0.972.**

Multilayer perceptron (MLP)

The Multilayer Perceptron exhibited good performance across most classes, with an accuracy of 0.943 for class 4. For class 6, it achieved an accuracy of 0.940, indicating strong performance in predicting strong earthquakes. The accuracies are: **Class 1: 0.928, Class 2: 0.937, Class 3: 0.914, Class 4: 0.943, Class 5: 0.894, Class 6: 0.940.**

Convolutional neural networks (CNN)

The Convolutional Neural Network model showed moderate accuracy across all classes, with a peak of 0.723 for class 6. However, its accuracy for class 6 was 0.723, indicating moderate capability in predicting strong earthquakes. The accuracies are: **Class 1: 0.612, Class 2: 0.671, Class 3: 0.605, Class 4: 0.681, Class 5: 0.600, Class 6: 0.723.**

Recurrent neural networks (RNN)

Recurrent Neural Networks achieved higher accuracy than CNN, with an accuracy of 0.923 for class 6 and 0.816 for class 4. For class 6, the RNN model showed strong accuracy, suggesting good capability to predict strong earthquakes. The accuracies are: **Class 1: 0.729, Class 2: 0.736, Class 3: 0.781, Class 4: 0.816, Class 5: 0.716, Class 6: 0.923.**

Long short-term memory networks (LSTM)

The Long Short-Term Memory Networks model showed varying performance across the classes, with its highest accuracy of 0.910 for class 4. Its accuracy for class 6 was 0.733, indicating moderate capability in predicting strong earthquakes. The accuracies are: **Class 1: 0.733, Class 2: 0.563, Class 3: 0.909, Class 4: 0.910, Class 5: 0.733, Class 6: 0.733.**

Gated recurrent units (GRU)

The Gated Recurrent Units model demonstrated moderate accuracy across classes, with an accuracy of 0.695 for class 4 and 0.582 for class 6, suggesting a moderate capability in predicting strong earthquakes. The accuracies are: **Class 1: 0.553, Class 2: 0.477, Class 3: 0.547, Class 4: 0.695, Class 5: 0.557, Class 6: 0.582.**

Transformer models

The Transformer model showed a moderate performance overall, with accuracies peaking at 0.588 for class 4. For class 6, its accuracy was 0.500, indicating less effectiveness in predicting strong earthquakes. The accuracies are: **Class 1: 0.394, Class 2: 0.444, Class 3: 0.437, Class 4: 0.588, Class 5: 0.488, Class 6: 0.500.**

Conclusion

Overall, the analysis shows that Random Forest, XGBoost, and LightGBM models demonstrated the highest accuracies in predicting class 6 (strong earthquakes), with Random Forest achieving the best performance at 0.982. Models such as Naive Bayes, CNN, and Transformer exhibited limited capability in correctly identifying strong earthquakes. The superior performance of Random Forest and XGBoost highlights the effectiveness of ensemble learning techniques in handling complex, multiclass earthquake prediction tasks. Meanwhile, some

neural network architectures, such as MLP and RNN, also performed reasonably well, but their performance varied more across different classes. This underscores the importance of selecting appropriate models and hyperparameters for specific predictive tasks in earthquake forecasting.

Comparison of confusion matrices for the best-performing models

This subsection compares the confusion matrices of the three best-performing models: XGBoost, LightGBM, and Random Forest. These models were selected based on their overall high accuracy in predicting earthquake classes, particularly their strong performance in identifying class 6 (strong earthquakes).

XGBoost

The confusion matrix for XGBoost shows a robust performance across all classes. The model correctly predicted the majority of samples for each class, with high diagonal values indicating accurate predictions. For class 6, which represents strong earthquakes, XGBoost correctly predicted 111 out of 115 samples, resulting in very high accuracy for this class. The misclassifications for class 6 were minimal, with only a few samples being incorrectly classified into other classes, mostly into class 5. Overall, XGBoost demonstrated strong predictive capability with minimal confusion between classes, particularly for severe earthquakes.

LightGBM

The confusion matrix for LightGBM also demonstrates strong performance, though it exhibits slightly more misclassifications compared to XGBoost. LightGBM correctly identified 104 out of 115 samples for class 6. The model shows a tendency to confuse class 6 with neighboring classes (e.g., class 5 and class 4), indicating that while it is highly effective, there is a slight reduction in accuracy compared to XGBoost, particularly in distinguishing between the most severe earthquakes. The model's performance is generally consistent, but it shows slightly more misclassifications in higher classes.

Random forest

Random Forest exhibited the highest accuracy among all models in predicting class 6. The confusion matrix shows that Random Forest correctly predicted 112 out of 115 samples for class 6, with only 3 samples being misclassified. The model demonstrated minimal confusion between different classes, particularly for the higher classes (class 5 and 6), which suggests that Random Forest is highly adept at distinguishing between varying earthquake magnitudes. The misclassifications were fewer compared to both LightGBM and XGBoost, reinforcing its status as the best model in this study.

Overall comparison

- Accuracy for class 6:** All three models performed well in predicting class 6, but Random Forest slightly outperformed the others by correctly classifying more samples. XGBoost also demonstrated high accuracy but showed minor confusion with class 5. LightGBM, while still effective, had slightly more difficulty distinguishing class 6 from other classes.
- Misclassification patterns:** XGBoost and LightGBM showed more misclassification between adjacent classes, such as classes 5 and 6. In contrast, Random Forest exhibited fewer misclassifications overall, particularly for higher classes, demonstrating a better capability to handle complex multiclass classification tasks.
- Conclusion:** The confusion matrix analysis confirms that while all three models—XGBoost, LightGBM, and Random Forest—are highly effective in predicting earthquake classes, Random Forest marginally outperforms the others in terms of correctly identifying strong earthquakes (class 6). The performance of these models highlights the effectiveness of ensemble learning techniques in handling complex, multiclass problems like earthquake prediction. Figures 19, 20, and 21 provide a visual representation of the confusion matrices for XGBoost, LightGBM, and Random Forest, respectively.

Best performer: random forest

Given that the Random Forest machine learning algorithm achieved the highest accuracy and successfully captured complex patterns in the earthquake data, it outperformed other models in terms of predictive accuracy. Consequently, we will focus our further analysis on Random Forest¹⁵³.

Obtaining the highest accuracy subset using the information gain method for random forest

To determine the most important features for predicting earthquake classes, we employed the Information Gain (IG) method to calculate the contribution of each feature towards reducing uncertainty or entropy in the dataset. The calculated IG values for each feature are presented in Table 7, ranked from highest to lowest importance. The top 15 features with the highest IG values were selected as the optimal subset for model training and evaluation.

The Information Gain (IG) values were calculated to identify features that most effectively reduce entropy, which is a measure of uncertainty or randomness in the dataset. Details of the entropy and Information Gain calculations are provided in the Appendix (see Appendix A.3). These calculations illustrate the process of selecting the features that provide the most valuable information for distinguishing between earthquake classes^{154,155}.

An iterative evaluation process was conducted by training a Random Forest model using subsets of features ranked by their IG values, starting with the most important feature and incrementally adding one feature at a time. For each subset, the model's accuracy was evaluated on the test dataset. The results of this evaluation are summarized in Table 8.

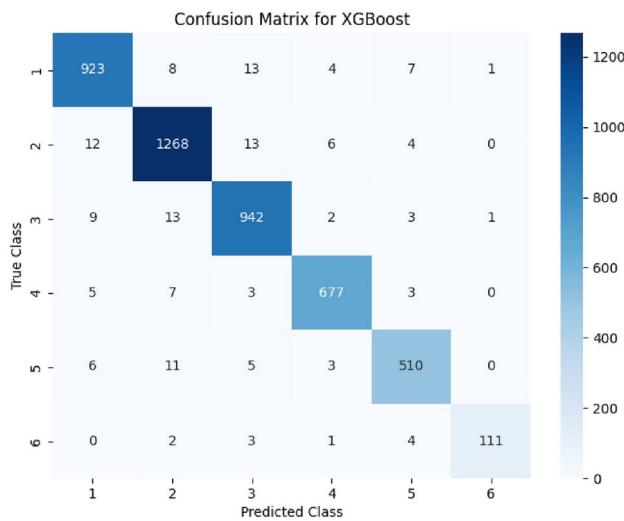


Fig. 19. Confusion matrix for the XGBoost model using all variables.

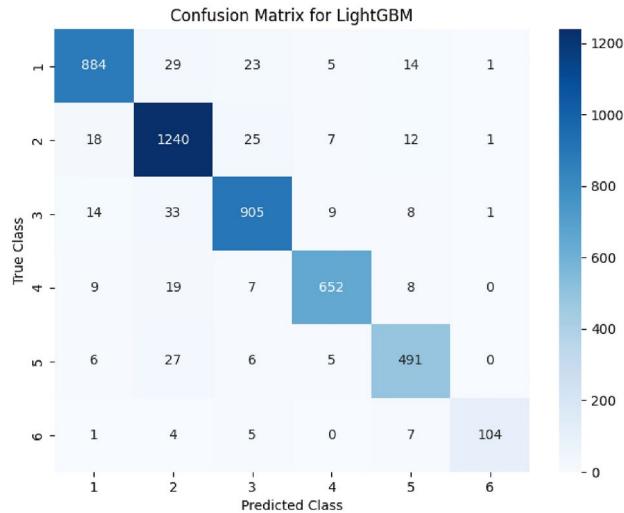


Fig. 20. Confusion matrix for the LightGBM model using all variables.

The analysis revealed that the highest accuracy of **0.9797** was achieved with a 15-variable subset. This subset, which included the top 15 features ranked by their IG values, demonstrated the optimal balance between model complexity and performance. The selected subset is detailed in Table 9.

In summary, the 15-variable subset achieved the highest predictive accuracy, demonstrating that these features collectively provide the most valuable information for earthquake prediction. The selected subset captures critical spatial, temporal, and seismic characteristics, providing a comprehensive understanding of the factors influencing earthquake behavior (see Table 9).

This subset includes variables that capture a range of important earthquake-related characteristics, such as the number of recent earthquakes, depth patterns, clustering, variation in inter-event times, magnitude changes, and geographical information. These features were chosen based on their ability to contribute significantly to reducing uncertainty in the model, thereby improving the model's predictive performance. The detailed calculations for entropy and information gain, which underpin this feature selection, are presented in the Appendix (see Appendix A.3)^{154,155}.

Feature selection and evaluation on training and test datasets

To address the potential concern of overfitting, we employed a comprehensive feature selection process and evaluated the model performance separately on both the training and test datasets.

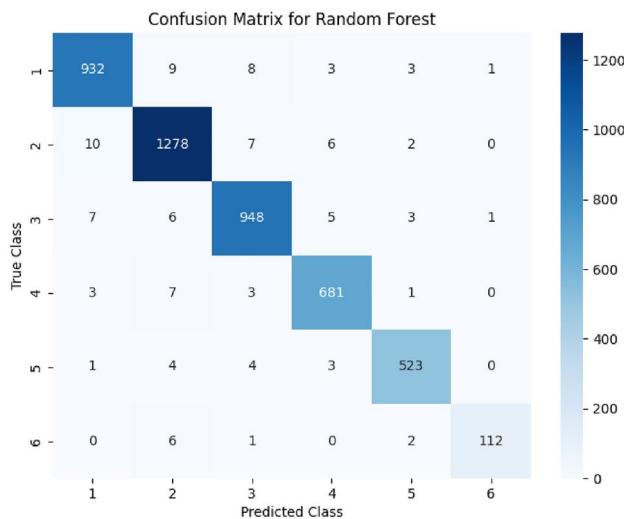


Fig. 21. Confusion matrix for the random forest model using all variables.

No.	Variable	Explanation	Importance
1	$N_{\text{eq},30}$	Number of earthquakes in the last 30 days	0.110651
2	\bar{d}_i	Rolling mean depth of earthquakes in the last 30 days	0.101531
3	C_{clust}	Clustering coefficient of earthquakes in the last 30 days	0.101056
4	$dE12$	A feature representing the change in energy release patterns	0.100016
5	$\sigma_{\text{mag},30}$	Standard deviation of magnitude for the last 30 days	0.098347
6	T	Elapsed time since the last earthquake	0.086100
7	CV	Coefficient of variation of the inter-event times	0.059963
8	b -value	Gutenberg-Richter b -value for the moment magnitude	0.059895
9	ΔM	Difference between the largest observed and expected magnitudes	0.057398
10	η	Sum of the mean square deviation from the regression line	0.054387
11	$M_{\text{max}}^{\text{last week}}$	Maximum magnitude recorded during the last week	0.028624
12	$\Delta b_{i-8,i-10}$	Incremental b -value between events $i - 8$ and $i - 10$	0.026027
13	Longitude	Longitude of the earthquake epicenter	0.021599
14	$\Delta b_{i-6,i-8}$	Incremental b -value between events $i - 6$ and $i - 8$	0.021284
15	Latitude	Latitude of the earthquake epicenter	0.017837
16	$\Delta b_{i-4,i-6}$	Incremental b -value between events $i - 4$ and $i - 6$	0.017439
17	$\Delta b_{i-2,i-4}$	Incremental b -value between events $i - 2$ and $i - 4$	0.014963
18	$\Delta b_{i,i-2}$	Incremental b -value between events i and $i - 2$	0.011481
19	Magnitude	Magnitude of the earthquake	0.011401

Table 7. Feature importance for earthquake prediction models (ordered high to low).

Feature selection process

The feature selection process involved multiple steps to ensure that only the most relevant predictors were included in the final model, minimizing the risk of overfitting:

- Multicollinearity analysis and mitigation:** Initially, a multicollinearity analysis was conducted using the Variance Inflation Factor (VIF) to identify and remove highly collinear variables. This step was necessary to reduce redundancy in the feature set and enhance the stability of the model. Details of the analysis and the subsequent variable reduction are provided in the section titled “Multicollinearity Analysis and Mitigation.”
- Iterative subset evaluation:** After addressing multicollinearity, an iterative subset evaluation process was performed to determine the optimal combination of features that maximized model performance. The evaluation was conducted by iteratively assessing model accuracy on the test dataset with increasing subsets of the top-ranked features based on their importance scores derived from a Random Forest model trained on the training dataset. This process identified the best-performing subset of 15 variables, as presented in Table 9. By combining multicollinearity mitigation with iterative subset evaluation, the feature selection ensured

Features	Accuracy
1 Variable Subset	0.3439
2 Variable Subset	0.4948
3 Variable Subset	0.7648
4 Variable Subset	0.9489
5 Variable Subset	0.9692
6 Variable Subset	0.9745
7 Variable Subset	0.9773
8 Variable Subset	0.9788
9 Variable Subset	0.9771
10 Variable Subset	0.9766
11 Variable Subset	0.9777
12 Variable Subset	0.9782
13 Variable Subset	0.9786
14 Variable Subset	0.9784
15 Variable Subset	0.9797
16 Variable Subset	0.9771
17 Variable Subset	0.9758
18 Variable Subset	0.9764
19 Variable Subset	0.9760

Table 8. Accuracy results for incremental feature subsets with the IG method for random forest.

No.	Variable	Explanation
1	$N_{\text{eq},30}$	Number of Earthquakes in the Last 30 Days
2	\bar{d}_i	Rolling Mean of Depth
3	C_{clust}	Clustering Coefficient
4	$dE12$	A feature representing the change in energy release patterns
5	$\sigma_{\text{mag},30}$	Standard Deviation of Magnitude
6	T	Elapsed time between the last n events
7	CV	Coefficient of variation of the inter-event times
8	b -value	Gutenberg-Richter b -value for the moment magnitude
9	ΔM	Difference between the largest observed and expected magnitudes
10	η	Sum of the mean square deviation from the regression line
11	$M_{\text{max}}^{\text{last week}}$	Maximum magnitude recorded during the last week
12	$\Delta b_{i-8,i-10}$	Incremental b -value between events $i - 8$ and $i - 10$
13	Longitude	Longitude of the earthquake epicenter
14	$\Delta b_{i-6,i-8}$	Incremental b -value between events $i - 6, i - 8$
15	Latitude	Latitude of the earthquake epicenter

Table 9. Subset with 0.9797% accuracy on the test dataset.

that the included predictors contributed unique and significant information to the models, thereby reducing the likelihood of overfitting.

Evaluation on training and test datasets

To comprehensively evaluate the potential for overfitting, the model's performance was assessed separately on both the training and test datasets. Presenting results for both datasets allows for the determination of whether the model's performance is consistent across different data sets, providing a more robust evaluation of model generalization.

The results showed that the model achieved a **Training Accuracy** of 1.0000, indicating perfect performance on the training data. However, this can often be a sign of overfitting if the test accuracy is significantly lower. In our case, the **Test Accuracy** was 0.9797, which is close to the training accuracy, suggesting minimal overfitting and good generalization capability.

Furthermore, additional metrics were calculated to provide a more detailed evaluation:

- **Precision:** 0.9797

- **Recall:** 0.9797
- **F1-Score:** 0.9797
- **ROC-AUC:** 0.9989 These metrics confirm that the model not only maintained high accuracy but also demonstrated balanced performance in terms of correctly predicting earthquake events (high Precision and Recall) while maintaining excellent discriminative ability (high ROC-AUC).

Conclusion

By incorporating a detailed feature selection process and presenting results separately for both the training and test datasets, the risk of overfitting was minimized, and the robustness of the model's performance was verified. The consistency between the training and test accuracies, along with high scores across multiple evaluation metrics, supports the conclusion that the model generalizes effectively to new, unseen data.

Conclusion

In this study, we conducted an extensive analysis of various references pertaining to earthquake prediction. Our research focused on the development of a predictive pattern matrix, leveraging machine learning algorithms and neural networks to forecast earthquakes. Through feature engineering of 21 diverse predictive features from historical earthquake records, we achieved an impressive accuracy of 97.97% with 15 features for the Los Angeles, California region, as seen in Table 9. This finding highlights the significance of integrating advanced computational techniques with rigorous data analysis, pointing towards a promising future for earthquake forecasting research and applications. Notably, our method demonstrated the capability to accurately predict the category of earthquakes across six distinct categories within a 30-day period. Achieving such a high level of accuracy is critical for enhancing disaster preparedness and response strategies in Los Angeles, a region susceptible to seismic activity. Our approach offers a comprehensive and precise method for earthquake prediction, providing valuable insights to the field of seismology.

Analysis of fine-tuned random forest model

In this study, we employed a Random Forest model to predict the maximum earthquake class in the coming 30 days, focusing specifically on a fine-tuned approach using a subset of 15 selected features. The primary goal was to optimize the model's predictive performance by selecting the most relevant features and fine-tuning the hyperparameters to achieve the highest possible accuracy. The selected 15 features are listed in Table 9.

Data splitting: training, validation, and test sets

To evaluate the performance of the Random Forest model, the dataset was divided into three distinct subsets: the *training set*, the *validation set*, and the *test set*.

- **Training set:** This subset, comprising 60% of the total data, was used to train the model. The training set allows the model to learn the underlying patterns and relationships within the data by adjusting its internal parameters accordingly.
- **Validation set:** The validation set, accounting for 20% of the data, was used during the hyperparameter tuning phase to evaluate different configurations of the model. This set provides an unbiased evaluation of the model's performance while fine-tuning its hyperparameters, helping to prevent overfitting and ensuring that the model generalizes well to new, unseen data.
- **Test set:** The final 20% of the data was reserved for the test set, which was used to evaluate the model's performance after the training and tuning phases were completed. The test set serves as an independent check of the model's ability to make accurate predictions on data it has not encountered before, providing a realistic assessment of its generalization capabilities. By splitting the data into these three sets, we aimed to ensure a robust and reliable evaluation of the model, balancing the need for both learning and validation.

Hyperparameter optimization process

To enhance the model's accuracy, we performed a grid search for hyperparameter tuning using the 15-variable subset. The hyperparameters optimized in the grid search included:

- **n_estimators:** The number of trees in the forest. We evaluated values of 100, 200, and 500.
- **max_depth:** The maximum depth of the tree. We considered values of None (allowing nodes to expand until all leaves are pure or until they contain less than the minimum samples required to split) and 10, 20, and 30.
- **min_samples_split:** The minimum number of samples required to split an internal node, tested with values of 2, 5, and 10.
- **min_samples_leaf:** The minimum number of samples required to be at a leaf node, with possible values of 1, 2, and 4.
- **bootstrap:** A boolean parameter indicating whether bootstrap samples are used when building trees, tested with values True and False. The grid search was performed using 3-fold cross-validation on the training set. This involved dividing the training set into three equal parts, or "folds." For each combination of hyperparameters, the model was trained on two of the folds and validated on the remaining fold, repeating this process three times so that each fold served as the validation set once. This resulted in a total of 540 fits (3 folds × 60 combinations of hyperparameters), allowing us to evaluate the different combinations comprehensively and identify the configuration that maximized validation accuracy. The choice of 3 folds strikes a balance between computational efficiency and robust model evaluation, ensuring sufficient variance without excessively increasing computation time.

Results of fine-tuning

The grid search process identified the best hyperparameters for the Random Forest model as follows:

- bootstrap: False
 - max_depth: 30
 - min_samples_leaf: 1
 - min_samples_split: 2
 - n_estimators: 200
- This configuration indicates that the model performed best without bootstrapping, with a maximum depth of 30 for the trees, and the minimum number of samples required for a split and at the leaf nodes set to their lowest possible values. Furthermore, the model used 200 trees to enhance its ensemble learning capability.

Interpretation of results

The fine-tuned Random Forest model achieved a validation accuracy of 0.9777, indicating that approximately 97.77% of the predictions on the validation set were correct. This high accuracy reflects the model's effectiveness in identifying the patterns and relationships within the earthquake data when trained on the selected 15 features.

When evaluated on the test set, the fine-tuned model achieved a test accuracy of 0.9808, suggesting that the model generalizes well to unseen data. The slight improvement in test accuracy compared to validation accuracy indicates that the model has successfully captured the underlying structure of the data without overfitting.

Conclusion

The results demonstrate that the selected 15-variable subset (Table 9) is highly informative for predicting the maximum earthquake class in the coming 30 days, and that the Random Forest model, when fine-tuned with the appropriate hyperparameters, provides robust and accurate predictions. The identified hyperparameters allow the model to leverage the full complexity of the data, thereby maximizing its predictive performance. This fine-tuned model can be used for further analyses and could serve as a reliable tool in earthquake forecasting.

Analysis of the confusion matrix for the 15-variable subset

The confusion matrix for the 15-variable subset, as shown in Figure 22, is based on the model's performance on the test data. This matrix provides a detailed breakdown of the model's classification results across different earthquake classes, helping to evaluate its ability to generalize to unseen data.

- **Class 1:** The model correctly classified 931 instances of Class 1, with 8 misclassified as Class 2, 12 as Class 3, 2 as Class 4, 2 as Class 5, and 1 as Class 6. This indicates that the model performs well in predicting Class 1, with relatively few misclassifications.
- **Class 2:** The model achieved high accuracy for Class 2, correctly identifying 1,284 instances. However, there were 11 instances misclassified as Class 1, 5 as Class 3, 2 as Class 4, 1 as Class 5, and none as Class 6. This shows a strong capability in identifying Class 2 but suggests some overlap between Class 1 and Class 2.
- **Class 3:** For Class 3, the model correctly classified 952 instances. However, it misclassified 4 instances as Class 1, 6 as Class 2, 4 as Class 4, 3 as Class 5, and 1 as Class 6. The confusion with neighboring classes indicates the model's slight difficulty in distinguishing between Classes 2, 3, and 4.
- **Class 4:** The model correctly identified 683 instances of Class 4. There were some misclassifications, with 3 instances predicted as Class 1, 7 as Class 2, and none in other classes. The model's performance for Class 4 is slightly less accurate, particularly with distinguishing it from Classes 2 and 3.

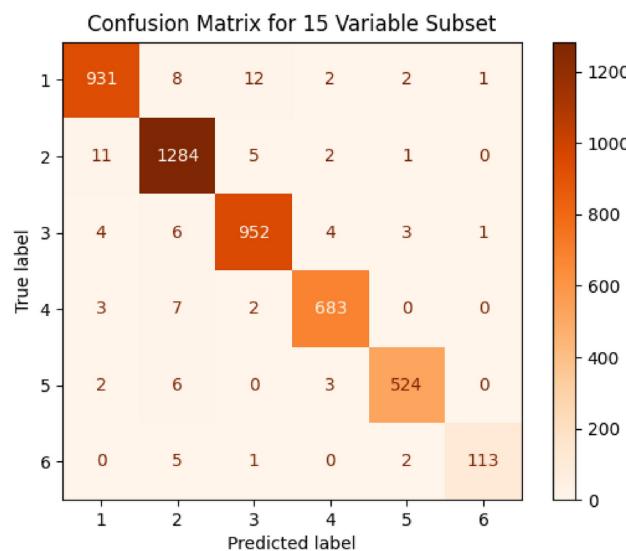


Fig. 22. Confusion matrix for 15-variable subset based on test data.

- **Class 5:** For Class 5, 524 instances were correctly classified. Misclassifications included 2 instances as Class 1, 6 as Class 2, and 3 as Class 4, with no confusion with other classes. The model shows good performance in predicting Class 5 but with some errors in distinguishing it from Classes 2 and 4.
- **Class 6:** The model had 113 correct predictions for Class 6, with 5 instances misclassified as Class 2, 1 as Class 3, 0 as Class 4, 2 as Class 5, and no misclassification into Class 1. This indicates that while the model is generally effective in predicting Class 6, there is some confusion with other classes, particularly with Class 2. Overall, the confusion matrix illustrates that the model maintains high accuracy across most classes, especially for Classes 1, 2, and 3. However, the confusion between Classes 2 and 6 and between Classes 2, 3, and 4 suggests that additional refinement may be necessary to enhance the model's differentiation capabilities between these classes.

Data availability

The dataset generated and analyzed during this study, including the engineered features used for the machine learning models, is publicly available on Zenodo at the following link: <https://zenodo.org/doi/10.5281/zenodo.13738726>. This dataset provides all the necessary information to reproduce the findings of this study.

Appendix

Example calculation for local clustering coefficient

Consider an earthquake event i with three neighbors: j , k , and l .

Neighbors:

- j , k , and l are all neighbors of i .
 - This means that $d_{ij} < \epsilon$, $d_{ik} < \epsilon$, and $d_{il} < \epsilon$.

Connections (edges) between neighbors:

- j and k are connected: There is a direct edge between j and k .
- k and l are connected: There is a direct edge between k and l .
- j and l are also connected: There is a direct edge between j and l . In this example:
- The neighbors of i form a complete subgraph because all possible edges between the neighbors j , k , and l exist.
- There are three edges: (j, k) , (k, l) , and (j, l) .

Calculation of E_i and $C_{\text{clust},i}$

- $k_i = 3$ (number of neighbors of i).
- $E_i = 3$ (number of edges between the neighbors). The Local Clustering Coefficient is calculated as:

$$C_{\text{clust},i} = \frac{2E_i}{k_i(k_i - 1)} \quad (88)$$

Substituting the values:

$$C_{\text{clust},i} = \frac{2 \times 3}{3 \times (3 - 1)} = \frac{6}{6} = 1 \quad (89)$$

This result $C_{\text{clust},i} = 1$ indicates that the neighbors of i form a complete subgraph, meaning all neighbors are directly connected to each other.

Entropy and information gain calculations

Entropy is a measure of the impurity or randomness in the data. For a dataset D with classes C_1, C_2, \dots, C_k , the entropy $H(D)$ is defined as:

$$H(D) = - \sum_{i=1}^k p(C_i) \log_2 p(C_i) \quad (90)$$

where $p(C_i)$ is the proportion of examples in D that belong to class C_i .

Example 1: binary classification with balanced classes

Suppose we have a dataset D with 10 examples, equally divided into two classes: C_1 and C_2 . The class distribution is as follows:

- $p(C_1) = \frac{5}{10} = 0.5$
- $p(C_2) = \frac{5}{10} = 0.5$

The entropy $H(D)$ is calculated as:

$$H(D) = -(p(C_1) \log_2 p(C_1) + p(C_2) \log_2 p(C_2)) \quad (91)$$

$$= -(0.5 \log_2 0.5 + 0.5 \log_2 0.5) \quad (92)$$

$$= -(0.5 \times -1 + 0.5 \times -1) \quad (93)$$

$$= -(-0.5 - 0.5) \quad (94)$$

$$= 1 \quad (95)$$

Example 2: binary classification with imbalanced classes

Now, consider another dataset D with 10 examples, but this time 7 examples belong to class C_1 and 3 examples belong to class C_2 . The class distribution is as follows:

- $p(C_1) = \frac{7}{10} = 0.7$
- $p(C_2) = \frac{3}{10} = 0.3$

The entropy $H(D)$ is calculated as:

$$H(D) = -(p(C_1) \log_2 p(C_1) + p(C_2) \log_2 p(C_2)) \quad (96)$$

$$= -(0.7 \log_2 0.7 + 0.3 \log_2 0.3) \quad (97)$$

$$= -(0.7 \times -0.5146 + 0.3 \times -1.737) \quad (98)$$

$$= -(-0.3602 - 0.5217) \quad (99)$$

$$= 0.8819 \quad (100)$$

Example 3: single class

Consider a dataset D with 10 examples, all belonging to a single class C_1 . The class distribution is as follows:

- $p(C_1) = \frac{10}{10} = 1$

The entropy $H(D)$ is calculated as:

$$H(D) = -(p(C_1) \log_2 p(C_1)) \quad (101)$$

$$= -(1 \log_2 1) \quad (102)$$

$$= -(1 \times 0) \quad (103)$$

$$= 0 \quad (104)$$

Information gain calculation

For a feature A with possible values $\{a_1, a_2, \dots, a_v\}$, the Information Gain $IG(D, A)$ is calculated as:

$$IG(D, A) = H(D) - \sum_{j=1}^v \frac{|D_j|}{|D|} H(D_j) \quad (105)$$

where:

- $H(D)$ is the entropy of the original dataset D .
- D_j is the subset of D where feature A has value a_j .
- $|D_j|$ is the number of examples in D_j .
- $|D|$ is the total number of examples in D .
- $H(D_j)$ is the entropy of subset D_j .

Example of information gain calculation

Consider a dataset D with 10 examples and two classes C_1 and C_2 . The class distribution is:

- 7 examples belong to class C_1
- 3 examples belong to class C_2

$$H(D) = - \left(\frac{7}{10} \log_2 \frac{7}{10} + \frac{3}{10} \log_2 \frac{3}{10} \right) \quad (106)$$

$$= -(0.7 \log_2 0.7 + 0.3 \log_2 0.3) \quad (107)$$

$$= -(0.7 \times -0.5146 + 0.3 \times -1.737) \quad (108)$$

$$= 0.8819 \quad (109)$$

After splitting:

For feature A :

- D_1 : 6 examples in C_1 , 1 in C_2
- D_2 : 1 in C_1 , 2 in C_2 Entropies of D_1 and D_2 :

$$H(D_1) = 0.5917, H(D_2) = 0.9183 \quad (110)$$

$$H_{\text{split}} = 0.6897 \quad (111)$$

$$IG(D, A) = 0.1922 \quad (112)$$

This demonstrates how Information Gain is calculated and used to rank features based on their effectiveness in reducing uncertainty. The details of these calculations provide the foundation for selecting the most valuable features in the dataset.

Received: 2 July 2024; Accepted: 14 October 2024

Published online: 18 October 2024

References

- Yavas, C. E., Chen, L., Kadlec, C. & Ji, Y. Predictive modeling of earthquakes in los angeles with machine learning and neural networks. *IEEE Access*. **12**, 108673–108702. <https://doi.org/10.1109/access.2024.3438556> (2024).
- Yavas, C. E., Chen, L., Kadlec, C. & Ji, Y. Near-perfect precision: ML and nn for 30-day earthquake forecasts. *Machine Learning and Knowledge Extraction* (2024). Submitted.
- Yavas, C. E., Chen, L., Kadlec, C. & Ji, Y. Machine learning for 30-day earthquake forecasts for san diego, california. *Machine Learning: Science and Technology* (2024). Submitted.
- Wolfe, C. J., Okubo, P. G., Ekström, G., Nettles, M. & Shearer, P. M. Characteristics of deep (≥ 13 km) Hawaiian earthquakes and Hawaiian earthquakes west of 155.55° W. *Geochemistry, Geophysics, Geosystems* **5**, 2003GC000618, <https://doi.org/10.1029/2003GC000618> (2004).
- Kossobokov, V. G. & Soloviev, A. A. Testing Earthquake Prediction Algorithms. *Journal of the Geological Society of India*. **97**, 1514–1519. <https://doi.org/10.1007/s12594-021-1907-8> (2021).
- Luo, G., Ding, F., Ma, H. & Yang, M. Pre-quake frequency characteristics of $M_s \geq 7.0$ earthquakes in mainland China. *Frontiers in Earth Science*. **10**, 992858, <https://doi.org/10.3389/feart.2022.992858> (2023).
- Johnson, P. A. et al. Laboratory earthquake forecasting: A machine learning competition. *Proceedings of the National Academy of Sciences*. **118**, e2011362118. <https://doi.org/10.1073/pnas.2011362118> (2021).
- Olsen, K. B., Archuleta, R. J. & Matarese, J. R. Three-Dimensional Simulation of a Magnitude 7.75 Earthquake on the San Andreas Fault. *Science*. **270**, 1628–1632, <https://doi.org/10.1126/science.270.5242.1628> (1995).
- Asim, K. M., Idris, A., Iqbal, T. & Martínez-Alvarez, F. Earthquake prediction model using support vector regressor and hybrid neural networks. *PLOS ONE*. **13**, e0199004. <https://doi.org/10.1371/journal.pone.0199004> (2018).
- Zhang, L., Si, L., Yang, H., Hu, Y. & Qiu, J. Precursory Pattern Based Feature Extraction Techniques for Earthquake Prediction. *IEEE Access*. **7**, 30991–31001. <https://doi.org/10.1109/ACCESS.2019.2902224> (2019).
- Skrickij, V. et al. Visual Measurement System for Wheel-Rail Lateral Position Evaluation. *Sensors*. **21**, 1297. <https://doi.org/10.3390/s21041297> (2021).
- Bilal, M. A., Ji, Y., Wang, Y., Akhter, M. P. & Yaqub, M. Early Earthquake Detection Using Batch Normalization Graph Convolutional Neural Network (BNGCNN). *Applied Sciences*. **12**, 7548. <https://doi.org/10.3390/app12157548> (2022).
- Hsu, T. & Pratomo, A. Early Peak Ground Acceleration Prediction for On-Site Earthquake Early Warning Using LSTM Neural Network. *Frontiers in Earth Science*. **10**, 911947. <https://doi.org/10.3389/feart.2022.911947> (2022).
- Olsen, K. B. Site Amplification in the Los Angeles Basin from Three-Dimensional Modeling of Ground Motion. *Bulletin of the Seismological Society of America*. **90**, S77–S94. <https://doi.org/10.1785/0120000506> (2000).
- Donnellan, A. et al. Potential for a large earthquake near Los Angeles inferred from the 2014 La Habra earthquake. *Earth and Space Science*. **2**, 378–385. <https://doi.org/10.1002/2015EA000113> (2015).
- Hauksson, E. Earthquakes, faulting, and stress in the Los Angeles Basin. *Journal of Geophysical Research: Solid Earth*. **95**, 15365–15394. <https://doi.org/10.1029/JB095iB10p15365> (1990).
- Shen, Z., Jackson, D. D. & Ge, B. X. Crustal deformation across and beyond the Los Angeles basin from geodetic measurements. *Journal of Geophysical Research: Solid Earth*. **101**, 27957–27980. <https://doi.org/10.1029/96JB02544> (1996).
- Loveless, J. P. & Meade, B. J. Stress modulation on the San Andreas fault by interseismic fault system interactions. *Geology*. **39**, 1035–1038. <https://doi.org/10.1130/G32215.1> (2011).

19. Romero, N., O'Rourke, T. D., Nozick, L. K. & Davis, C. A. Seismic Hazards and Water Supply Performance. *Journal of Earthquake Engineering*. **14**, 1022–1043. <https://doi.org/10.1080/13632460903527989> (2010).
20. Roten, D., Olsen, K. B., Day, S. M., Cui, Y. & Fäh, D. Expected seismic shaking in Los Angeles reduced by San Andreas fault zone plasticity. *Geophysical Research Letters*. **41**, 2769–2777, <https://doi.org/10.1002/2014GL059411> (2014).
21. Shaw, J. H. & Suppe, J. Earthquake hazards of active blind-thrust faults under the central Los Angeles basin, California. *Journal of Geophysical Research: Solid Earth*. **101**, 8623–8642. <https://doi.org/10.1029/95JB03453> (1996).
22. Zechar, J. D. & Jordan, T. H. Testing alarm-based earthquake predictions. *Geophysical Journal International*. **172**, 715–724. <https://doi.org/10.1111/j.1365-246X.2007.03676.x> (2008).
23. Huang, X., Luo, M. & Jin, H. Application of improved ELM algorithm in the prediction of earthquake casualties. *PLOS ONE*. **15**, e0235236, <https://doi.org/10.1371/journal.pone.0235236> (2020).
24. the RELM Working Group *et al.* First Results of the Regional Earthquake Likelihood Models Experiment. *Pure and Applied Geophysics*. **167**, 859–876, <https://doi.org/10.1007/s0024-010-0081-5> (2010).
25. Al-Heety, E. A., Rafea, H. F. & Mohammad, O. J. Evaluation of Return Period and Occurrence Probability of the Maximum Magnitude Earthquakes in Iraq and Surroundings. *IOP Conference Series: Earth and Environmental Science*. **1300**, 012001. <https://doi.org/10.1088/1755-1315/1300/1/012001> (2024).
26. Rubinstein, J. L., Ellsworth, W. L., Chen, K. H. & Uchida, N. Fixed recurrence and slip models better predict earthquake behavior than the time- and slip-predictable models: I. Repeating earthquakes. *Journal of Geophysical Research: Solid Earth*. **117**, 2011JB008724, <https://doi.org/10.1029/2011JB008724> (2012).
27. Yang, X., Du, S. & Ma, J. Do Earthquakes Exhibit Self-Organized Criticality?. *Physical Review Letters*. **92**, 228501. <https://doi.org/10.1103/PhysRevLett.92.228501> (2004).
28. Huang, J., Wang, X., Zhao, Y., Xin, C. & Xiang, H. LARGE EARTHQUAKE MAGNITUDE PREDICTION IN TAIWAN BASED ON DEEP LEARNING NEURAL NETWORK. *Neural Network World*. **28**, 149–160, <https://doi.org/10.14311/NNW.2018.28.009> (2018).
29. Kavianpour, P., Kavianpour, M., Jahani, E. & Ramezani, A. A CNN-BiLSTM model with attention mechanism for earthquake prediction. *The Journal of Supercomputing*. **79**, 19194–19226. <https://doi.org/10.1007/s11227-023-05369-y> (2023).
30. Geller, R. J., Jackson, D. D., Kagan, Y. Y. & Mulargia, F. Earthquakes Cannot Be Predicted. *Science*. **275**, 1616–1616. <https://doi.org/10.1126/science.275.5306.1616> (1997).
31. Eberhard, D., A. J., Zechar, J. D. & Wiemer, S. A prospective earthquake forecast experiment in the western Pacific: A prospective earthquake forecast experiment in the western Pacific. *Geophysical Journal International*. **190**, 1579–1592. <https://doi.org/10.1111/j.1365-246X.2012.05548.x> (2012).
32. Tehseen, R., Farooq, M. S. & Abid, A. Earthquake Prediction Using Expert Systems: A Systematic Mapping Study. *Sustainability*. **12**, 2420. <https://doi.org/10.3390/su12062420> (2020).
33. Ogata, Y. A Prospect of Earthquake Prediction Research. *Statistical Science*. **28**, <https://doi.org/10.1214/13-STS439> (2013).
34. Banna, M. H. A. *et al.* Attention-Based Bi-Directional Long-Short Term Memory Network for Earthquake Prediction. *IEEE Access*. **9**, 56589–56603. <https://doi.org/10.1109/ACCESS.2021.3071400> (2021).
35. Kagan, Y. Y. Are earthquakes predictable?. *Geophysical Journal International*. **131**, 505–525. <https://doi.org/10.1111/j.1365-246X.1997.tb06595.x> (1997).
36. Ma, N., Bai, Y. & Meng, S. Return Period Evaluation of the Largest Possible Earthquake Magnitudes in Mainland China Based on Extreme Value Theory. *Sensors*. **21**, 3519. <https://doi.org/10.3390/s21103519> (2021).
37. Velasco Herrera, V. M. *et al.* Long-Term Forecasting of Strong Earthquakes in North America, South America, Japan, Southern China and Northern India With Machine Learning. *Frontiers in Earth Science*. **10**, 905792. <https://doi.org/10.3389/feart.2022.905792> (2022).
38. Michael, A. J. Testing prediction methods: Earthquake clustering versus the Poisson Model. *Geophysical Research Letters*. **24**, 1891–1894. <https://doi.org/10.1029/97GL01928> (1997).
39. Koder, Y. *et al.* Earthquake early warning for the 2016 Kumamoto earthquake: performance evaluation of the current system and the next-generation methods of the Japan Meteorological Agency. *Earth, Planets and Space*. **68**, 202. <https://doi.org/10.1186/s40623-016-0567-1> (2016).
40. Yuan, X. *et al.* Analysis and Prediction of the SARIMA Model for a Time Interval of Earthquakes in the Longmenshan Fault Zone. In Mokhtari, M. (ed.) *Natural Hazards - New Insights*, <https://doi.org/10.5772/intechopen.109174> (IntechOpen, 2023).
41. Hajikhodaverdikhan, P., Nazari, M., Mohsenizadeh, M., Shamshirband, S. & Chau, K.-W. Earthquake prediction with meteorological data by particle filter-based support vector regression. *Engineering Applications of Computational Fluid Mechanics*. **12**, 679–688. <https://doi.org/10.1080/19942060.2018.1512010> (2018).
42. Astuti, W., Sediono, W., Akmeliawati, R., Albinu, A. M. & Salami, M. J. E. Investigation of the characteristics of geoelectric field signals prior to earthquakes using adaptive STFT techniques. *Natural Hazards and Earth System Sciences*. **13**, 1679–1686. <https://doi.org/10.5194/nhess-13-1679-2013> (2013).
43. Nishikawa, T. Comparison of statistical low-frequency earthquake activity models, <https://doi.org/10.21203/rs.3.rs-3780230/v1> (2023).
44. Nimmagadda, S. L. & Dreher, H. Ontology based data warehouse modeling and mining of earthquake data: prediction analysis along Eurasian-Australian continental plates. In *2007 5th IEEE International Conference on Industrial Informatics*, 597–602, <https://doi.org/10.1109/INDIN.2007.4384825> (IEEE, Vienna, Austria, 2007). ISSN: 1935-4576.
45. Narasimha Prasad, L. V., Murthy, P. S. & Kumar Reddy, C. K. Analysis of magnitude for earthquake detection using primary waves and secondary waves. In *2013 International Conference on Human Computer Interactions (ICHCI)*, 1–6, <https://doi.org/10.1109/ICHCI-IEEE.2013.6887820> (IEEE, Chennai, India, 2013).
46. Yang, F. *et al.* Auto-REP: An Automated Regression Pipeline Approach for High-efficiency Earthquake Prediction Using LANL Data. In *2022 14th International Conference on Computer and Automation Engineering (ICCAE)*, 127–134, <https://doi.org/10.1109/ICCAE55086.2022.9762437> (IEEE, Brisbane, Australia, 2022).
47. Zheng, X. & Tao, Z. Preliminary Evaluation of Crustal Medium Parameters in Western China. *E3S Web of Conferences*. **406**, 01003, <https://doi.org/10.1051/e3sconf/202340601003> (2023).
48. Hussain, H., Shuangxi, Z., Usman, M. & Abid, M. Spatial Variation of b-Values and Their Relationship with the Fault Blocks in the Western Part of the Tibetan Plateau and Its Surrounding Areas. *Entropy*. **22**, 1016. <https://doi.org/10.3390/e22091016> (2020).
49. Gitis, V., Derendyaev, A. & Petrov, K. Analyzing the Performance of GPS Data for Earthquake Prediction. *Remote Sensing*. **13**, 1842. <https://doi.org/10.3390/rs131091842> (2021).
50. Zhai, D., Zhang, X. & Xiong, P. Detecting Thermal Anomalies of Earthquake Process Within Outgoing Longwave Radiation Using Time Series Forecasting Models. *Annals of Geophysics*. **63**, 7. <https://doi.org/10.4401/ag-8057> (2020).
51. Woith, H., Petersen, G. M., Hainzl, S. & Dahm, T. Review: Can Animals Predict Earthquakes?. *Bulletin of the Seismological Society of America*. **108**, 1031–1045. <https://doi.org/10.1785/0120170313> (2018).
52. Kanamori, H. & Brodsky, E. E. The Physics of Earthquakes. *Physics Today*. **54**, 34–40. <https://doi.org/10.1063/1.1387590> (2001).
53. (SCEDC), S. C. E. D. C. Southern California earthquake data center (2024). Accessed: 2024-06-16.
54. (SCEDC), S. C. E. D. C. Radius search tool (2024). Accessed: 2024-06-16.
55. (SCEDC), S. C. E. D. C. Scedc change history (2024). Accessed: 2024-06-16.
56. (USGS), U. S. G. S. Magnitude types and definitions (2024). Accessed: 2024-06-16.

57. Tang, B.-h. & Chen, Q. Dynamic Prediction of Casualties after Earthquakes Based on Systematic Review and Empirical Data. In *Proceedings of the 2019 International Conference on Modeling, Analysis, Simulation Technologies and Applications (MASTA 2019)*, <https://doi.org/10.2991/masta-19.2019.51> (Atlantis Press, Hangzhou, China, 2019).
58. Jia, Z. et al. Traumatic experiences and mental health consequences among child survivors of the 2008 Sichuan earthquake: a community-based follow-up study. *BMC Public Health.* **13**, 104. <https://doi.org/10.1186/1471-2458-13-104> (2013).
59. Zhang, L., Liu, Y., Liu, X. & Zhang, Y. Rescue efforts management and characteristics of casualties of the Wenchuan earthquake in China. *Emergency Medicine Journal.* **28**, 618–622. <https://doi.org/10.1136/EMJ.2009.087296> (2011).
60. Guo, T. T., Xu, X. W. & Yu, G. H. Disasters Characteristics of 2008 Wenchuan Earthquake and Analysis of Buildings Destruction. *Applied Mechanics and Materials.* **204–208**, 2505–2513. <https://doi.org/10.4028/www.scientific.net/AMM.204-208.2505> (2012).
61. Li, W. S., Chan, S. Y., Chau, W. W., Law, S.-W. & Chan, K. M. Mobility, prosthesis use and health-related quality of life of bilateral lower limb amputees from the 2008 Sichuan earthquake. *Prosthetics & Orthotics International.* **43**, 104–111. <https://doi.org/10.1177/0309364618792720> (2019).
62. Kalantar Motamed, M. H., Sagafinia, M., Ebrahimi, A., Shams, E. & Kalantar Motamed, M. Major Earthquakes of the Past Decade (2000–2010): A Comparative Review of Various Aspects of Management. *Trauma Monthly.* **17**, 219–229, <https://doi.org/10.5812/traumamon.4519> (2012).
63. Yavas, C. E., Chen, L., Kadlec, C. & Ji, Y. Los Angeles, California. *Earthquake Dataset with Feature-Engineered Variables.* <https://doi.org/10.5281/ZENODO.13738726> (2024).
64. Ghose, S. L. et al. Continent-wide recent emergence of a global pathogen in African amphibians. *Frontiers in Conservation Science.* **4**, 1069490. <https://doi.org/10.3389/fcosc.2023.1069490> (2023).
65. Li, Liu & Qi. Regional Heterogeneity of Migrant Rent Affordability Stress in Urban China: A Comparison between Skilled and Unskilled Migrants at Prefecture Level and Above. *Sustainability.* **11**, 5920, <https://doi.org/10.3390/su11215920> (2019).
66. Lima, M. L., Romanelli, A. & Massone, H. E. Decision support model for assessing aquifer pollution hazard and prioritizing groundwater resources management in the wet Pampa plain. *Argentina. Environmental Monitoring and Assessment.* **185**, 5125–5139. <https://doi.org/10.1007/s10661-012-2930-4> (2013).
67. Muttitanon, W. Clustering Analysis Influenza Disease to Identify Spatio-Temporal Spread Pattern in Thailand. *International Journal of Geoinformatics* 81–89, <https://doi.org/10.52939/ijg.v17i5.2015> (2021).
68. Sinha, R., Singh, S., Mishra, K. & Swarnkar, S. Channel morphodynamics and sediment budget of the Lower Ganga River using a hydrogeomorphological approach. *Earth Surface Processes and Landforms.* **48**, 14–33. <https://doi.org/10.1002/esp.5325> (2023).
69. Fernández-Álvarez, R. & Fernández-Navá, R. Adaptive co-management of urban forests: monitoring reforestation programs in Mexico City. *Polibotánica.* **0**, <https://doi.org/10.18387/polibotanica.49.15> (2020).
70. Mustafa, A., Van Rompaey, A., Cools, M., Saadi, I. & Teller, J. Addressing the determinants of built-up expansion and densification processes at the regional scale. *Urban Studies.* **55**, 3279–3298. <https://doi.org/10.1177/0042098017749176> (2018).
71. Lancellotti, B. V. et al. Complex Drivers of Riparian Soil Oxygen Variability Revealed Using Self-Organizing Maps. *Water Resources Research.* **59**, e2022WR034022, <https://doi.org/10.1029/2022WR034022> (2023).
72. Babuna, P., Yang, X. & Bian, D. Water Use Inequality and Efficiency Assessments in the Yangtze River Economic Delta of China. *Water.* **12**, 1709. <https://doi.org/10.3390/w12061709> (2020).
73. Xia, Q. et al. The superposition effects of air pollution on government health expenditure in China- spatial evidence from GeoDetector. *BMC Public Health.* **22**, 1411. <https://doi.org/10.1186/s12889-022-13702-y> (2022).
74. Vazquez, C. et al. The effects of increasing land use intensity on soil nematodes: A turn towards specialism. *Functional Ecology.* **33**, 2003–2016. <https://doi.org/10.1111/1365-2435.13417> (2019).
75. Jiang, B. Head/Tail Breaks: A New Classification Scheme for Data with a Heavy-Tailed Distribution. *The Professional Geographer.* **65**, 482–494. <https://doi.org/10.1080/00330124.2012.700499> (2013).
76. Somerville, P. G. A post-Tohoku earthquake review of earthquake probabilities in the Southern Kanto District. *Japan. Geoscience Letters.* **1**, 10. <https://doi.org/10.1186/2196-4092-1-10> (2014).
77. Holschneider, M., Zöller, G., Clements, R. & Schorlemmer, D. Can we test for the maximum possible earthquake magnitude?. *Journal of Geophysical Research: Solid Earth.* **119**, 2019–2028. <https://doi.org/10.1002/2013JB010319> (2014).
78. Taroni, M., Vocatelli, G. & De Polis, A. Gutenberg-Richter B-Value Time Series Forecasting: A Weighted Likelihood Approach. *Forecasting.* **3**, 561–569. <https://doi.org/10.3390/forecast3030035> (2021).
79. Volant, P., Grasso, J., Chatelain, J. & Frogneux, M. b-Value, aseismic deformation and brittle failure within an isolated geological object: Evidences from dome structure loaded by fluid extraction. *Geophysical Research Letters.* **19**, 1149–1152. <https://doi.org/10.1029/92GL01074> (1992).
80. Yousefzadeh, M., Hosseini, S. A. & Farnagh, M. Spatiotemporally explicit earthquake prediction using deep neural network. *Soil Dynamics and Earthquake Engineering.* **144**, 106663. <https://doi.org/10.1016/j.soildyn.2021.106663> (2021).
81. Baselga, S. A combined estimator using TEC and b-value for large earthquake prediction. *Acta Geodaetica et Geophysica.* **55**, 63–82. <https://doi.org/10.1007/s40328-019-00281-5> (2020).
82. implications for earthquake prediction. Chen, J. & Zhu, S. Spatial and temporal b-value precursors preceding the 2008 Wenchuan, China, earthquake ($M_w = 7.9$). *Geomatics, Natural Hazards and Risk.* **11**, 1196–1211. <https://doi.org/10.1080/19475705.2020.1784297> (2020).
83. Martinsson, J. & Törnman, W. Modelling the Dynamic Relationship Between Mining Induced Seismic Activity and Production Rates, Depth and Size: A Mine-Wide Hierarchical Model. *Pure and Applied Geophysics.* **177**, 2619–2639. <https://doi.org/10.1007/s00024-019-02378-y> (2020).
84. Bohnhoff, M., Malin, P., Ter Heege, J., Deflandre, J.-P. & Sicking, C. Suggested best practice for seismic monitoring and characterization of non-conventional reservoirs. *First Break.* **36**, 59–64. <https://doi.org/10.3997/1365-2397.n0070> (2018).
85. Shodiq, M. N., Kusuma, D. H., Rifqi, M. G., Barakbah, A. R. & Harsono, T. Adaptive Neural Fuzzy Inference System and Automatic Clustering for Earthquake Prediction in Indonesia. *JOIV: International Journal on Informatics Visualization.* **3**, 47–53, <https://doi.org/10.30630/joiv.3.1.204> (2019).
86. Salam, M. A., Ibrahim, L. & Abdelminaam, D. S. Earthquake Prediction using Hybrid Machine Learning Techniques. *International Journal of Advanced Computer Science and Applications.* **12**, <https://doi.org/10.14569/IJACSA.2021.0120578> (2021).
87. Hashemi, M. & Karimi, H. A. Seismic Source Modeling by Clustering Earthquakes and Predicting Earthquake Magnitudes. In Leon-Garcia, A. et al. (eds.) *Smart City 360°*, vol. 166, 468–478, https://doi.org/10.1007/978-3-319-33681-7_39 (Springer International Publishing, Cham, 2016). Series Title: Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering.
88. Michael, A. J. Testing prediction methods: Earthquake clustering versus the Poisson Model. *Geophysical Research Letters.* **24**, 1891–1894. <https://doi.org/10.1029/97GL01928> (1997).
89. Priambodo, B., Mahmudy, W. F. & Rahman, M. A. Earthquake Magnitude and Grid-Based Location Prediction using Backpropagation Neural Network. *Knowledge Engineering and Data Science.* **3**, 28–39, <https://doi.org/10.17977/um018v3i12020p28-39> (2020).
90. Gitis, V. & Derendyaev, A. The Method of the Minimum Area of Alarm for Earthquake Magnitude Prediction. *Frontiers in Earth Science.* **11**, 585317. <https://doi.org/10.3389/feart.2020.585317> (2020).
91. Korkmaz, M. A study over the general formula of regression sum of squares in multiple linear regression. *Numerical Methods for Partial Differential Equations.* **37**, 406–421. <https://doi.org/10.1002/num.22533> (2021).

92. Turcotte, D. L. & Rundle, J. B. Self-organized complexity in the physical, biological, and social sciences. *Proceedings of the National Academy of Sciences*. **99**, 2463–2465. <https://doi.org/10.1073/pnas.012579399> (2002).
93. Fahandezhsadi, M. & Sadi, H. F. Earthquake Magnitude Prediction using Probabilistic Classifiers, <https://doi.org/10.21203/rs.3.rs-36094/v1> (2020).
94. Saichev, A. & Sornette, D. Distribution of the largest aftershocks in branching models of triggered seismicity: Theory of the universal Bâth law. *Physical Review E*. **71**, 056127. <https://doi.org/10.1103/PhysRevE.71.056127> (2005).
95. Radzyner, Y., Galun, M. & Nadler, B. A Statistical Approach to Estimate Seismic Monitoring Stations' Biases and Error Levels. *Bulletin of the Seismological Society of America*. **113**, 2596–2614. <https://doi.org/10.1785/0120230009> (2023).
96. Sadhukhan, B., Chakraborty, S., Mukherjee, S. & Samanta, R. K. Climatic and seismic data-driven deep learning model for earthquake magnitude prediction. *Frontiers in Earth Science*. **11**, 1082832. <https://doi.org/10.3389/feart.2023.1082832> (2023).
97. Faro, D., McGill, A. L. & Hastie, R. Naïve theories of causal force and compression of elapsed time judgments. *Journal of Personality and Social Psychology*. **98**, 683–701. <https://doi.org/10.1037/a0019261> (2010).
98. Nguyen, A. et al. Time Matters: Time-Aware LSTMs for Predictive Business Process Monitoring, <https://doi.org/10.48550/ARXIV.2010.00889> (2020). Version Number: 3.
99. Zhou, W., Liang, Y., Ming, Z. & Dong, H. EARTHQUAKE PREDICTION MODEL BASED ON DANGER THEORY IN ARTIFICIAL IMMUNITY. *Neural Network World*. **30**, 231–247, <https://doi.org/10.14311/NNW.2020.30.016> (2020).
100. Kwiatek, G. et al. Months-long preparation of the 2023 MW 7.8 Kahramanmaraş earthquake, Türkiye, <https://doi.org/10.21203/rs.3.rs-2657873/v1> (2023).
101. Salam, M. A., Ibrahim, L. & Abdelminaam, D. S. Earthquake Prediction using Hybrid Machine Learning Techniques. *International Journal of Advanced Computer Science and Applications*. **12**, <https://doi.org/10.14569/IJACSA.2021.0120578> (2021).
102. Rosenau, M. & Oncken, O. Fore-arc deformation controls frequency-size distribution of megathrust earthquakes in subduction zones. *Journal of Geophysical Research: Solid Earth*. **114**, 2009JB006359, <https://doi.org/10.1029/2009JB006359> (2009).
103. Chen, X., Wang, M., Chuan, Y., Wei, Y. & Zhang, P. Topographic Controls on the Distribution of Coseismic Landslides: A Case Study Using the Coefficient of Variation of the 2014 Ludian, Yunnan, China, Ms6.5 Earthquake. *Lithosphere*. **2021**, 6678652, <https://doi.org/10.2113/2022/6678652> (2022).
104. Muhammad, D., Ahmad, I., Khalil, M. I., Khalil, W. & Ahmad, M. O. A Generalized Deep Learning Approach to Seismic Activity Prediction. *Applied Sciences*. **13**, 1598. <https://doi.org/10.3390/app13031598> (2023).
105. Ziebarth, M. J., Von Speccht, S., Heidbach, O., Cotton, F. & Anderson, J. G. Applying Conservation of Energy to Estimate Earthquake Frequencies from Strain Rates and Stresses. *Journal of Geophysical Research: Solid Earth*. **125**, e2020JB020186, <https://doi.org/10.1029/2020JB020186> (2020).
106. Hibert, C., Michéa, D., Provost, F., Malet, J.-P. & Geertsema, M. Exploration of continuous seismic recordings with a machine learning approach to document 20 yr of landslide activity in Alaska. *Geophysical Journal International*. **219**, 1138–1147. <https://doi.org/10.1093/gji/ggz354> (2019).
107. Grout, C., Hibert, C., Malet, J.-P. & Provost, F. *Identifying landslides from massive seismic data and machine learning: the case of the European Alps*. <https://doi.org/10.5194/egusphere-egu23-7062> (2023).
108. Jufriansah, A., Khusnani, A., Saputra, S. & Suwandi Wahab, D. Forecasting the Magnitude Category Based on The Flores Sea Earthquake. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*. **7**, 1439–1447, <https://doi.org/10.29207/resti.v7i6.5495> (2023).
109. Brykov, M. N. et al. Machine Learning Modelling and Feature Engineering in Seismology Experiment. *Sensors*. **20**, 4228. <https://doi.org/10.3390/s20154228> (2020).
110. Jena, R. et al. Explainable Artificial Intelligence (XAI) Model for Earthquake Spatial Probability Assessment in Arabian Peninsula. *Remote Sensing*. **15**, 2248. <https://doi.org/10.3390/rs15092248> (2023).
111. Tsuboi, S., Saito, M. & Kikuchi, M. Real-time earthquake warning by using broadband P Waveform. *Geophysical Research Letters*. **29**, <https://doi.org/10.1029/2002GL016101> (2002).
112. Chittora, P. et al. Experimental analysis of earthquake prediction using machine learning classifiers, curve fitting, and neural modeling, <https://doi.org/10.21203/rs.3.rs-1896823/v2> (2022).
113. Biswas, S., Kumar, D. & Bera, U. K. Prediction of earthquake magnitude and seismic vulnerability mapping using artificial intelligence techniques: a case study of Turkey, <https://doi.org/10.21203/rs.3.rs-2863887/v1> (2023).
114. Liu, Y. et al. Applying interpretable machine learning algorithms to predict risk factors for permanent stoma in patients after TME. *Frontiers in Surgery*. **10**, 1125875. <https://doi.org/10.3389/fsurg.2023.1125875> (2023).
115. Marc, O., Meunier, P. & Hovius, N. Prediction of the area affected by earthquake-induced landsliding based on seismological parameters. *Natural Hazards and Earth System Sciences*. **17**, 1159–1175. <https://doi.org/10.5194/nhess-17-1159-2017> (2017).
116. Mahmoudi, J., Arjomand, M. A., Rezaei, M. & Mohammadi, M. H. Predicting the Earthquake Magnitude Using the Multilayer Perceptron Neural Network with Two Hidden Layers. *Civil Engineering Journal*. **2**, 1–12, <https://doi.org/10.28991/cej-2016-00000008> (2016).
117. Lara, P., Bletery, Q., Ampuero, J., Inza, A. & Tavera, H. Earthquake Early Warning Starting From 3 s of Records on a Single Station With Machine Learning. *Journal of Geophysical Research: Solid Earth*. **128**, e2023JB026575, <https://doi.org/10.1029/2023JB026575> (2023).
118. Dias, V. H. A. & Papa, A. R. R. Application of Neural Networks in Probabilistic Forecasting of Earthquakes in the Southern California Region. *International Journal of Geosciences*. **09**, 397–413. <https://doi.org/10.4236/ijg.2018.96025> (2018).
119. Chuo, Y.-J. Earthquake Shake Detecting by Data Mining from Social Network Platforms. *Applied Sciences*. **10**, 812. <https://doi.org/10.3390/app10030812> (2020).
120. Şengezer, B., Ansal, A. & Bilen, O. Evaluation of parameters affecting earthquake damage by decision tree techniques. *Natural Hazards*. **47**, 547–568. <https://doi.org/10.1007/s11069-008-9238-2> (2008).
121. Cin, M. & Değirmencay, c. A. Decision-Making of Middle School Students during an Earthquake. *Review of International Geographical Education Online*. **8**, 556–570, <https://doi.org/10.33403/rigeo.505271> (2018).
122. Asgarian, B. & Ordoubadi, B. PROBABILISTIC EVALUATION OF SEISMIC PERFORMANCE FOR A STEEL MOMENT FRAME USING DAMAGE INDICES. In *Proceedings of the 6th International Conference on Computational Methods in Structural Dynamics and Earthquake Engineering (COMPDYN 2015)*, 1779–1796, <https://doi.org/10.7712/120117.5528.17007> (Institute of Structural Analysis and Antiseismic Research School of Civil Engineering National Technical University of Athens (NTUA) Greece, Rhodes Island, Greece, 2017).
123. Yamaga, N. & Mitsui, Y. Machine Learning Approach to Characterize the Postseismic Deformation of the 2011 Tohoku-Oki Earthquake Based on Recurrent Neural Network. *Geophysical Research Letters*. **46**, 11886–11892. <https://doi.org/10.1029/2019GL084578> (2019).
124. Chen, J. & Chaudhari, N. Bidirectional segmented-memory recurrent neural network for protein secondary structure prediction. *Soft Computing*. **10**, 315–324. <https://doi.org/10.1007/s00500-005-0489-5> (2006).
125. Albaba, M., Qassab, A. & Yilmaz, A. Human activity recognition and classification using of convolutional neural networks and recurrent neural networks. *International Journal of Applied Mathematics Electronics and Computers*. **8**, 185–189, <https://doi.org/10.18100/ijamec.803105> (2020).
126. Merghadi, A., Abderrahmane, B. & Tien Bui, D. Landslide Susceptibility Assessment at Mila Basin (Algeria): A Comparative Assessment of Prediction Capability of Advanced Machine Learning Methods. *ISPRS International Journal of Geo-Information*. **7**, 268. <https://doi.org/10.3390/ijgi7070268> (2018).

127. Yang, S., Jin, A. & Xu, Y. Recognition of Oil and Gas Reservoir Space Based on Deep Learning. *E3S Web of Conferences*. **267**, 01038, <https://doi.org/10.1051/e3sconf/202126701038> (2021).
128. Ross, Z. E., Meier, M., Hauksson, E. & Heaton, T. H. Generalized Seismic Phase Detection with Deep Learning. *Bulletin of the Seismological Society of America*. **108**, 2894–2901. <https://doi.org/10.1785/0120180080> (2018).
129. Yue, L. et al. Seismic event classification based on a two-step convolutional neural network, <https://doi.org/10.21203/rs.3.rs-2433400/v1> (2023).
130. Hsu, T. & Pratomo, A. Early Peak Ground Acceleration Prediction for On-Site Earthquake Early Warning Using LSTM Neural Network. *Frontiers in Earth Science*. **10**, 911947. <https://doi.org/10.3389/feart.2022.911947> (2022).
131. Cao, C. et al. Long Short-Term Memory Networks for Pattern Recognition of Synthetical Complete Earthquake Catalog. *Sustainability*. **13**, 4905. <https://doi.org/10.3390/su13094905> (2021).
132. Abri, R. & Artuner, H. LSTM-Based Deep Learning Methods for Prediction of Earthquakes Using Ionospheric Data. *Gazi University Journal of Science*. **35**, 1417–1431, <https://doi.org/10.35378/gujs.950387> (2022).
133. Azadkia, M. Optimal choice of k for k -nearest neighbor regression, <https://doi.org/10.48550/ARXIV.1909.05495> (2019). Version Number: 4.
134. Elizabeth Yancey, R., Xin, B. & Matloff, N. Modernizing k-nearest neighbors. *Stat.* **10**, e335. <https://doi.org/10.1002/sta4.335> (2021).
135. Setiyorini, T. & Asmono, R. T. PENERAPAN METODE K-NEAREST NEIGHBOR DAN GINI INDEX PADA KLASIFIKASI KINERJA SISWA. *Jurnal Techno Nusa Mandiri*. **16**, 121–126, <https://doi.org/10.33480/techno.v16i2.747> (2019).
136. Wang, W., Wu, G.-F. & Song, X.-Y. The application of neural networks to comprehensive prediction by seismology prediction method. *Acta Seismologica Sinica*. **13**, 210–215. <https://doi.org/10.1007/s11589-000-0012-0> (2000).
137. Akter, S. EARTHQUAKE PREDICTION BY USING EVIDENTIAL REASONING APPROACH. *International Journal of Research in Engineering and Technology*. **04**, 149–151, <https://doi.org/10.15623/ijret.2015.0412028> (2015).
138. Chouliaras, G. Investigating the earthquake catalog of the National Observatory of Athens. *Natural Hazards and Earth System Sciences*. **9**, 905–912. <https://doi.org/10.5194/nhess-9-905-2009> (2009).
139. Alabi, A. A., Akinyemi, O. D. & Adewale, A. Seismicity Pattern in Southern Africa from 1986 to 2009. *Earth Science Research*. **2**, p1. <https://doi.org/10.5539/esr.v2n2p1> (2012).
140. Liu, W., He, K., Gao, Q. & Liu, C.-Y. Application of EMD-Based SVD and SVM to Coal-Gangue Interface Detection. *Journal of Applied Mathematics*. **1–6**, 2014. <https://doi.org/10.1155/2014/283606> (2014).
141. Zhang, H.-M., Zhou, S., Xu, C. & Zhang, Y.-R. A REAL-TIME AUTOMATIC METHOD FOR TARGET LOCATING UNDER UNKNOWN WALL CHARACTERISTICS IN THROUGH-WALL IMAGING. *Progress In Electromagnetics Research M*. **89**, 189–197. <https://doi.org/10.2528/PIERM19111101> (2020).
142. Yin, J., Denolle, M. & He, B. A multi-task encoder-decoder to separating earthquake and ambient 1 noise signal in seismograms, <https://doi.org/10.1002/essoar.10510129.1> (2022).
143. Huang, J. et al. A hybrid machine-learning model to estimate potential debris-flow volumes. *Geomorphology*. **367**, 107333. <https://doi.org/10.1016/j.geomorph.2020.107333> (2020).
144. Xiong, P., Marchetti, D., De Santis, A., Zhang, X. & Shen, X. SafeNet: SwArm for Earthquake Perturbations Identification Using Deep Learning Networks. *Remote Sensing*. **13**, 5033. <https://doi.org/10.3390/rs13245033> (2021).
145. Morfidis, K. & Kostinakis, K. Special Issue on Application of Artificial Neural Networks for Seismic Design and Assessment. *Applied Sciences*. **12**, 6192, <https://doi.org/10.3390/app12126192> (2022).
146. Qu, H., Feng, T., Zhang, Y. & Wang, Y. Ensemble Learning with Stochastic Configuration Network for Noisy Optical Fiber Vibration Signal Recognition. *Sensors*. **19**, 3293. <https://doi.org/10.3390/s19153293> (2019).
147. Nowicki Jesse, M. A. et al. A Global Empirical Model for Near-Real-Time Assessment of Seismically Induced Landslides. *Journal of Geophysical Research: Earth Surface*. **123**, 1835–1859, <https://doi.org/10.1029/2017JF004494> (2018).
148. Mignan, A. & Broccardo, M. Neural Network Applications in Earthquake Prediction (1994–2019): Meta-Analytic and Statistical Insights on Their Limitations. *Seismological Research Letters*. **91**, 2330–2342. <https://doi.org/10.1785/0220200021> (2020).
149. Rawat, A., Chatterjee, R. S., Kumar, D., Kumar, H. & Suman, S. Can Site Specific parameters help to identify the Seismically Induced Damage Pattern: An Assessment, <https://doi.org/10.21203/rs.3.rs-2721236/v1> (2023).
150. De Vilder, S., Massey, C., Lukovic, B., Taig, T. & Morgenstern, R. What drives landslide risk? Disaggregating risk analyses, an example from the Franz Josef Glacier and Fox Glacier valleys, New Zealand. *Natural Hazards and Earth System Sciences*. **22**, 2289–2316. <https://doi.org/10.5194/nhess-22-2289-2022> (2022).
151. Elkhouly, S. H. & Ali, G. Seismic Discrimination Between Nuclear Explosions and Natural Earthquakes using Multi-Machine Learning Techniques. *Pure and Applied Geophysics*. <https://doi.org/10.1007/s00024-024-03463-7> (2024).
152. Murwantara, I. M., Yugopuspito, P. & Hermawan, R. Comparison of machine learning performance for earthquake prediction in Indonesia using 30 years historical data. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*. **18**, 1331, <https://doi.org/10.12928/telkomnika.v18i3.14756> (2020).
153. Kholid Budiman1, Y. N. I. Analysis of earthquake forecasting using random forest. *Journal of Soft Computing Exploration*. **2**, <https://doi.org/10.5246/joscx.v2i2.51> (2021).
154. Reddy, G. S. & Chittineni, S. Entropy based C4.5-SHO algorithm with information gain optimization in data mining. *PeerJ Computer Science*. **7**, e424, <https://doi.org/10.7717/peerj-cs.424> (2021).
155. Zhang, L. A Feature Selection Algorithm Integrating Maximum Classification Information and Minimum Interaction Feature Dependency Information. *Computational Intelligence and Neuroscience*. **1–10**, 2021. <https://doi.org/10.1155/2021/3569632> (2021).

Acknowledgements

This work is supported by the National Science Foundation under Grant No. 2321939. We would also like to express our gratitude to the Department of Information Technology at Georgia Southern University for their invaluable support and resources.

Author contributions

C.E.Y., L.C., C.K., and Y.J. conceived the study and designed the experiments. C.E.Y. and L.C. conducted the experiments. C.E.Y., L.C., C.K., and Y.J. analyzed the results. All authors reviewed and approved the final manuscript.

Additional information

Correspondence and requests for materials should be addressed to C.E.Y.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2024

Terms and Conditions

Springer Nature journal content, brought to you courtesy of Springer Nature Customer Service Center GmbH (“Springer Nature”).

Springer Nature supports a reasonable amount of sharing of research papers by authors, subscribers and authorised users (“Users”), for small-scale personal, non-commercial use provided that all copyright, trade and service marks and other proprietary notices are maintained. By accessing, sharing, receiving or otherwise using the Springer Nature journal content you agree to these terms of use (“Terms”). For these purposes, Springer Nature considers academic use (by researchers and students) to be non-commercial.

These Terms are supplementary and will apply in addition to any applicable website terms and conditions, a relevant site licence or a personal subscription. These Terms will prevail over any conflict or ambiguity with regards to the relevant terms, a site licence or a personal subscription (to the extent of the conflict or ambiguity only). For Creative Commons-licensed articles, the terms of the Creative Commons license used will apply.

We collect and use personal data to provide access to the Springer Nature journal content. We may also use these personal data internally within ResearchGate and Springer Nature and as agreed share it, in an anonymised way, for purposes of tracking, analysis and reporting. We will not otherwise disclose your personal data outside the ResearchGate or the Springer Nature group of companies unless we have your permission as detailed in the Privacy Policy.

While Users may use the Springer Nature journal content for small scale, personal non-commercial use, it is important to note that Users may not:

1. use such content for the purpose of providing other users with access on a regular or large scale basis or as a means to circumvent access control;
2. use such content where to do so would be considered a criminal or statutory offence in any jurisdiction, or gives rise to civil liability, or is otherwise unlawful;
3. falsely or misleadingly imply or suggest endorsement, approval , sponsorship, or association unless explicitly agreed to by Springer Nature in writing;
4. use bots or other automated methods to access the content or redirect messages
5. override any security feature or exclusionary protocol; or
6. share the content in order to create substitute for Springer Nature products or services or a systematic database of Springer Nature journal content.

In line with the restriction against commercial use, Springer Nature does not permit the creation of a product or service that creates revenue, royalties, rent or income from our content or its inclusion as part of a paid for service or for other commercial gain. Springer Nature journal content cannot be used for inter-library loans and librarians may not upload Springer Nature journal content on a large scale into their, or any other, institutional repository.

These terms of use are reviewed regularly and may be amended at any time. Springer Nature is not obligated to publish any information or content on this website and may remove it or features or functionality at our sole discretion, at any time with or without notice. Springer Nature may revoke this licence to you at any time and remove access to any copies of the Springer Nature journal content which have been saved.

To the fullest extent permitted by law, Springer Nature makes no warranties, representations or guarantees to Users, either express or implied with respect to the Springer nature journal content and all parties disclaim and waive any implied warranties or warranties imposed by law, including merchantability or fitness for any particular purpose.

Please note that these rights do not automatically extend to content, data or other material published by Springer Nature that may be licensed from third parties.

If you would like to use or distribute our Springer Nature journal content to a wider audience or on a regular basis or in any other manner not expressly permitted by these Terms, please contact Springer Nature at

onlineservice@springernature.com