International Journal of Advanced Technology and Engineering Exploration, Vol 9(95)
ISSN (Print): 2394-5443 ISSN (Online): 2394-7454
http://dx.doi.org/10.19101/IJATEE.2021.876293

Research Article

Neural network modeling of seismic behaviour of the hellenic Arc: strengths and limitations

Dariia Voloshchuk[1, 2], Antonios J. Konstantaras[1*], Alexandra Moshou[1], Nataliia Kasianova[2], Irina Skorniakova[2], Panagiotis Argyrakis[1] and Nikolaos S. Petrakis[1]
Hellenic Mediterranean University, Department of Electronic Engineering, Romanou 3, Chania, Greece, GR 73133[1]
National Aviation University, Department of Business Analytics and Digital Economy, Liubomyra Huzara Avenue, 1, Kyiv, Ukraine, 03058[2]

Received: 29-June-2022; Revised: 26-October-2022; Accepted: 27-October-2022

Abstract

The strategy of earthquake-proof construction and seismic risk reduction requires constant improvement of methods of calculation and compilation of increasingly informative normative forecast maps of seismic hazards. Despite the wide range of available methods for fixing deformations of the earth's crust, a reliable seismic forecast is still not possible because local changes in parameters do not always lead to earthquakes, and environmental heterogeneity does not allow to single out any bright shift that can make one think about future earthquakes. The introduction of modern mathematical methods and the development of the newest computer technologies based on artificial intelligence (AI) give a chance to predict the occurrence of natural disasters, in particular, earthquakes. This study aims to build a mathematical apparatus for earthquake prediction, which is based on the use of neural networks (NNs) to process large amounts of information. Artificial neural networks (ANNs) can be used to approximate any complex functional connections. The article presents the results of developing a neural network model (NNM) for forecasting occurrence numbers and sizes of medium-strong earthquakes (Mw >=4 on the Richter scale). To build a forecast NN, data on earthquakes recorded in Greece for the period 2000-2020 (about 2,500 events) were used. The NN receives input from three independent variables: geographical coordinates of the earthquake's latitude, geographical coordinates of the earthquake's longitude, and the earthquake's depth. The construction of a NN to predict strong earthquakes was implemented in the development environment RStudio programming language R. Neuralnet package was used to build the required NN, which contains a very flexible function for training feed-forward neural networks (FFNNs) and allows you to simulate many internal hidden layers and hidden network neurons. We have also used the nnet package, which is a universal tool for building predictive models in NN programming. The result is a NN of the multilayer perceptron type, which includes 2 hidden layers consisting of 5 and 3 neurons, respectively, which generate input data at the output of the network. The NN perceived model of seismicity not only describes the process of occurrence (generation) of earthquakes in Greece, but can also be used to estimate magnitudes of forthcoming seismic events.

Keywords

Artificial intelligence, Neural networks, Seismic behaviour modelling, Magnitude forecasting, Earthquakes, Hellenic Arc.

1.Introduction

In 2020, 850 cases have been reported worldwide of natural damage due to geophysical, meteorological, hydrological and climatological factors (three times more than in 1980) [1]. The first studies of seismic hazard were performed in the late XIX century in England [2].

According to statistics, strong earthquakes occur on average more than 100 times a year. The regions of the Earth where the risk of earthquakes is highest are the areas where oceanic or continental plates occur, and the areas located on both sides of such plates (*Figure 1*).

The level of seismological danger was measured and estimated at the maximum accelerations of ground motion (m/s2), which can be exceeded with a probability of 10% over the next 50 years.

*Author for correspondence
This work was supported in part by the Institutionalized CompuTech Research Lab of the Department of Electronic Engineering, Hellenic Mediterranean University, Greece under Grant HMU-ELKE-80280.

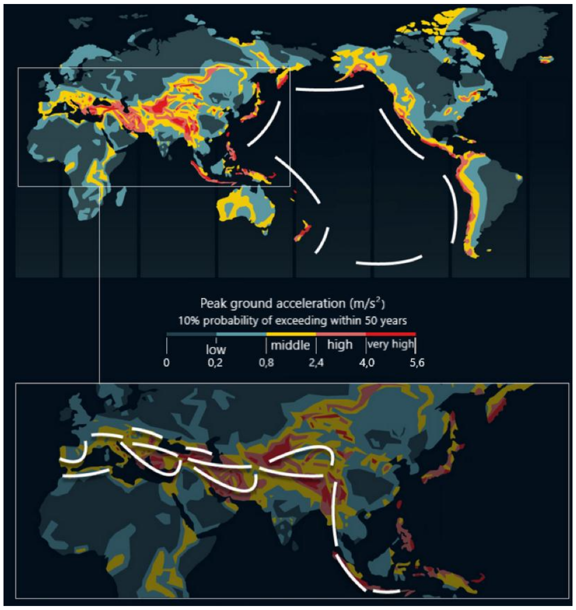

Figure 1 Map of seismic hazard of different regions of our planet (compiled by the author according to [3])

In *Figure 1* areas of low seismic hazard, which occupy more than 70% of the land, are painted in blue and blue tones. Pink and red colors illustrate the presence of a high degree of danger, which is about 8% of the earth's surface. An updated ground-motion model (GMM) for earthquakes with a magnitude of 6-9 Mw was presented in the work of scientists D.Goldberg, D.Melgar, G.Hayes, B.Crowell, and V.Sahakian using observations of the Global Navigation Satellite System (GNSS) peak ground displacement (PGD). GMM earthquake models provide information on several scientific and engineering parameters, including source characterization, seismic hazard evaluations, loss estimates, and seismic design standards. A typical GMM model is characterized by simplified metrics that describe the source of the earthquake (magnitude), the observation distance and location conditions [4].

The first examples of successful earthquake forecasting date back to the 1970s [5]. The most active problem was in the United States, Japan and China. The Americans and Japanese relied on large-scale networks to collect geophysical information and improve data processing in high seismic areas, and China on a large, hard-working and executive population.

In the following years, earthquakes were reported to be predicted in China [5], but much more often the element struck unexpectedly. At the same time, unconfirmed forecasts caused panic among the population and led to large economic losses, so in the future, it was even decided to limit the practice of evacuation measures [6].

In the USA and Japan, scientists did not limit themselves to recording the ground motion. They measured the level, temperature, and chemical composition of water in wells, the speed of ground motion, recorded anomalies of the gravitational and geomagnetic fields, and monitored atmospheric, ionospheric, and geoelectrical phenomena. At the same time, the Union of Soviet Socialist Republics (USSR) did not have enough opportunities to deploy regional observation networks, instead a number of high-class local test sites for complex geophysical observations were created [7].

Research into the processes that characterize a possible earthquake and new ways of recognizing anomalies, including the use of artificial intelligence (AI) techniques, were expected to bring success, but hopes were dashed.

The huge increase in the amount of geophysical information has not led to a qualitative improvement in the efficiency of the forecast. Geophysicists have been able to observe a large number of different physical anomalies, presumably related to processes that indicate the possibility of an earthquake (e.g., rapid movements of the earth's surface before the Haicheng earthquake), but the vast majority of them have not been observed in other earthquakes or at other sites. Despite all efforts, it has not been possible to obtain an effective and cost-effective earthquake prediction, in which the losses from the predicted disaster, which were prevented, would exceed the losses from false alarms [6, 7].

Over the past few years, space-based observation methods based on AI technologies, namely neural networks (NNs), have been widely used in earthquake prediction researches. New satellite technologies allow to monitor and analyze deformations of the Earth's surface, changes in soil temperature during deep fluid emissions, changes in the ionosphere associated with strong earthquakes. In earthquake forecasting, NASA, for example, relies on the use of the high-precision global positioning system global positioning system (GPS), as well as satellite radars that appeared a little later, with synthetic aperture Interferometric Synthetic Aperture Radar (InSAR). GPS allows to track the position of points of the earth's surface, where stationary receivers are installed, and to estimate the speed of their movement with an accuracy of millimeters. InSAR technology is used to monitor the earth's surface displacements for the established time intervals between successive surveys of the studied area. Combining GPS and InSAR data provides incredible accuracy in monitoring the movements of the earth's surface. Another task is to track the signal from these data, which allows predicting the location and strength of a future earthquake [8, 9].

Among modern European observation systems, the French program based on the Detection of Electro-Magnetic Emissions Transmitted from Earthquake Regions (DEMETER) satellite launched in 2004 was also of particular interest. It provided for both remote and ground observations to verify and link space data. This program was aimed at predicting earthquakes based on data on changes in the state of the ionosphere using artificial neural networks (ANNs) [10]. The rapid spread of AI is significant to some extent affects society, and changes the way we work, live and interact. Therefore, the most important consideration during the development of AI must be provided its usefulness to humanity, and for this it must be at the same time "Human-friendly" and "Earth-friendly". Today AI technologies are successfully developing in two directions [11]:

- Semiotic: the creation of systems that mimic such processes as speech, thinking, and expression of emotions;
- Biological: the creation of NNs that are built on the biological principle.

The potential for the use of AI is very wide because it is already used in many areas: medicine, finance, industry, trade, and, of course, human life. Because AI is the driving force of the fourth industrial revolution, its use capacity could help achieve sustainable results favorable to humanity and the planet, on which we live (*Figure 2*).

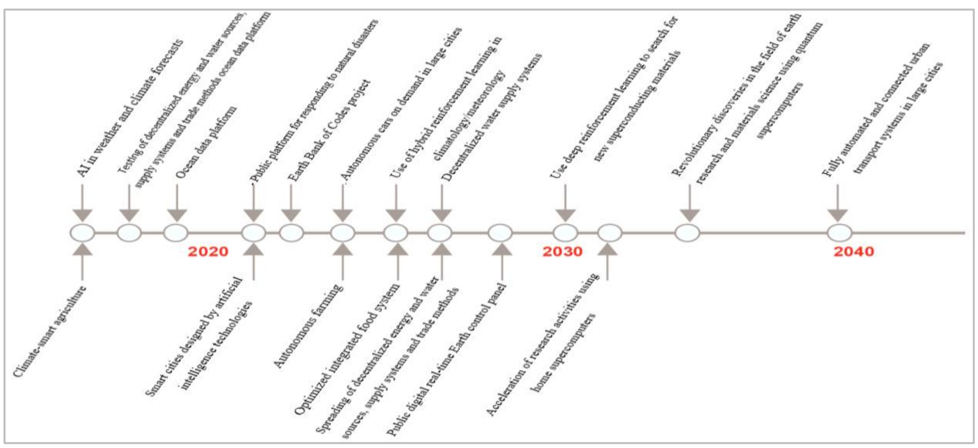

Figure 2 Cardinal factors of AI for the good of the Earth (approximate time scale) (compiled by the author according to [12])

Late advances in technologies such as parallel processing for handling and visualizing diverse big data associated to natural disasters, as well as advances in artificial intelligence, such as as deep learning, provide valuable tools in the study of complex natural phenomena with non-linear processes [13–15]. In predicting natural disasters, such as earthquakes, it is important to detect the various hidden relationships between the variables studied during the research. NNs as global approximators are capable of depicting underlying/hidden relationships between often unknown system variables. Providing that successful training of the NN has been accomplished, it is then possible to apply to the trained NN seismic data sets from neighbouring areas and investigate whether the neighboring areas also uphold similar underlying/hidden relationships [16].

In this paper, the task of earthquake magnitude prediction is set. To solve it, the application of a feed-forward multilayer neural network proved to be effective. This type of NN does not require an extended dataset, since the earthquake prediction problem is not solved as a pattern recognition problem, in which the use of a back propagation neural network (BPNN) is better suited.

This paper begins with a brief review of the most commonly used earthquake prediction methods based on ANN technologies. Next, it discusses the most commonly used type of static feed-forward neural networks (FFNNs) for predicting the occurrence of future earthquakes and explains their general operation. Then it continues with the description of the method of predicting the frequency of occurrence of strong earthquakes in Greece and the evaluation of the effectiveness of its use. Next, the general model of the NN for predicting strong earthquakes in Greece, namely the earthquake magnitude, is described, and a strategy for solving geophysical problems based on the constructed model is formulated. Finally, the built model is evaluated based on the calculated values of the total error and the general conclusions of research on the effectiveness of this method in predicting future earthquakes are formed.

2.Literature review

In the problem of earthquake prediction, ANNs are used for both prediction and pattern recognition. Such NNs as ANNs with back propagation learning algorithm [17–19], radial basic function NN (RBF) [20, 21], nonlinear autoregressive network with exogenous inputs (NARX) [22, 23], recurrent neural network (RNN) [24–26], convolutional neural network (CNN) [27–30], probabilistic neural network (PNN) [31], deep feed-forward fully connected neural network (DNN) [30], multi-layer perceptron (MLP) [32, 33], self-organizing map (SOM) [34] were used for earthquake prediction. These NNs, depending on the input data, have two most common applications: for predicting the magnitude of the possible earthquake [32, 33] and the location and time interval of the earthquake [29–35].

Statistical methods based on the abovementioned NNs are used for long-term forecasting of strong earthquakes. However, they are difficult to apply for short-term forecasting of either strong or weak earthquakes. Weak earthquakes ($Mw<3$) are rarely a matter of interest due to the waves from such earthquakes being enormously difficult to detect and track. However, the use of approaches that have proven themselves well in conditions of high similarity between the nearest seismic signals leads to a large number of false detections. Multilayer CNNs can be the most accurate in detecting weak earthquakes. For this purpose, a NN was applied to the detection of synthetic microseismic events using records from a well with a relatively low noise level [36].

Many works are devoted to the task of predicting the time interval of earthquakes [37, 38]. The interval between earthquakes is characterized by the seismic activity of each area. Kao Din Chong in his work describes an algorithm for constructing a medium-term forecast of strong earthquakes ($M>6$) based on seismic data related to the time of earthquake occurrence. The author focuses on the development of the principle of selecting the forecast time interval for medium-term earthquake prediction using a BPNN [39]. The main difficulty of this method can be considered the type of NN chosen by the author, which requires the consideration of a large number of earthquakes. Therefore, the forecasts that can be obtained from a relatively limited catalog of earthquakes in the selected area cannot be considered sufficiently accurate.

In the field of scientific research of earthquakes and their forecasting, an important task is to study aftershocks [35, 37–41]. With the help of NN Phoebe M. R. DeVries, Fernanda Viégas, Martin Wattenberg & Brendan J. Meade it is possible to investigate the earthquake foci and to predict aftershocks. The new approach will allow predicting repeated seismic shocks that may occur in the period up to a year after the main seismic shock. The NN was trained on 199 earthquakes over the past decades and 130,000 aftershocks. The coverage was 50 km vertically and 100 km horizontally from the epicenter of each earthquake [42]. The researchers note that the NN is still far from a perfect aftershock prediction, but it is characterized by extraordinary potential. New risks and challenges associated with earthquake forecasting generate increasingly complex methodologies for their modeling and prediction. Modeling algorithms combine specific NN types with different types of NN training. The NN type underlying these algorithms varies depends on the task at hand. The most appropriate NN for early earthquake prediction in the process of estimating the probable parameters of the earthquake epicenter location and magnitude is a multilayer FFNN. The main problem common to such type of NN is the adjustment of weights from input to hidden neurons. In this case, the back-propagation learning algorithm is used. The essence of the algorithm is that the errors associated with the weight coefficients of the hidden layers are determined by back-propagating the errors of the output layer neurons [43]. A cyclicity of training takes place until the error value of the weighting coefficients is minimized. Among the most typical challenges to the prediction of earthquakes in modern conditions, there are:

- the existence of unknown factors that may affect the accuracy of the built earthquake prediction model;
- the difficulty of measuring or tracking known factors that are precursors of a possible earthquake (the smell of gas in an area that is not characterized by this phenomenon, abnormal behavior of insects and other animals, sparks between closely spaced power lines, blue lighting of the inner surface of buildings);
- differentiation of factors that characterize the approach of an earthquake: deformation of the earth's crust, anomalies of geomagnetic fields and heat flow, sharp changes in the properties of rocks (electrical, seismic, etc.), geochemical anomalies, violations of the water regime, atmospheric phenomena;
- quite often the connection between these factors and earthquake occurrence is non-linear.
- Scientists face these challenges when solving such tasks of earthquake forecasting as:
- identifying the relationship between earthquake affecting coefficient and the nature of the place of occurrence using DNN [44];
- observation of electromagnetic waves characterizing the level of seismicity with using MLP [45];
- short-term forecasting based on chaotic analysis of time series containing data from recent earthquakes in a specific region using RNN [46];
- forecasting the magnitude using PNN [47] and the time and location of a strong earthquake based on modeling RNN [48], using a vector of eight mathematically calculated parameters called indicators of seismicity;
- non-linear forecasting for modeling dissection from earthquake time series using a multilayered FFNN model based on the back-propagation algorithm [49];
- finding the relationship between radon and earthquake based on three-layer Levenberg-Marquardt feed-forward learning algorithm [50];
- analysis of spatio-temporal electric field data measured by various stations as a precursor to an earthquake in relation to regional seismicity using SOM based NN [51] etc.

3.Materials and methods

3.1Artificial intelligence technologies and other state-of-the-art methods for earthquake prediction

Various approaches are used to predict earthquakes, including methods and algorithms based on expert systems [13]. Among the most effective methods of machine learning in the earthquake, prediction is ANNs, the method of reference vectors, the method of k-nearest neighbors, the naive Bayesian classifier and the "random forest" algorithm proposed by L. Breiman and A. Cutler [52].

Several promising methods employ various AI technologies aiming toward earthquake prediction yet forecasting earthquakes remains an unsolved problem and open front in seismology.

Determinist methods of seismic hazard analysis based on geological data were described by E. Krinitzsky, who determined that when forecasting earthquakes, the main thing is to determine the nearest active fault and calculate the maximum possible earthquake. Then the maximum possible intensity of seismic shocks on the Earth's surface is calculated based on the theory of seismic waves, taking into account their attenuation as they move away from the epicenter of the earthquake and the impact on local soils, which are usually presented in the form of a horizontally layered medium [53].

The risk of earthquakes increases in urban areas due to high population density and extensive infrastructure. However, these factors also complicate earthquake forecasting. The logistical difficulties of deploying detecting devices, as well as the seismic noise that the city is constantly creating, complicate an already difficult task. Traditional earthquake detecting methods that look for seismic wave-related events above a certain noise level may simply miss less powerful signals. Also, although the detection threshold for small earthquakes can be lowered by relying, for example, on the local similarity of signals, seismic noise attenuation itself can significantly increase the sensitivity to detect the desired seismic signals. Among the researches of scientists who have been actively engaged in the investigation of the consequences of error detection and earthquake location, one can also distinguish the works of Husen and Hardebeck, Stabile et al., Zaliapin and Ben-Zion [54–56].

Traditional noise reduction methods are based on simple spectral filtering of signals and are therefore ineffective if the earthquake signal is overlapped by noise in the same frequency range. The problem can be solved by noise attenuation in the frequency-time domain, which will help to detect a seismic signal against the background of noise. Traditional detection methods look for energy pulses with amplitudes that exceed the detection threshold and cannot distinguish an earthquake from other signals, such as waveforms generated, such as traffic. However, this requires a complex analysis of a large array of data in search of the necessary relationships between signals, which is almost an ideal task for machine learning methods [16, 57].

Earthquake detection and wave phase selection are quite time-consuming procedures performed by analysts when processing seismic records. Phase selection consists of estimating the time of arrival of the primary (P) and secondary (S) waves at the seismic monitoring station. The authors Sergi Mus Leon, Beatriz Otero Calvino, Leonardo Alvarado Vivas et al. developed (FFNNs) and CNN to explore numerous architecture configurations to find appropriate hyperparameter patterns for efficient P wave earthquake detection [58]. The algorithm developed by the authors requires some post-processing of the network output based on an adequate level of triggering, which is not a final solution to solve the issue of accurate prediction of earthquake occurrence.

Another approach to earthquake forecasting estimates is based upon potential relationships between the depth of the earthquake, i.e., hypocentre, and the surface location were the main earthquake manifests (epicentre), to predict the earthquake's magnitude [59]. Such a problem can be solved by employing static analysis. One of the tools of static analysis is programs that use networks of artificial neurons. To obtain a qualitative result of NN modeling, it is necessary to consider a sufficient number of examples and corresponding solutions for training NNs, ie establishing and memorizing patterns that link them together [59]. Such examples are the data of strong earthquakes (Mw>=4 on the Richter scale), recorded in Greece from 2000-2020 [60, 61]. For this type of forecasting, MLP, also called feed-forward multilayer NN, is used in more than 50% of cases for earthquake prediction based on structured earthquake catalogues [62]. Based on the number of studies using this type of NN architecture in earthquakes predictions with more accurate results, it was chosen for further research in this paper.

3.2 Hellenic Arc seismotectonic situation and seismic data for neural network analysis

Greece is located in the area between the African and Eurasian plates, which is very vulnerable to seismic activity and has historically often been affected by the subterranean elements. The Hellenic Seismic Arc begins from Corfu, descends to Kefalonia, Zakynthos, the coast of the Peloponnese, furthers below Crete and ends at Rhodes, where it descends even lower (*Figure 3*). There are nineteen (19) active tectonic faults located in the Greek vicinity [60, 61]. Every day, the Geodynamic Institute of the National Observatory of Athens, which monitors Greek seismicity and coordinates a national seismological network of more than 300 seismological stations across the country, normally records 20 to 30 imperceptible earthquakes. These earthquakes have a magnitude below Mw. 3.0 on the Richter scale.

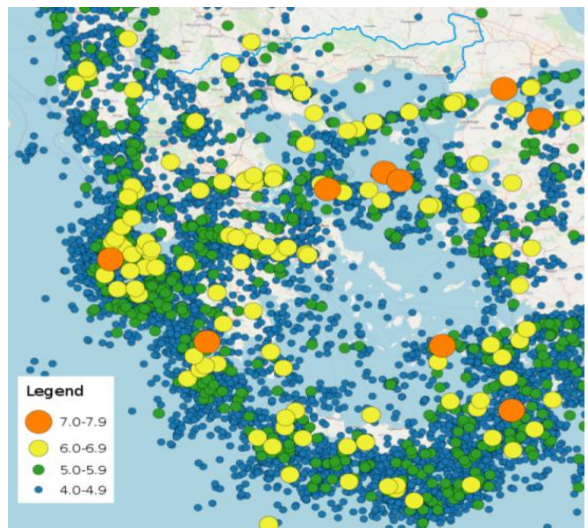

Figure 3 Approximate map of earthquakes in Greece in the period 1900-2020 (compiled by the author according to [16, 17])

The article presents the results of the development of a neural network model (NNM) for estimating occurrence numbers and estimated magnitudes of medium-strong earthquakes (Mw>=4 on the Richter scale). This threshold has been introduced to compensate for the artificially inflicted increment in the recorded small-sized earthquake that is due to the increase in the overall number and sensitivity of modern seismic recording stations [63–66]. The following data from the training sample for the period 2000-2015 (n = 2002 events) and the test sample for the period 2016-2020 (n = 437 events) are used to construct the forecast NN, importing to it information regarding earthquakes' geographical coordinates of latitude and longitude, depth and magnitude.

ANNs are used to approximate any complex functional connections. Using NN modeling, the type of relationship between input variables and output variables is not indicated, in contrast to generalized linear models, as confirmed by the McCullagh and Nelder study, where this is mandatory, for example, as a linear combination [67].

3.3 Customization of feed-forward neural network for earthquake prediction

Feed-forward networks in predicting short- and long-term earthquakes are primarily characterized by their static nature, which allows the weights of the NN to remain fixed after their determination and not change over time [63]. Feed-forward means that this type of NN can predict long-term and short-term earthquakes, but they cannot receive back propagation from multi-layer outputs and the BPNN, which is mostly in different local conditions during the training phase [52]. Thus, this type of network does not repeat iterations until the final solution is obtained, but directly converts input signals into output data regardless of the previous input data [68]. However, the probability of obtaining the desired result increases when the network is tested with perfectly represented input signals. The network adjusts the weights over many iterative cycles, honing its output to the most accurate value [52].

There are no established rules for determining the exact number of neurons in the hidden layer. Classically, the best configuration is determined from the input data by trial and error, starting with a small number of nodes [68]. However, Huang and Huang (1991) show that the upper bound on the number of neurons needed to accurately reproduce the desired results of the training samples is m training samples. Thus, the number of neurons in the hidden layer should never exceed the number of training samples. Furthermore, to keep the training problem bounded, the number of training samples should always be greater than the number of internal weights. In practice, $m \approx 10ntot$ (total number) is considered a good choice. Hence, the number of neurons must be limited; otherwise, the training set will simply be memorized by the network [69].

A two-layer FFNN was constructed for forecasting training. This was possible because of the large dataset comprised of data on earthquakes for a period of twenty (20) years instead of a comparable BPNN that would require fewer data [13, 70]. This NN architecture gave the best result in comparison to various other multilayer NNs tested using unknown testing data; meeting expectations as it is one of the most commonly used NN architectures in earthquake predictions along with neuro-fuzzy models [71–73]. Geographical coordinates of latitude and longitude of the earthquake, and depth of the earthquake are inputs values, and the output value is the magnitude on the Richter scale.

Before forecasting initiates, the available data are subjected to preliminary optimization for the correct operation of the NN including the removal of factor values, normalization of numerical data, and verification of available data for the appropriate format.

When defining a NN training algorithm, it seems that the best solution to determine when to stop training is the case when a local minimum is reached or when the convergence rate has become very small, i.e., the improvement from iteration to iteration is zero or minimal. However, Geman et al. show that this leads to overfitting, i.e., to memorization of the training set. Hence, the resulting weight distribution will be optimal for the training samples, but will lead to poor performance in general [74]. A similar phenomenon occurs in tomography problems where it is known as overfitting [75].

A classic solution to this dilemma is to use a partitioned set of examples. One part is used for training; the other part is used as a reference set to quantify the overall performance. Training is stopped when the discrepancy of the reference set reaches a minimum. This method is known as delay cross-validation [68].

Although this method generally gives good results, it results in a shorter training set, which can be a problem if only a limited number of examples are available. Since this method requires dividing the number of existing examples, the final number of training samples used is further reduced. The dataset we use in this paper consists of a sufficient number of observations, which allowed us to apply this particular training algorithm and obtain the desired result [68].

The final data set is divided into training and test sets in the ratio of 83% to 17% [70]. The control set shall be created from the training set automatically during the training of the model and is 10% of the training data set. The sampling is based on R values, which define the ratio between the real and the expected results. The closer the R value is to 1, the more accurate the results are [76]. To avoid the problem of split datasets, there are some other methods: generalized cross-validation methods, residual analysis and theoretical measures that investigate both the results obtained and the complexity of the network.

In the process of functioning, the NN generates the output Y for a given input X via the function Y = G(X). With the known network architecture, the type of function G is determined by the values of synaptic weights (coefficients) and offsets (threshold signals) of the network [77].

The simplest multi-layer perceptron consists of an input layer with n covariates and an output layer with one output neuron. It can be described by the following Equation 1:

$$o(x) = f(w_0 + \sum_{i=1}^{n} w_i x_i) = f(w_0 + w^T x), \quad (1)$$

where w_0 is the threshold value; $w=(w_1 ,..., w_n)$ is the vector consisting of all synaptic weights without taking into account the threshold value; $x=(x_1 , ..., x_n)$ is the vector of all input parameters [77].

In this case, all calculated weights are equivalent to the parameters of the general linear regression model (GLM) [77].

However, K.Hornik, M.Stichcombe, and H.White refuted the idea that one hidden layer is sufficient to model any partially continuous function [77]. Multi-layer perceptron, consisting of J hidden neurons, is calculated by the following Equation 2:

$$o(x) = f(w_0 + \sum_{j=1}^{J} w_j \cdot f(w_{0j} + \sum_{i=1}^{n} w_{ij} x_i)) =$$
$$= f(w_0 + \sum_{j=1}^{J} w_j \cdot f(w_{0j} + w_j^T x)), \quad (2)$$

where w_0 is the threshold value; w_{0j} is the threshold value of jth hidden neuron; w_j is the synaptic weight, which corresponds to the synapse, which begins with the jth hidden neuron to the original neuron; $w_j=(w_1,..., w_{nj})$ is the vector consisting of all synaptic weights that correspond to the synapses leading to the jth hidden neuron; $x=(x_1,..., x_n)$ is the vector of all input covariates [77].

Assume that the solution is the function Y=F(X), given by pairs of data $(X_1 , B_1), (X_2 , B_2),...,(X_N ,Y_N)$, for which $Y_k = F(X_k)$ (k = 1, 2,...,N) [27]. The purpose of NN training is the synthesis of the function G, which will be close to F with the corresponding error of approximation to the data of the training sample E. The training of the NN becomes a multidimensional optimization of a large dimension.

The total error E as the difference of the square between the predicted and observed results is calculated as Equation 3:

$$E = \frac{1}{2}\sum_{l=1}^{L} \sum_{h=1}^{H} (o_{lh} - y_{lh})^2 \quad (3)$$

where l = 1,…,L is the number of input-output pairs; h = 1,…, H - number of outputs; o_{lh} and y_{lh} - predicted and observed outputs, respectively [77].

The mean absolute percentage error (MAPE) is the arithmetic mean of the absolute errors $|e_i| = o_{lh} - y_{lh}$, which is calculated as follows Equation 4:

$$MAPE = \frac{\sum_{l=1}^{L}\sum_{h=1}^{H}|o_{lh}-y_{lh}|}{L\times H} = \frac{\sum_{l=1}^{L}\sum_{h=1}^{H}|e_i|}{L\times H} \quad (4)$$

Equation 1 was used in creating a NN with one hidden layer for forecasting the periodicity of strong earthquakes based on the monthly data of earthquakes that have occurred in a period of twenty (20) years in Greece. Equations 2 and 3 were used to create and estimate the performance of a two-layer FFNN for forecasting the maximum magnitude of earthquakes monthly. Equation 4 calculates a measure of the prediction accuracy of forecasting methods of earthquakes to identify the better model.

4.Results

4.1Predicting the frequency of strong earthquakes

Building a NN to predict medium-strong earthquakes is implemented in the development environment RStudio programming language R, which is ideal for statistical data processing. Neuralnet package was used to build the required NN, which contains a very flexible function training FFNNs and allows you to model many internal hidden layers and hidden network neurons. We also used nnet package, which is a fairly simple universal tool for building predictive models in NN programming.

To analyze the level of seismicity in Greece, research on the frequency of strong earthquakes (M>=4) was conducted using the nnet package. The results of this research are displayed using a linear graph. For this purpose, data were used that characterize the monthly number of strong earthquakes that occurred in Greece in the period 2000-2020 (n = 252 events) (*Figure 4*). Using the nnet library for this analysis should take into account its feature - it allows you to build NNs with only one hidden layer.

The following algorithm was used to obtain the result of predicting the frequency of strong earthquakes (*Figure 5*), The final graph of the linear dependence for input data was obtained using the algorithm above in the RStudio development environment (*Figure 6*):

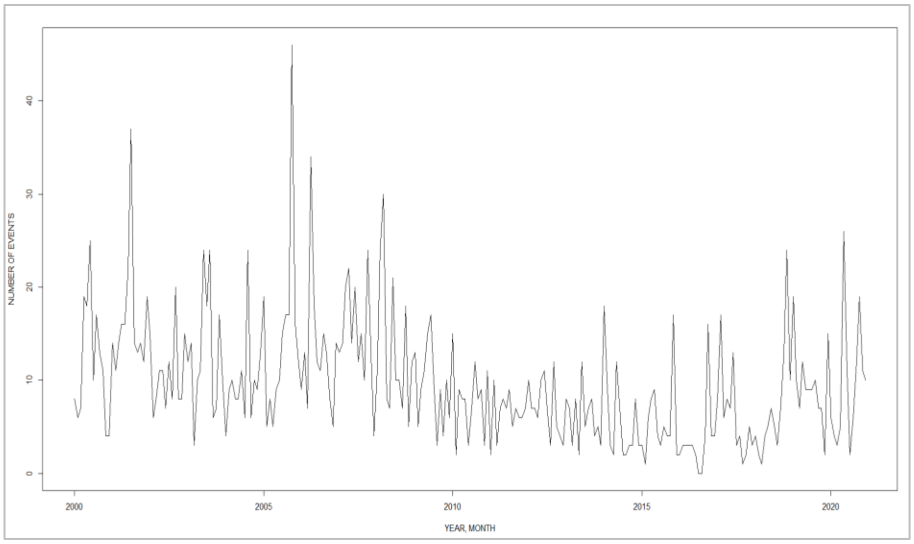

Figure 4 Periodicity of strong earthquakes in Greece for the period 2000-2020 (n = 252 events)

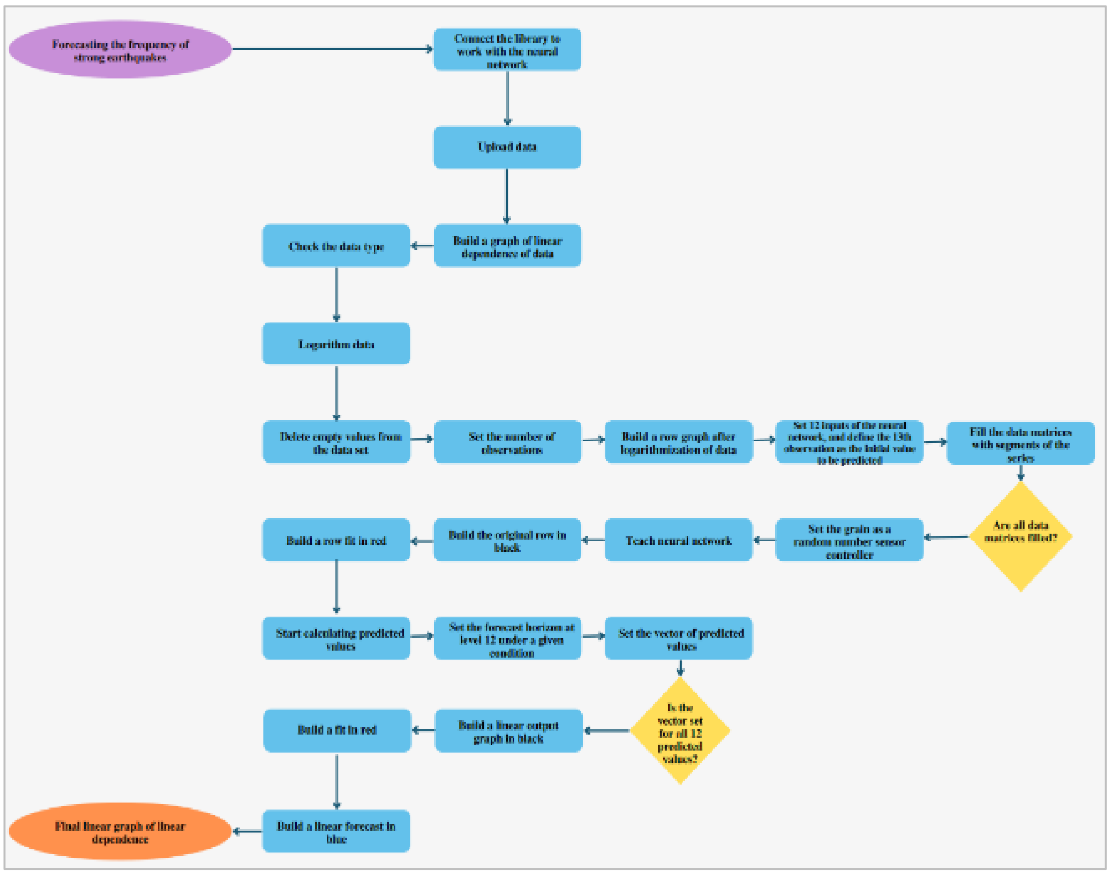

Figure 5 Algorithm for predicting the frequency of strong earthquakes [*Appendix I*]

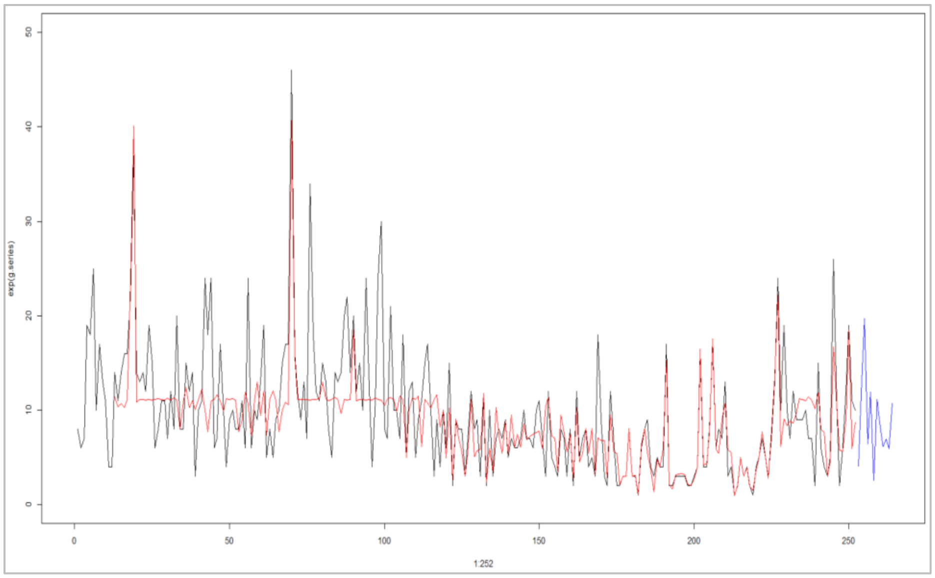

Figure 6 The final graph of linear dependence: black shows the original series, red - fit, blue - forecast for n = 252 events for the period 2000-2020

A data set consisting of 240 random events of strong earthquakes (M_w>=4 on the Richter scale) in the period 2000-2020 was used to train the NN. The test data set included 12 events, which is approximately 5% of the total sample. The forecast is made for 1 observation ahead.

4.2 Neural network modeling

The neuralnet function used to train the NN makes it possible to determine the required number of hidden layers and hidden neurons according to the desired complexity of the NNM. The complexity of the calculated function increases with the addition of hidden layers of hidden neurons. The default value is one hidden layer with one hidden neuron. The output of the NN directly depends on the activation function *f* (Equation 5):

$$f(u) = \frac{1}{1+e^{-u}}, \quad (5)$$

where u means the weighted sum of the outputs of all hidden layers [77].

Unlike FFNNs, for BPNNs it is advisable to use nonlinear sigmoid functions as an activation function a hyperbolic tangent, or a logistic function [71].

The logistic function, for example, is suitable for binary variables because it reflects the output of each neuron in the interval [0, 1].

The NN activation function for our data set is: NN = neuralnet (MAGNITUDE ~ LONG + LAT + DEPTH, trainNN, hidden = c (5.3), err.fct = "sse", linear.output = F) according to Equations 5. The NN receives data of three independent variables, i.e., geographical latitude coordinates, longitude geographic coordinates, and earthquake depth, which pass through two hidden layers of the network, consisting of five and three hidden neurons, respectively [70,71,77] Input data propagate through the above NN architecture producing a crisp output of the network. The NN was obtained to predict strong earthquakes (M_w>=4 on the Richter scale) in Greece using the following algorithm in the RStudio development environment (*Figure 7*).

The following statements are relevant using the algorithm in *Appendix II.*

- the grain "50" as a random number controller is not the stable number; we can change this value and see how the NN behaves;
- normalizing the training (83% of all data) and test (17% of all data) data, take into account that the data must be in the interval [0;1] ("2" in the function "apply" indicates columns used for analysis; when "1" indicates rows that is not appropriate in our case
- prescribing the appropriate NN activation function, set magnitude as dependent variable and longitude, latitude and depth as covariates, and set also a differentiable function "err.fct" that is used for the calculation of the error and "linear.output=false", so that input network value are within [0;1].

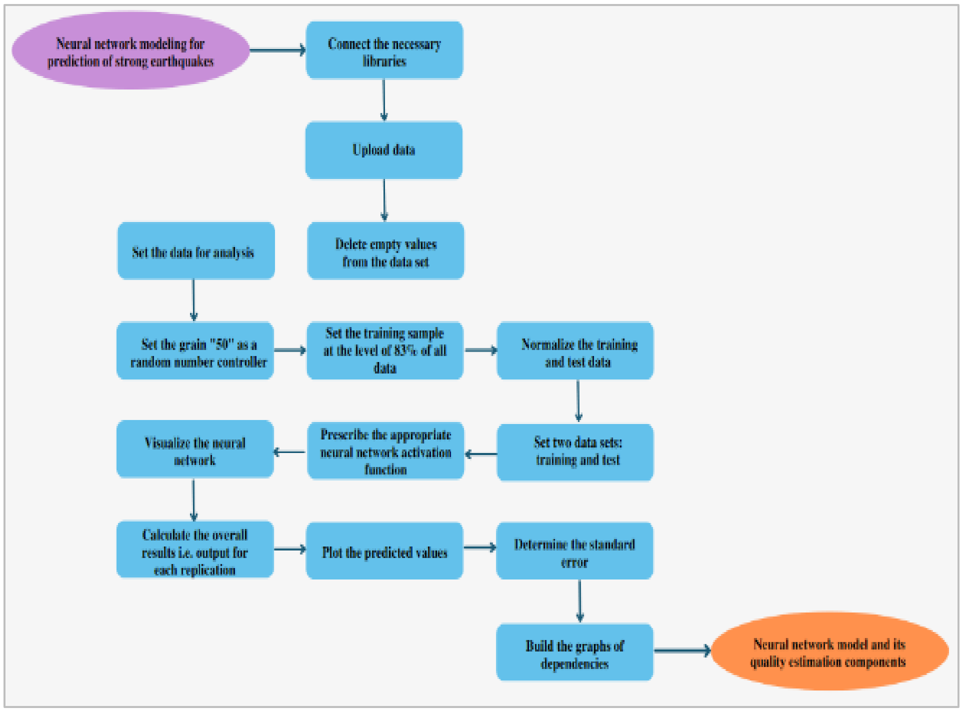

Figure 7 Algorithm of NN modeling in the problem of forecasting strong earthquakes [*Appendix II*]

To build the final forecast, more than 50 NNs were tested with various parameters: the number of hidden layers, the size of the training sample, the NN regularization parameter. The random search method was used in the RStudio development environment for the automatic calculation of these parameters [Appendix II]. The main feature of this method is that it does not specify the optimum search program in advance. However, automatic calculation also has a drawback - the speed of learning the model decreases [71]. Additional training were conducted based on the steps described above. Then MAPE was calculated to test the training accuracy of the NNM depending on the parameters that were calculated by the random search method. Based on this, a table was created with the results of the study for 15 selected NNMs with the most relevant results (*Table 1*).

The final forecast was built based on the algorithm above and the characteristics of the selected trained network that obtained the best MAPE results - MAPE=12.07 (*Figure 8*).

In the NN learning process, 6582 steps were performed until all absolute partial derivatives of the error function became close to 0.01 (default threshold). The relative error of the network calculated by Equation 3 is E = 16.82, which indicates fairly successful learning of the NN and the possibility of obtaining highly reliable results compared to other NN models [71]. From the graph and table above, it follows that the quality of the trained model depends primarily on the number of learning cycles and the number of hidden layers. Based on the results, we can say that the impact on the quality of the model also has the size of selected data for its training. The corresponding synaptic weights of connections between neurons of the constructed 2-layer NN are presented in *Figure 9.*

Table 1 Dependence of training quality on input parameters

Number of hidden layers	**Number of training steps**	**Absolute partial derivatives of the error**	**Training data set (% of all data)**	**MAPE**	**Total error (E)**
2	162	10	80%	56.2	89.7
2	2890	10	85%	34.0	78.4
2	6582	0.01	83%	12.07	16.82
3	1154	0.1	85%	27.2	38.0
3	3876	0.001	80%	19.89	23.4

Number of hidden layers	Number of training steps	Absolute partial derivatives of the error	Training data set (% of all data)	MAPE	Total error (E)
3	18	10	90%	94.7	99.8
5	25	100	80%	97.0	99.9
5	456	10	85%	91.0	94.5
12	356	100	80%	93.8	96.1
12	2401	0.1	85%	44.0	52.1
15	1090	0.001	80%	47.9	55.0
15	880	0.0001	85%	34.9	45.8
25	256	10	80%	77.8	89.2
25	480	10	85%	75.0	81.9
50	915	10	80%	80.3	85.0

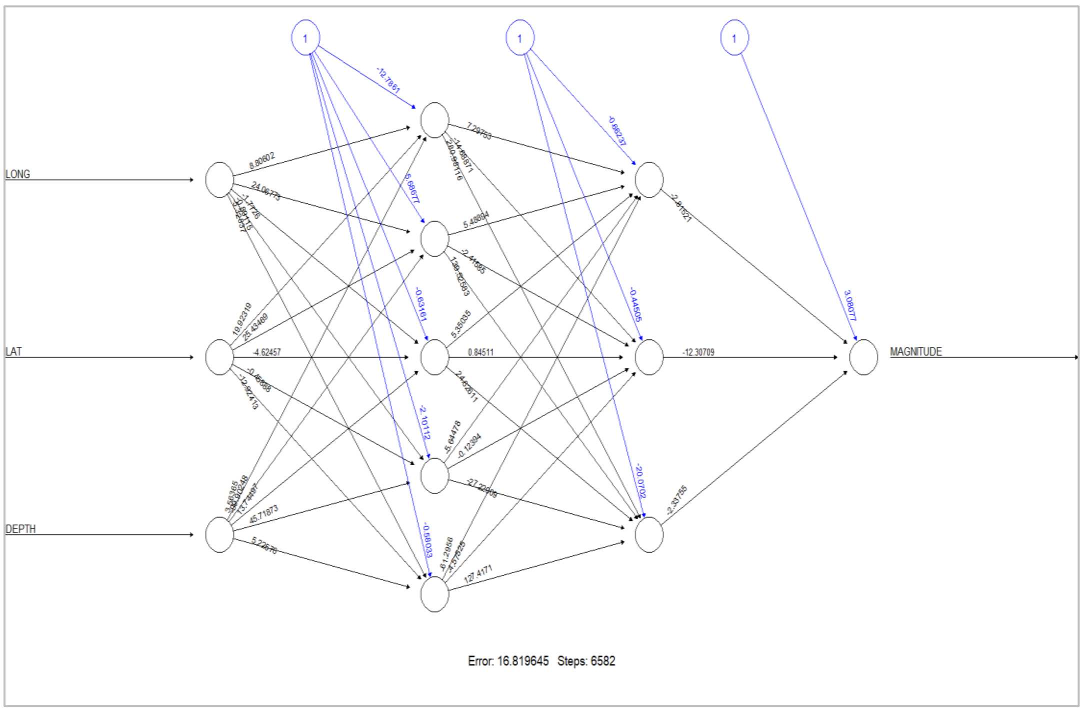

Figure 8 NN for predicting strong earthquakes (M_w>=4 on the Richter scale) in Greece

Synaptic weights characterize the strength of the connection between neurons and can be used to detect the effect of each covariate on the dependent variable - the magnitude of the earthquake.

For instance, from *Figure 6* any neuron from the first constant is connected to each of the five hidden neurons of the first hidden layer with the following inputs: -12.79, 5.69, -0.63, -2.1, -0.58, which are described by the first line of *Figure 9.* The second line describes outputs from the first independent variable - longitude, as inputs to each of the five hidden neurons of the first hidden layer, etc. Negative synaptic weights mean an inversely proportional relationship between covariates and the dependent variable.

```
> NN$weights
[[1]]
[[1]][[1]]
           [,1]       [,2]       [,3]       [,4]        [,5]
[1,] -12.786097   5.686773 -0.6316072 -2.1011217  -0.5803304
[2,]   8.806019  24.067733 -1.7726017 -0.8911510  -5.3283750
[3,]  19.923185  25.434689 -4.6245705 -0.4588819 -12.9241263
[4,]   3.563651 -99.902484 13.7449704 45.7187307   5.2257554

[[1]][[2]]
            [,1]        [,2]      [,3]
[1,]  -0.6623743  -0.4450467 -20.07020
[2,]   7.2975288 -14.6887111 280.98116
[3,]   5.4889394  -2.4158505 139.52583
[4,]   5.3503534   0.8451080  24.62611
[5,]  -5.6447833  -0.1239352 -27.22009
[6,] -61.2956019  -4.5752466 127.41710

[[1]][[3]]
           [,1]
[1,]   3.080774
[2,]  -2.815211
[3,] -12.307093
[4,]  -2.337551
```

Figure 9 Synaptic weights of connections between neurons of the constructed 2-layer NN

5.Discussion

Assessing the impact of each of the 3 independent variables on the dependent variable - the magnitude of the earthquake, visualize the data, building the following graphs of dependencies (*Figure 10*) [*Appendix II*]. Generalized weights are given for all independent variables within one range. The distribution of generalized weights allows us to conclude that all independent variables have a nonlinear effect because their variance generalized weight is generally more than one, which is why we believe that each independent variable has a significant impact on the initial result.

The last step in the NN analysis is to plot the predicted values and determine the standard error of the model (*Figure 11*) [*Appendix II*].

With the help of the constructed graph (*Figure 11*), we can estimate the accuracy of the constructed forecast and trace the correspondence of the constructed forecast values to the actual data of the sample, especially in the range M_w = 4.0-4.5. The root mean square error of the model is measured at 3.44, which indicates a fairly good quality of the predicted model [71], ie a good level of data integration, as there is a small difference between predicted and observed values compared to the error of the training model (E = 16.82). The difference between actual and forecast values is reasonable (*Table 2*).

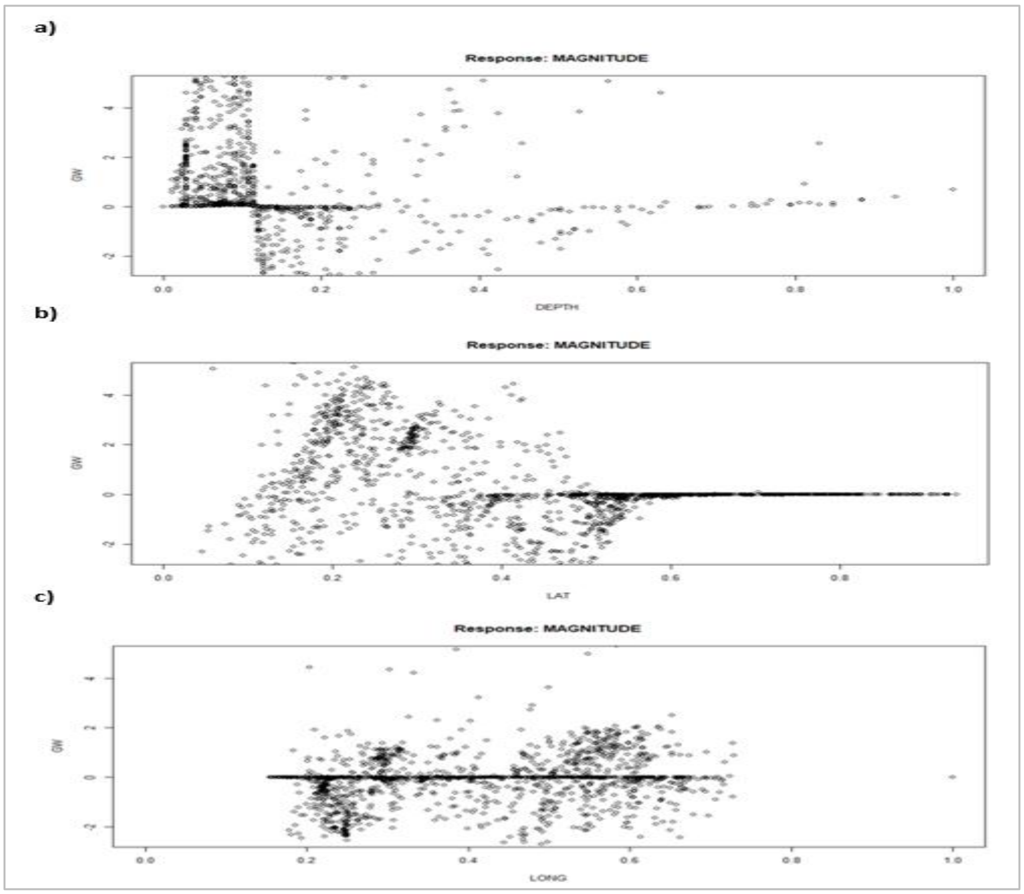

Figure 10 Graphs of generalized weights for the independent variable of depth (a), the independent variable of latitude (b), and the independent variable of longitude (c)

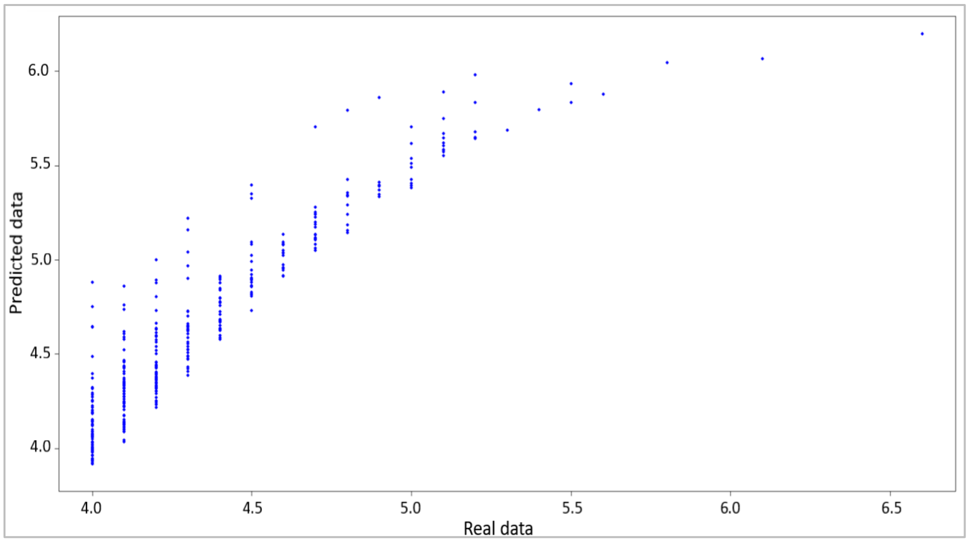

Figure 11 Graph of prediction of strong earthquakes (Mw>=4 on the Richter scale) in Greece, built on the basis of the developed NN, for the study of which used data from the training sample for the period 2000-2015 (n = 2002 events) and n = 437 events for the period 2016-2020 were presented as a test data set

Table 2 Actual values and obtained prediction data with MLP neural network

Magnitude	**2000-2004**	**2005-2009**	**2010-2014**	**2015-2020**
M_w (actual)	5,5	5,0	4,6	4,2
M_w (forecast)	5,8	5,4	5,1	4,3

The calculations of *Table 2* are obtained from the ratio of real data to the percentage that characterizes the probability of obtaining reliable forecasted data [*Appendix II*].

As a result, an optimal neural network MLP (or feed-forward multilayer NN) was obtained with four layers: input (3 neurons), the first hidden layer (5 neurons), the second hidden layer (3 neurons) and output (1 neuron) for forecasting earthquakes in Greece with the value R = 0,72. A complete list of abbreviations is shown in *Appendix III.*

6.Conclusion

Fundamental studies of seismicity and seismic hazard forecasting are continuous, as observation systems are improved, qualitatively new information is received, and new mathematical and computational models of seismicity and design of seismic structures are developed. The strategy of seismic construction and seismic risk reduction requires constant improvement of methods of calculation and compilation of increasingly informative normative forecast maps of seismic hazard. Seismic hazards cannot be reduced, the acceptable seismic risk of natural disasters can be reduced by having information on the location of hazards, the location of natural disasters, and their consequences.

The assessment of seismic hazard, which is predictive in nature, depends on several parameters of the earthquake focus, which were used in this study, i.e., the geographical coordinates of the epicenter, the depth of the source, and magnitude. Using NN modeling to predict the occurrence of earthquakes in Greece, it was possible to calculate the magnitude of future earthquakes. A methodology has been developed to calculate seismic hazards in settlements located within the radius of influence of a group of seismic zones in Greece, and to build a map of seismic zoning of large areas, for example, for the whole of Europe using a spatiotemporal clustering algorithm [78, 79]. This could be the next step in identifying and exploring new potentially different seismic zones [80].

The key point in improving the accuracy of the built NN may be to expand the data sample during model training and increase/decrease the number of hidden layers and neurons, as the final graph predicts the occurrence of medium-strong earthquakes (Mw>=4) in Greece the accuracy of the model is lost with increasing magnitude of the earthquake. This may be due to insufficient data on the magnitude of the earthquake, which can be easily corrected by expanding the data sample.

A limitation of using this approach is that it is not possible to predict the exact date and time of the seismic event, because the data type is not able to convert to the numerical type for modeling an ANN. At the present stage of research, American developers have managed to introduce a NN called UrbanDenoiser, which was taught to filter anthropogenic noise in the city from seismic signals signaling an earthquake. However, given the data set for training, the NN can only work in California [81]. Therefore, the main goal of improving such devices is to adapt them to the specifics of any region, but this requires more than one year of research and expansion of impact factors specific to the characteristics [82] of different seismic areas.

Acknowledgment
None.

Conflicts of interest
The authors have no conflicts of interest to declare.

Author's contribution statement
Dariia Voloshchuk, Antonios J. Konstantaras, Alexandra Moshou: Conceptualization, investigation, data curation, writing original draft, writing, and editing. **Nataliia Kasianova, Irina Skorniakova, Panagiotis Argyrakis and Nikolaos S. Petrakis:** Study conception, design, supervision, manuscript review and revision, and final decision.

References
[1] http://natcatservice.munichre.com/. Accessed 20 May 2022.
[2] Musson RM. Intensity-based seismic risk assessment. Soil Dynamics and Earthquake Engineering. 2000; 20(5-8):353-60.
[3] http://static.seismo.ethz.ch/GSHAP/eu-af-me/euraf.html. Accessed 15 May 2022.
[4] Goldberg DE, Melgar D, Hayes GP, Crowell BW, Sahakian VJ. A ground- motion model for GNSS peak ground displacement. Bulletin of the Seismological Society of America. 2021; 111(5):2393-407.
[5] Yeats RS. Living with earthquakes in the Pacific Northwest: Chapter 9: Tsunami!.
[6] Huang F, Li M, Ma Y, Han Y, Tian L, Yan W, Li X. Studies on earthquake precursors in China: A review for recent 50 years. Geodesy and Geodynamics. 2017; 8(1):1-2.
[7] Mubarak MA, Riaz MS, Awais M, Jilani Z, Ahmad N, Irfan M, et al. Earthquake prediction: a global review and local research. Proceedings of the Pakistan Academy of Science. 2009; 46(4):233-46.
[8] Gitis V, Derendyaev A, Petrov K. Analyzing the performance of GPS Data for earthquake prediction. Remote Sensing. 2021; 13(9).
[9] https://science.nasa.gov/technology/technology-highlights/new-radar-to-monitor-volcanoes-and-earthquakes-from-space. Accessed 13 September 2022.
[10] Alvan HV, Azad FH. Satellite remote sensing in earthquake prediction. a review. In national postgraduate conference 2011 (pp. 1-5). IEEE.
[11] https://medium.com/intuitionmachine/deep-learning-and-semiotics-b9bb16045005. Accessed 20 May 2022.
[12] https://blogs.microsoft.com/on-the-issues/2017/12/11/ai-for-earth-can-be-a-game-changer-for-our-planet/. Accessed 15 May 2022.
[13] Al Banna MH, Taher KA, Kaiser MS, Mahmud M, Rahman MS, Hosen AS, et al. Application of artificial intelligence in predicting earthquakes: state-of-the-art and future challenges. IEEE Access. 2020; 8:192880-923.
[14] Maravelakis E, Konstantaras A, Kabassi K, Chrysakis I, Georgis C, Axaridou A. 3DSYSTEK web-based point cloud viewer. In IISA 2014, the 5th international conference on information, intelligence, systems and applications 2014 (pp. 262-6). IEEE.
[15] Axaridou A, Chrysakis I, Georgis C, Theodoridou M, Doerr M, Konstantaras A, Maravelakis E. 3D-SYSTEK: recording and exploiting the production workflow of 3D-models in Cultural Heritage. In IISA 2014, The 5th international conference on information, intelligence, systems and applications 2014 (pp. 51-6). IEEE.
[16] Amit DJ, Tsodyks MV. Effective neurons and attractor neural networks in cortical environment. Network: Computation in Neural Systems. 1992; 3(2).
[17] Derbal I, Bourahla N, Mebarki A, Bahar R. Neural network-based prediction of ground time history responses. European Journal of Environmental and Civil Engineering. 2020; 24(1):123-40.
[18] Moustra M, Avraamides M, Christodoulou C. Artificial neural networks for earthquake prediction using time series magnitude data or seismic electric signals. Expert systems with applications. 2011; 38(12):15032-9.
[19] Zhou H, Li J, Chen X. Establishment of a seismic topographic effect prediction model in the Lushan M s 7.0 earthquake area. Geophysical Journal International. 2020; 221(1):273-88.
[20] Ghorbani S, Barari M, Hoseini M. Presenting a new method to improve the detection of micro-seismic events. Environmental Monitoring and Assessment. 2018; 190(8):1-3.
[21] Liu Q, Sun P, Fu X, Zhang J, Yang H, Gao H, Li Y. Comparative analysis of BP neural network and RBF neural network in seismic performance evaluation of pier columns. Mechanical Systems and Signal Processing. 2020.
[22] Xiaoshu G, Cheng C, Hetao H, Shenghu J. Response prediction using the PC-NARX model for SDOF systems with degradation and parametric uncertainties. Earthquake Engineering and Engineering Vibration. 2022; 21(2):325-40.

[23] Ye T, Shu T, Li J, Zhao P, Wang Y. Study on dynamic stability prediction model of slope in eastern Tibet section of Sichuan-Tibet highway. Mathematical Problems in Engineering. 2022.

[24] Huang Y, Han X, Zhao L. Recurrent neural networks for complicated seismic dynamic response prediction of a slope system. Engineering Geology. 2021.

[25] Datta A, Wu DJ, Zhu W, Cai M, Ellsworth WL. DeepShake: shaking intensity prediction using deep spatiotemporal RNNs for earthquake early warning. Seismological Society of America. 2022; 93(3):1636-49.

[26] Fayaz J, Xiang Y, Zareian F. Generalized ground motion prediction model using hybrid recurrent neural network. Earthquake Engineering & Structural Dynamics. 2021; 50(6):1539-61.

[27] Oh BK, Park Y, Park HS. Seismic response prediction method for building structures using convolutional neural network. Structural Control and Health Monitoring. 2020; 27(5).

[28] Zhang R, Liu Y, Sun H. Physics-guided convolutional neural network (PhyCNN) for data-driven seismic response modeling. Engineering Structures. 2020.

[29] Zhang X, Zhang J, Yuan C, Liu S, Chen Z, Li W. Locating induced earthquakes with a network of seismic stations in Oklahoma via a deep learning method. Scientific reports. 2020; 10(1):1-12.

[30] Linville LM. Event- based training in label- limited regimes. Journal of Geophysical Research: Solid Earth. 2022; 127(1).

[31] Kim T, Song J, Kwon OS. Probabilistic evaluation of seismic responses using deep learning method. Structural Safety. 2020.

[32] Jain R, Nayyar A, Arora S, Gupta A. A comprehensive analysis and prediction of earthquake magnitude based on position and depth parameters using machine and deep learning models. Multimedia Tools and Applications. 2021; 80(18):28419-38.

[33] Sreejaya KP, Basu J, Raghukanth ST, Srinagesh D. Prediction of ground motion intensity measures using an artificial neural network. Pure and Applied Geophysics. 2021; 178(6):2025-58.

[34] Ida Y, Ishida M. Analysis of seismic activity using self-organizing map: implications for earthquake prediction. Pure and Applied Geophysics. 2022; 179(1):1-9.

[35] Kail R, Burnaev E, Zaytsev A. Recurrent convolutional neural networks help to predict location of earthquakes. IEEE Geoscience and Remote Sensing Letters. 2021; 19:1-5.

[36] Akram J, Ovcharenko O, Peter D. A robust neural network-based approach for microseismic event detection. In SEG technical program expanded abstracts 2017 (pp. 2929-33). Society of Exploration Geophysicists.

[37] Rundle JB, Stein S, Donnellan A, Turcotte DL, Klein W, Saylor C. The complex dynamics of earthquake fault systems: new approaches to forecasting and nowcasting of earthquakes. Reports on Progress in Physics. 2021; 84(7).

[38] Lior I, Ziv A. Generic source parameter determination and ground- motion prediction for earthquake early warninggeneric source parameter determination and ground- motion prediction for earthquake early warning. Bulletin of the Seismological Society of America. 2020; 110(1):345-56.

[39] Chong KD. Research and application of neural network technologies in the problem of earthquake forecasting: (on the example of the northwestern region of Vietnam). Dissertations, 2013. Moskow, RUDN University.

[40] Nicolis O, Plaza F, Salas R. Prediction of intensity and location of seismic events using deep learning. Spatial Statistics. 2021.

[41] Segou M, Parsons T. A new technique to calculate earthquake stress transfer and to probe the physics of aftershocks. Bulletin of the Seismological Society of America. 2020; 110(2):863-73.

[42] Holyoak KJ. Parallel distributed processing: explorations in the microstructure of cognition. Science. 1987; 236:992-7.

[43] Rumelhart D, Mcclelland J, PDP Research Group. parallel distributed processing: explorations in the microstructures of cognition. MIT Press; Cambridge MA. 1986.

[44] Su YP, Zhu QJ. Application of ANN to prediction of earthquake influence. In second international conference on information and computing science 2009 (pp. 234-7). IEEE.

[45] Itai A, Yasukawa H, Takumi I, Hata M. Multi-layer neural network for precursor signal detection in electromagnetic wave observation applied to great earthquake prediction. In IEEE-Eurasip nonlinear signal and image processing 2005.

[46] Shahi S, Fenton FH, Cherry EM. Prediction of chaotic time series using recurrent neural networks and reservoir computing techniques: a comparative study. Machine Learning with Applications. 2022.

[47] Adeli H, Panakkat A. A probabilistic neural network for earthquake magnitude prediction. Neural Networks. 2009; 22(7):1018-24.

[48] Panakkat A, Adeli H. Recurrent neural network for approximate earthquake time and location prediction using multiple seismicity indicators. Computer- Aided Civil and Infrastructure Engineering. 2009; 24(4):280-92.

[49] Lakshmi SS, Tiwari RK. Model dissection from earthquake time series: a comparative analysis using modern non-linear forecasting and artificial neural network approaches. Computers & Geosciences. 2009; 35(2):191-204.

[50] Negarestani A, Setayeshi S, Ghannadi-maragheh M, Akashe B. Layered neural networks based analysis of radon concentration and environmental parameters in earthquake prediction. Journal of Environmental Radioactivity. 2002; 62(3):225-33.

[51] Ozerdem MS, Ustundag B, Demirer RM. Self-organized maps based neural networks for detection of possible earthquake precursory electric field patterns.

Advances in Engineering Software. 2006; 37(4):207-17.
[52] Voloshchuk D, Kasianova N. Machine learning as an earthquake prediction tool: current and future challenges for the global economy. International scientific-practical conference “economic and business administration development: scientific currencies and solutions” 2022 (pp. 26-8), Kyiv, NAU
[53] Krinitzsky EL. How to combine deterministic and probabilistic methods for assessing earthquake hazards. Engineering Geology. 2003; 70(1-2):157-63.
[54] Zechar JD, Hardebeck JL, Michael AJ, Naylor M, Steacy S, Wiemer S, et al. Community online resource for statistical seismicity analysis. Seismological Research Letters. 2011; 82(5):686-90.
[55] Stabile TA, Serlenga V, Satriano C, Romanelli M, Gueguen E, Gallipoli MR, et al. The INSIEME seismic network: a research infrastructure for studying induced seismicity in the High Agri Valley (southern Italy). Earth System Science Data. 2020; 12(1):519-38.
[56] Zaliapin I, Ben-zion Y. Artefacts of earthquake location errors and short-term incompleteness on seismicity clusters in southern California. Geophysical Journal International. 2015; 202(3):1949-68.
[57] Abbott LF. Learning in neural network memories. Network: Computation in Neural Systems. 1990; 1(1).
[58] León SM, Calviño BO, Vivas LA, Corretger RC, Ulacio OR. Small-layered feed-forward and convolutional neural networks for efficient P wave earthquake detection. Expert Systems with Applications. 2022.
[59] Byrne JH, Berry WO. Neural models of plasticity: experimental and theoretical approaches. Academic Press; 2013.
[60] http://dggsl.geol.uoa.gr/en_index.html. Accessed 20 May 2022.
[61] https://www.gein.noa.gr/services/cat.html. Accessed 20 May 2022.
[62] Mignan A, Broccardo M. Neural network applications in earthquake prediction (1994–2019): meta- analytic and statistical insights on their limitations. Seismological Research Letters. 2020; 91(4):2330-42.
[63] Bodri B. A neural-network model for earthquake occurrence. Journal of Geodynamics. 2001; 32(3):289-310.
[64] Konstantaras A, Vallianatos F, Varley MR, Makris JP. Soft-computing modelling of seismicity in the southern Hellenic Arc. IEEE Geoscience and Remote Sensing Letters. 2008; 5(3):323-7.
[65] Konstantaras AJ. Classification of distinct seismic regions and regional temporal modelling of seismicity in the vicinity of the Hellenic seismic Arc. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing. 2012; 6(4):1857-63.
[66] Konstantaras A, Petrakis NS, Frantzeskakis T, Markoulakis E, Kabassi K, Vardiambasis IO, et al. Deep learning neural network seismic big-data analysis of earthquake correlations in distinct seismic regions. International Journal of Advanced Technology and Engineering Exploration. 2021; 8(84):1410-23.
[67] http://www.utstat.toronto.edu/~brunner/oldclass/2201s11/readings/glmbook.pdf. Accessed 15 May 2022.
[68] Van DBM, Jutten C. Neural networks in geophysical applications. Geophysics. 2000; 65(4):1032-47.
[69] Huang SC, Huang YF. Bounds on number of hidden neurons of multilayer perceptrons in classification and recognition. In international symposium on circuits and systems 1990 (pp. 2500-3). IEEE.
[70] https://www.v7labs.com/blog/train-validation-test-set. Accessed 17 May 2022.
[71] Hornik K, Stinchcombe M, White H. Multilayer feedforward networks are universal approximators. Neural Networks. 1989; 2(5):359-66.
[72] Konstantaras A, Varley MR, Vallianatos F, Collins G, Holifield P. Neuro-fuzzy prediction-based adaptive filtering applied to severely distorted magnetic field recordings. IEEE Geoscience and Remote Sensing Letters. 2006; 3(4):439-41.
[73] Konstantaras A, Varley MR, Vallianatos F, Makris JP, Collins G, Holifield P. Detection of weak seismo-electric signals upon the recordings of the electrotelluric field by means of neuro-fuzzy technology. IEEE Geoscience and Remote Sensing Letters. 2007; 4(1):161-5.
[74] Geman S, Bienenstock E, Doursat R. Neural networks and the bias/variance dilemma. Neural Computation. 1992; 4(1):1-58.
[75] Scales JA, Snieder R. What is a wave? Nature. 1999; 401(6755):739-40.
[76] Hai DT, Vinh ND, Trieu CD. Long-term earthquake prediction in Lai Chau-Dien Bien region on the basis of time-magnitude model. Journal of Science and Technology. 2002; 40(4):45-53.
[77] Günther F, Fritsch S. Neuralnet: training of neural networks. The R Journal. 2010; 2(1):30-8.
[78] Konstantaras AJ, Katsifarakis E, Maravelakis E, Skounakis E, Kokkinos E, Karapidakis E. Intelligent spatial-clustering of seismicity in the vicinity of the Hellenic seismic Arc. Earth Science Research. 2012; 1(2):1-10.
[79] Konstantaras A. Deep learning and parallel processing spatio-temporal clustering unveil new Ionian distinct seismic zone. Informatics. 2020; 7(4):1-10.
[80] Konstantaras AJ. Expert knowledge-based algorithm for the dynamic discrimination of interactive natural clusters. Earth Science Informatics. 2016; 9(1):95-100.
[81] Yang L, Liu X, Zhu W, Zhao L, Beroza GC. Toward improved urban earthquake monitoring through deep-learning-based noise suppression. Science Advances. 2022; 8(15):1-9.
[82] Moshou A, Argyrakis P, Konstantaras A, Daverona AC, Sagias NC. Characteristics of recent aftershocks sequences (2014, 2015, 2018) derived from new seismological and geodetic data on the Ionian islands, Greece. Data. 2021; 6(2):1-27.

Dariia Voloshchuk was born in Kamianets-Podilskyi, Khmelnytskyi region, Ukraine. She is currently studying for a Master's degree at the Department of Business Analytics and Digital Economy at the Faculty of Economics and Business Administration at the National Aviation University (Kyiv, Ukraine) and as an Erasmus+ exchange student at the Department of Electronic Engineering at the Hellenic Mediterranean University (Crete, Greece). She also was an Erasmus+ exchange student at the Faculty of Economics and Business at the University of Cádiz, Spain.
After graduating with an Honours Bachelor`s Degree at the National Aviation University, she continues pursuing scientific, social and sports activities at the university. She actively participates in various scientific conferences highlighting methods and ways to overcome different economic issues and threats inherent in modern society both locally and globally. In parallel, she deeply explores the modern IT environment in many different ways, taking to attention to its special role and significance of mankind life, especially, during and after a pandemic. She has explored artificial intelligence since the third year of study and she is an author of more than 25 publications on economic and IT topics. Her Bachelor`s thesis on the topic "Modeling of information support of artificial intelligence systems" describes a lot of aspects of artificial intelligence. It was the best thesis at the Faculty. Also, she is the best student of the Faculty and the recipient of an Academic Scholarship of the President of Ukraine.
Email: dasha1voloshchuk@gmail.com

Dr. Antonios J. Konstantaras is an Associate Professor in Software Engineering at the Hellenic Mediterranean Univerity (HMU). He received his doctorate from the University of Central Lancashire, U.K. in 2004 in Soft Computing. He has published over 25 journal articles in various fields of software engineering and serves as editor in 2 scientific journals.
Email: akonstantaras@hmu.gr

Alexandra Moshou holds a BSc degree in Mathematics from the University of Crete and a MSc in Applied Mathematics from National and Kapodistrian Univeristy of Athens (NKUA). She received her doctorate from the National and Kapodistrian University of Athens in Mathematical Methods in Seismology. She is a post-doc researcher at the Department of Electronic Engineering at Hellenic Mediterranean University (HMU). She has published over 20 articles in various fields of seismology. She is reviewer in more 40 scientific journals and serves as editor in 1 scientific journals.
Email: amoshou@hmu.gr

Nataliia Kasianova was born in Kramatorsk, Donetsk region, Ukraine. Graduated from Donetsk State University with a degree in economic cybernetics in 1992. She earned her Doctor of Economics at the Institute of Industrial Economics of the National Academy of Sciences of Ukraine in 2012. She has 30 years of teaching and research experience. She currently works as the Head of the Department of Business Analytics and Digital Economy of the National Aviation University (Kyiv, Ukraine). Her research interests include the problems of managing the development of systems in conditions of uncertainty and chaos, the use of artificial intelligence in economic research. She is author and co-author of more than 200 scientific publications.
Email: nat_kas@ukr.net

Irina Skorniakova was born in eastern Ukraine in 1980. She received a Master's degree in finance from Volodymyr Dahl East Ukrainian National University. In 2007, she defended her thesis and received a PhD from NASU in Kyiv, Ukraine. For many years she has worked in senior positions at the National Academy of Sciences of Ukraine, researched issues related to strategic regional and industrial development, export potential. She has led the development of large-scale projects, such as the economic block of the «Kyiv City Development Strategy Until 2025». She is a member of the Scientific and Technical Council of the State Agency for Resorts and Tourism of the Ministry of Infrastructure of Ukraine. She works as an associate professor at the National Aviation University. She teaches students disciplines related to business digitalisation, digital economy, Big Data and artificial intelligence. Her current scientific interests include research into the latest digital technologies using Big Data processing techniques, modern analytical tools, methods and managerial decision-making models that are informationally and innovatively focused on creating new values for clients.
Email: iryna.skorniakova@npp.nau.edu.ua

Panagiotis Argyrakis holds a B.S. degree in Electronic Engineering and an MSc in Telecommunication from the University of Peloponnese, also he is a Ph.D. Candidate in the Department of Informatics and Telecommunications, of the University of Peloponnese (UoP). He has published more than 36 original papers in international peer-review journals and conferences.
Email: pargyrak@hmu.gr

Nikolaos S. Petrakis is Lecturer at the Electronic Engineering Department of the Hellenic Mediterranean University, in Informatics with Specialization in Computer Programming and Computer Networks. He received the Diploma in Electrical Engineering (1990) from the Polytechnic Institute of Timisoara, and the Ph.D. degree in Reliability and Computer Engineering (1995) from the Technical University of Timisoara, Romania. He has authored or coauthored 18+ research conference papers. Dr. Eng. Nikolaos Petrakis is an active member of the Institute of Electrical & Electronic Engineers (IEEE), Technical Chamber of Greece and Hellenic Association of Electrical Engineering.
Email: nik.s.petrakis@hmu.gr

Appendix I

```
## connect the library to work with the neural network
library (nnet)
## download data
data = events
## build a graph of linear dependence of data
plot (events)
## check the data type
class (events)
## logarithm data, because the output data was a time series
g.series = log (as.numeric (events))
## delete empty values from the data set
events <- na.omit (events)
## set the number of observations
n.obs = length (g.series)
## build a row graph after logarithmization of data
plot (1: n.obs, g.series, type = "l")
## set 12 inputs of the neural network, and define the 13th observation as the initial value to be predicted
g.2 = matrix (rep (0,240 * 13), nrow = 240, ncol = 13)
## fill the data matrices with segments of the series
for (i in 1: 240)
{
g.2 [i,] = g.series [i: (12 + i)]
}
## set the grain as a random number sensor controller
set.seed (12345)
## teach neural network
g.net <-nnet (g.2 [, 1: 12], g.2 [, 13], size = 6, linout = TRUE, rank = 0.1, decay = 0.001, maxit = 1000)
## build the original row in black
plot (1: 252, g.series, type = "l")
## build a row fit in red
lines (13: 252, g.net $ fitted.values, col = "red")
## begin to calculate the predicted values
g.forecast = g.2 [nrow (g.2), - 1]
## set the forecast horizon at level 12 under a given condition
pred.n = 12
## start calculating predicted values
pred.1 = rep (-9999, pred.n)
for (i in 1: pred.n)
{
pred.1 [i] = predict (g.net, g.forecast, type = "raw")
g.forecast = c (g.forecast [-1], pred.1 [i])
}
## b14uild a linear output graph in black*
plot (1: 252, exp (g.series), type = "l", xlim = c (0,252 + pred.n), ylim = c (0,50))
## build a fit in red*
lines (13: 252, exp (g.net $ fitted.values), col = "red")
## build a linear forecast in blue*
lines (252 + 1) :( 252 + pred.n), exp (pred.1), col = "blue")
```

*be sure to set the value of the exponent, because the data was converted to format logarithms.

Appendix II

```
## connect the necessary libraries
library (ggplot2)
library (neuralnet)
library (caret)
## uploading data
data = Greece_data
head (Greece_data)
delete = createDataPartition (Greece_data $ MAGNITUDE, p = 1, list = F)
Greece_data = Greece_data [delete,]
## normalize the data, taking into account the prerequisite of the neural network: the data must be from 0 to 1
samplesize = 0.83 * nrow (Greece_data)
set.seed (50)
index = base :: sample (seq_len (nrow (Greece_data)), size = samplesize)
datatrain = Greece_data [index,]
datatest = Greece_data [-index,]
max = apply (Greece_data, 2, max)
min = apply (Greece_data, 2, min)
scaled = as.data.frame (scale (Greece_data, center = min, scale = max-min))
## we divide training (83% of all data) and test (17% of all data) data sets
trainNN = scaled [index,]
testNN = scaled [-index,]
colnames (trainNN)
## prescribe the appropriate neural network activation function
NN = neuralnet (MAGNITUDE ~ LONG + LAT + DEPTH, trainNN, hidden = c (5,3), err.fct = "sse", linear.output = F)
## visualize the neural network
plot (NN)
## overal results i.e. output for each replication
NN$net.result
NN$weights
NN$result.matrix
## compare the input vector and actual output
NN$covariate
infert$MAGNITUDE
##outcome for all observations and overall assessment of results
NN$net.result[[1]]
NN1=ifelse(NN$net.result[[1]]>0.5,1,0)
NN1
## plot the predicted values
predict_testNN = compute (NN, testNN [, c (2: 4)])
predict_testNN = (predict_testNN $ net.result * (max (Greece_data $ MAGNITUDE) -min (Greece_data $ MAGNITUDE) + min (Greece_data $ MAGNITUDE))
plot (datatest $ MAGNITUDE, predict_testNN, col = "blue", pch = 16, ylab = "Predicted data", xlab = "Real data")
##determine the standard error
RMSE.NN = (sum (datatest $ MAGNITUDE-predict_testNN) ^ 2) / nrow (datatest)) ^ 0.5
RMSE.NN
## build the graphs of dependencies
par(mfrow=c(2,2))
gwplot(NN,selected.covariate="LAT", min=-2.5, max=5)
gwplot(NN,selected.covariate="LONG", min=-2.5, max=5)
gwplot(NN,selected.covariate="DEPTH", min=-2.5, max=5)
```

Appendix III

S. No.	Abbreviation	Description
1	AI	Artificial Intelligence
2	ANN	Artificial Neural Network
3	BPNN	Back Propagation Neural Network
4	CNN	Convolutional Neural Network
5	DEMETER	Detection of Electro-Magnetic Emissions Transmitted from Earthquake Regions
6	DNN	Deep feed-forward fully connected Neural Network
7	FFNN	Feed-forward Neural Network
8	GNSS	Global Navigation Satellite System
9	GMM	Ground-motion Model
10	GLM	General Linear Regression Model
11	GPS	Global Positioning System
12	InSAR	Interferometric Synthetic Aperture Radar
13	MAPE	Mean Absolute Percentage Error
14	MLP	Multi-layer Perceptron
15	NARX	Nonlinear Autoregressive Network with Exogenous Inputs
16	NN	Neural Network
17	NNM	Neural Network Model
18	PGD	Peak Ground Displacement
19	PNN	Probabilistic Neural Network
20	RBF	Radial Basic Function Neural Network
21	RNN	Recurrent Neural Network
22	SOM	Self-organizing Map
23	USSR	Union of Soviet Socialist Republics