Peak ground acceleration prediction using artificial neural networks for Kachchh, Gujarat, India

Prantik Mandal

Received: 5 December 2021 / Accepted: 20 July 2022 / Published online: 12 August 2022
© The Author(s), under exclusive licence to Springer Nature B.V. 2022

Abstract
The uncertainty in the empirical ground motion prediction models (GMMs) for any region depends on several parameters. In the present work, we apply an artificial neural network (ANN) to design a GMM of peak ground acceleration (PGA) for Kachchh, Gujarat, India, utilizing independent input parameters viz., moment magnitudes, hypocentral distances, focal depths and site proxy (in terms of average seismic shear-wave velocity from the surface to a depth of 30 m ($V_{s30}$)). The study has been performed using a PGA dataset consisting of eight engineering seismoscope records of the 2001 $M_w$7.7 Bhuj earthquake and 237 strong-motion records of 32 significant Bhuj aftershocks of $M_w$3.3–5.6 (during 2002–2008) with epicentral distances ranging from 1.0 to 288 km. We apply a feed-forward back propagation ANN method with 8 hidden nodes, which is found to be optimal for the selected PGA database and input–output mapping. The standard deviation of the error has been utilized to examine the performance of our model. We also test the ground motion predictability of our ANN model using real recordings of the 2001 Bhuj mainshock, two $M_w$5.6 Kachchh aftershocks and the 1999 $M_w$6.4 Chamoli mainshock. The standard deviation of PGA prediction error estimates in log10 units is found to be $\pm 0.2554$. Also, the model predictability of our ANN model suggests a good prediction of the PGA for earthquakes of $M_w$5.6–7.7, which are occurring in Kachchh, Gujarat, India.

Keywords Ground motion prediction · Artificial neural networks · Peak ground acceleration · Attenuation · Hypocentral distances · Magnitudes

1 Introduction
In general, ground motion prediction equations (GMPEs) or attenuation relationships form a key input for the earthquake hazard assessment of any region (Joyner and Boore, 1993; Ambraseys et al. 1996; Campbell, 1997). In general, the linear regression analysis is applied on the observed and/or synthetic peak ground acceleration data to determine GMPEs (Joyner and Boore, 1993; Ambraseys et al. 1996; Campbell, 1997; Boore and
Atkinson, 2008; Nath et al. 2009; Anbazhagan et al. 2013). However, this method introduces errors in ground motion predictability due to the assumption of prior functional form. Further, the availability of less numbers of independent input parameters in India in comparison with other developed countries like USA and Europe has resulted in most of the available GPMEs in India as functions of magnitude and hypocentral distances (Nath et al. 2009; Anbazhagan et al. 2013). It is a well-established fact that inclusion of other independent input parameters (like focal depths, $V_{s30}$, soil conditions, focal mechanisms, etc.) can lead to relatively robust prediction in ground motions (Abrahamson and Silva, 2008; Campbell and Bozorgnia, 2008; Castellaro et al. 2008; Kokusho and Sato, 2008). The CSIR-National Geophysical Research Institute (NGRI), Hyderabad, India, had monitored the earthquake activity in Kachchh during 2002–2016, with a close seismic network of 8–12 three-component broadband seismographs and 10–20 three-component strong-motion accelerographs (Fig. 1), which provided an excellent dataset for the robust estimation of moment magnitude, focal depths, hypocentral distances and focal mechanisms. Further toward hazard assessment, several micro-zonation studies have been carried in

---

**Fig. 1** Elevation (in m) map of the Kachchh, Gujarat, India, showing three-component strong-motion stations (marked by solid filled blue triangles). Black dotted lines mark different tectonic faults in the region viz., ABF: Allah Bund fault; IBF: Island Belt fault; GF: Gedi fault; NWF: North Wagad fault; KMF: Kachchh mainland fault; KHF: Katrol Hill fault. The inset shows the elevation map of India showing locations of the 1967 $M_{w}$6.3 Koyna, 1999 $M_{w}$6.4 Chamoli, and 2001 $M_{w}$7.7 Bhuj earthquakes. The study area is marked by a yellow square. AS marks the Arabian sea, while BOB represents the Bay of Bengal.

© Springer
Kachchh, which provided \( V_{s30} \) estimates for the different regions of Kachchh that enabled us to construct maps of \( V_s \) at shallow depths below the Kachchh basin (Mandal and Asano, 2009). Therefore, the availability of PGA data and input parameters (like \( M_w \), focal depths, hypocentral distances, and \( V_{s30} \)) for the Kachchh region has motivated us to apply an artificial neural network method to construct a ground motion model (GMM) for the PGA prediction.

Due to the availability of large data set, nowadays the artificial neural network (ANN) has been used commonly to determine the GMPEs (Ahmad et al. 2008; Alavi and Gandoni, 2011; Güllü and Erçelebi, 2007; Rumelhart et al. 1986; Derras et al. 2012 and 2016; Irshad et al. 2008). By applying an ANN methodology, a simple but powerful relationship is derived with \( M_w \), Focal Depths, Hypocentral distances and shear velocity at 30 m depth (\( V_{s30} \)), for predicting PGA for Kachchh, Gujarat, India. For data fitting, pattern recognition and time series prediction, ANN has been demonstrated to be a powerful tool in a very complex situation. Massive parallel computations with a large number of variables can also be easily performed by an ANN, which can also grasp the nonlinear aspects of the earthquake dataset. Thus, a data-driven ANN can provide robust prediction for ground motions.

In the present work, we apply a feed-forward back propagation ANN method with 8 hidden layers, which is found to be optimal for the selected PGA database and input–output mapping for the Kachchh region. Here, the ANN method also uses the Levenberg–Marquardt training algorithm for mapping input–output datasets, available within ANN toolbox of MATLAB 2018b. Finally, the ground motion predictability of the derived GMM has been tested using recorded PGAs of different Indian earthquakes and predicted PGAs from available GMPEs for India and eastern Northern America.

### 2 Strong motion acceleration dataset

For developing a GMM for PGA prediction using an ANN method, strong motion data of 237 recordings of 32 Kachchh earthquakes of \( M_w 3.3–5.6 \), during 2002–2008, are used (Fig. 1). For our study, eight engineering seismoscope (SRR) records of the 2001 \( M_w 7.7 \) Bhuj earthquake are also utilized. The robust estimates of hypocentral parameters and moment magnitudes have been obtained by using data from 12 broadband seismographs and 10–20 strong-motion accelerographs (Fig. 1). For our ANN modelling, we used the moment magnitudes (\( M_w \)), focal depths (\( D \), m) and hypocentral distances (\( R \), m) as independent inputs while we used \( V_{s30} \) (m/s) as the site proxy. The hypocentral distances for the dataset vary from 335 to 28900 m, while focal depths and \( M_w \) are varying from 3 to 32 km and 3.3–7.7, respectively (Fig. 2a, b, c and d). The dataset is restricted to earthquakes in Kachchh, Gujarat for which the fault rupture lies mainly above a depth of 40 km. The values of \( V_{s30} \) have been obtained from several micro-zonation studies in Kachchh, Gujarat, which were used to construct three-dimensional shallow (0–0.9 km) shear velocity structure below the Kachchh basin (Mandal and Asano, 2019). For peak values, we use the larger of the two horizontal components in the directions as originally recorded. Others (e.g. Campbell, 1981) have used the mean of the two components. The open circles in Fig. 2a, b, c and d show the distribution of the peak acceleration data with moment magnitudes, hypocentral distances, focal depths and \( V_{s30} \) values. For the 2001 \( M_w 7.7 \) Bhuj earthquake, the sources of strong-motion data have been given in previous publications (Cramer and Kumar, 2003; Iyengar and Raghukanth, 2004). The dataset consists of a total of 245
For our ANN modelling, we used 70% (i.e. 171) of dataset for training, while 30% of dataset has been used for testing and validation of the model.

Utilizing the above-discussed PGA dataset of Kachchh earthquakes, Mandal et al. (2009) have derived the following ground motion attenuation relationship for the Kachchh region using the regression analysis following the Joyner-Boore (1993)’s method for a magnitude independent shape

\[
\ln(Y) = -7.9527 + 1.4043M_w - \ln \left( \frac{r_{jb}^2 + 19.82^2}{S} \right)^{0.5} - 0.0682S
\]

for 3.1 < M_w ≤ 7.7

where, \(Y\) is the peak horizontal acceleration in ‘g’, \(M_w\) is the moment magnitude, and \(r_{jb}\) is the closet distance to the surface projection of the fault rupture in km. And, \(S\) is 0 for a soft soil site while \(S\) is 1 for a stiff soil site. However, Eq. (1) can predict ground motion with a standard deviation of ±0.8243. Thus, it is apparent that Eq. (1) cannot provide very robust ground motion prediction (with standard deviation exceeding 0.9). This observation demands a serious effort to improve our GMPE for the Kachchh region, lying in the seismic zone V (BIS, 2002).

![Fig. 2 Observed data used in the ANN modelling. Distribution of records with respect to a Peak ground acceleration (PGA, in m/s*s) and hypocentral distances (in m), b PGA and moment magnitudes (Mw), c PGA and Focal depths (in m), and d PGA and V\textsubscript{s30} (in m/s)]
3 Artificial neural network (ANN) method

A statistical learning method was for the first time developed by McCulloch and Pitts (1943), based on the concept of biological neural networks in the brain, which is known as the Artificial Neural Network (ANN) (Perlovsky, 2001). Different problems without algorithmic solutions or with complex solutions can be obtained through ANN by developing predictive model for the outputs given the inputs. The neurons of each layer are fully connected to the neurons of other layers with connection weights. The connection weights come from the training process of the network, which is done here by implementing the Levenberg–Marquardt back-propagation algorithm (Marquardt, 1963).

Here, we have used a multilayer perception model (MLP) network, which is a kind of ANN with feed-forward back propagation architecture, for predicting PGA, consisting of an input layer, a hidden layer, and an output layer (Cybenko, 1989; Fig. 3a). One or more neurons (nodes) characterize each layer. In this method, first a collection of all nodes, which are multiplied with weights, is performed at each node of hidden layer. Next, a bias is attached to this sum, which is transformed through a nonlinearity function prior to being transferred to the next layer. Functions such as hyperbolic tangent, sigmoid and linear functions can be used as transfer function. Subsequently, the same procedure is followed in this layer to obtain the network output results. The error between the network output and actual observation is estimated at the output layer. Finally, the desired outputs are achieved by back propagating this error to the input layer, through hidden layer in the network. All the synaptic weights of the network remain fixed during the forward processing, while they are adjusted according to an error correction rule during the backward processing (Haykin, 1999). The FFBP network structure used in the present study is shown in Fig. 3b. Here, the input layer is considered to consist of \( x_n \) neurons, while \( h_m \) neurons characterize the hidden layer. And, \( y_l \) neurons characterize the single output layer. The weights (w) consider acting as the link between very nodes in one layer to other layers. Here, \( w_{ij} \) marks the input to hidden weights, while \( w_{ik} \) represents the hidden to output weights. And, the output of the network (\( y_k \)) (Gunaydin and Gunaydin, 2008) can be written as,

\[
y_{ik} = \hat{f} \left[ \sum_{j=0}^{m} w_{jk} \cdot f \left( \sum_{i=1}^{n} (w_{ij} \cdot x_i + b_1 + b_2) \right) \right]
\]

where \( b_1 \) and \( b_2 \) are the biases for the first and second layer, respectively. The activation functions between input and hidden layers, and hidden and output layers are \( f(.) \) and \( \hat{f}(.) \), respectively. Here, functions like the tangent sigmoid (tan-sigmoid), logarithmic sigmoid (log-sigmoid) and linear activation functions have been used for both \( f(.) \) and \( \hat{f}(.) \) for obtaining the best prediction performance.

For learning model, the back-propagation network model (Rumelhart, 1986) is used here, which aims at minimizing iteratively the mean sum squared error (MSE), that defined by

\[
E = \frac{1}{2P} \sum_{p=1}^{P} \sum_{k=1}^{l} (T_{pk} - y_{pk}), \quad p = 1, 2, 3, \ldots, P
\]

where \( T_{pk} \) and \( y_{pk} \) are observed target and predicted output at \( k \)th output node of \( p \)th pattern, respectively. The total number of training patterns considered here is \( P \). At each iteration,
the global error (E) is minimized by adjusting the weights in each layer of the network until convergence is achieved. Otherwise, the same process is repeated for further iterations.

A Learning Epoch is defined as each step in the learning phase. Here, for learning phase Levenberg–Marquardt algorithm (Levenberg, 1944; Marquardt, 1963) has been used that minimizes E and is expressed as:

\[ E = \frac{1}{2} \sum \left( y_i - f(x_i) \right)^2 \]
\[ W_{\text{new}} = W_{\text{old}} - [J^T J + gI]^{-1} J^T E \left( W_{\text{old}} \right) \]  

(4)

where \( J \) is the Jacobian of the error function \( E \), \( I \) is the identity matrix, and \( \gamma \) marks the iteration step value. Here, an adaptive learning rate is used that changes dynamically during the training stage from 0 to 1. We increase the learning rate by the factor learning increment if performance decreases toward the goal for an epoch. Otherwise, we adjust the learning rate by the factor learning decrement when performance increases for an epoch. Here, we use a value of 0.0001 as the performance goal throughout all FFBP simulations. After completing the training phase of the network successfully, a testing dataset (30% of the total data points) is used to examine the performance of the trained model.

The normalization of training input and output datasets is done through the following equation:

\[ x_{ni} = a \cdot \frac{x_i - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}} + b \]  

(5)

where \( x_i \) is the observed data obtained from the \( i^{th} \) record and \( x_{ni} \) is the normalized value of the \( i^{th} \) record. And, \( x_{\text{max}} \) and \( x_{\text{min}} \) are the maximum and minimum values, respectively. Different values can be assigned for scaling factors ‘a’ and ‘b’. Here, we have used different values for a, and b to obtain the best prediction performance. Finally, different values of ‘a’ and ‘b’ are considered for different input and output parameters as discussed in the Supplementary data. This procedure has resulted in normalization of both inputs and output within the range of \([-1.5, +1.5]\). Several trials are made to select the optimum node numbers for the hidden layers. Finally, we obtain the best performance when we used 8 nodes for hidden layers. We stopped the networks training after maximum 10,000 epochs.

4 Proposed ANN Model

Here, an artificial neural network (ANN) has been constructed to predict PGA wherein the site proxy \( \log_{10}(Vs_{30}) \) has been used as an input. Other basic inputs are moment magnitude \( (M_w) \), \( \log_{10} \) (focal depth, \( D \) in m), and \( \log_{10} \) (hypocentral distance, \( R \) in m), while the output is considered as \( \log_{10}(y, \text{in m/s}^2) \). These inputs and output are normalized using Eq. 4 prior to applying the feed-forward ANN method. A feed-forward supervised learning ANN architecture consisting of input, output, and a single hidden layer is chosen in this study as shown in Fig. 3a, b, c and b. One hidden layer is generally acknowledged for describing a nonlinear physical process. The input model with four input variables \([\text{Moment Magnitude, } \log_{10}(D), \log_{10}(R), \log_{10}(Vs_{30})]\) and eight number of nodes in the hidden layer, which reported the least \( \sigma \) during three phases, is considered as the final selected model. The activation functions selected between the input-hidden layer and the hidden-output layer are Tanhsigmoid and linear, respectively. The ANN architecture of the derived model is shown in Fig. 3b.

The general structure of the ANN network of the \( i^{th} \) neuron from the hidden layer is shown in Fig. 4a, wherein information weighted with corresponding connection weights is received by each neuron. While the input to the activation function is formed by the summation of the weighted inputs and the bias, \( b_i \) (Fig. 4a). Finally, the following equation is used to compute the output of the neuron:

\[ Y_i = \phi \left( W_{1i}X_{M_{w,n}} + W_{2i}X_{D_{n}} + W_{3i}X_{R_{j,n}} + W_{4i}X_{V_{30,n}} + b_i \right) \]  

(6)
where $Y_i$ is the output of the $i$th neuron in the hidden layer. $W_{ji}$ defines the connection weight of the $j$th neuron from the input layers and $i$th neuron from the hidden layer. And, $b_i$ defines the bias for $i$th neuron in the hidden layer. Here, the bias is assumed to follow a linear Tanhsigmoid function of $\phi(x) = \tanh(x) = (2/(1 + e^{-2x}) – 1$. Other types of activation function can also be used for the ANN network viz., identity, binary step, logistic, ArcTan, Rectified linear unit, exponential linear unit. The selection of activation function does not necessarily control the accuracy of the ANN model, but it can affect the values of weights.

Fig. 4 Schematic representation of neurons in the proposed ANN model of PGA. a $i$th neuron in hidden layer and b output neuron
between different nodes. During the training process, the value of the bias \((b_i)\) is determined. The output \(Y_i\) is the input of the neurons in the next layer. The general structure of the output is shown in Fig. 4b. It shows clearly that the values of \(Y_i\) calculated in the hidden neurons are set as inputs for this neuron. The values of PGA are predicted by passing through the weighted summation from the activation function. This is achieved by adopting the use of a linear activation function which scales the output of the summation junction to the actual values of the ground motion parameters.

5 Attenuation Models

The above-discussed ANN models are trained and turned into the mathematical formulation of PGA \((cm/s^2)\) as mentioned below:

\[
\log_{10}(PGA) = \left[ b_{PGA} + \sum_{i=1}^{8} v_{i,PGA} \times \left( \frac{2}{1 + e^{-2F_{i,PGA}}} - 1 \right) \right] \tag{7}
\]

where \(b_{PGA}\) represents the bias values of output neurons for ANN models of PGA. While \(v_{i,PGA}\) marks the connection weight between the \(i^{th}\) neuron from the hidden layer and output neuron for each ANN model (Fig. 4a and b). And, \(F_{i,PGA}\) is computed using the following equation:

\[
F_{i,PGA} = w_{1i,PGA} \times \frac{(M_w - 5.5)}{2} + w_{2i,PGA} \times \log_{10}R + w_{3i,PGA} \\
\times \frac{(D - 18)}{10} + w_{4i,PGA} \times (6.3 \times \log_{10}(V_{S30}) - 16.8) \tag{8}
\]

In the analysis, the original data are rescaled to dimensionless values before being input to the networks (see the normalization of data section in the Appendix). We examined several values of number of nodes for hidden layers and finally selected 8 nodes that showed smallest validation loss among the tested models. The logistic sigmoid and rectified linear functions are used as activation functions in the last hidden layer and the other hidden layers, respectively. The linear (identity) function is used in the output layer. The connection weights and biases for our proposed ANN model here are listed in Table 1. Our derived attenuation model is given in Eqs. (7) and (8).

6 Results and discussions

Figure 5a shows the performance of our modelling, suggesting that our model was trained for 11 times or epochs. However, by 5 iterations the model was trained and training was stopped because the model was overfitting after 5 iterations. Here, we used the Levenberg–Marquardt training algorithm to train our dataset. The best validation performance is 0.08825 at epoch 5 (Fig. 5a). The validation dataset of our model aligns with the best fit between 5 and 11 epochs, while our model for training and test datasets suggests lower values than the best fit values between 2 and 11 epochs. Figure 5b shows the error histogram for our NN modelling showing most of the larger positive and negative values are near to the zero error line (shown by a yellow thick line), which suggests the error distribution of our NN modelling is good. Thus, our model is well trained.
Table 1  Connection weights and bias values for the proposed ANN models

| PGA | W₁,PGA | W₂,PGA | W₃,PGA | W₄,PGA | W₅,PGA | W₆,PGA | W₇,PGA | W₈,PGA | b₁,PGA | b₂,PGA |
|-----|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| 1   | 20.09  | 3.45   | -4.18  | 9.27   | 1.60   | 8.72   | 14.53  | 4.8519 | -0.41  | 0.18   | 0.067  |
| 2   | 3.45   | 10.54  | -0.18  | -9.98  | 3.93   | 9.39   | 5.27   | 0.27   | -1.99  | 0.21   |
| 3   | -4.18  | -0.19  | 13.31  | -4.53  | 26.03  | -10.96 | 11.89  | 15.35  | -0.73  | -0.41  |
| 4   | -9.27  | -9.98  | -4.53  | 20.58  | -9.04  | -5.38  | 2.73   | 1.12   | 1.34   | 0.57   |
| 5   | 1.60   | 3.93   | 26.03  | -9.05  | 57.37  | -15.63 | 32.45  | 34.28  | 0.17   | 0.48   |
| 6   | 8.72   | 9.39   | -10.95 | -5.38  | -15.63 | 18.63  | -4.0   | -11.56 | -0.55  | -0.50  |
| 7   | 14.53  | 5.27   | 11.89  | 2.73   | 32.44  | -4.0   | 29.90  | 23.39  | -1.97  | -0.29  |
| 8   | 4.85   | 0.27   | 15.34  | 1.12   | 34.28  | -11.56 | 23.39  | 23.13  | 2.04   | -0.34  |
Figure 6a, b, c and d shows the regression analysis of the neural network modelling for our training, validation, test and complete datasets. Our training dataset consists of 171 data points (i.e. 70% of total data points), while validation and test dataset are having 37 data points (i.e. 15%) each. And, the complete dataset consists of 1 total of 245 data points. Our observed and predicted PGA values (through FFBP modelling) show

Figure 5  a A plot between Mean squared error (MSE) and numbers of Epochs showing the validation of the best performance of the FFBP ANN used in the study is 0.08825 at epoch 5. The performance of ANN on the training, validation and test data is shown by blue, green and red lines, respectively, while the best performance is marked by grey dotted line, b A plot showing error histogram for the training, validation and test data

Figure 6a, b, c and d shows the regression analysis of the neural network modelling for our training, validation, test and complete datasets. Our training dataset consists of 171 data points (i.e. 70% of total data points), while validation and test dataset are having 37 data points (i.e. 15%) each. And, the complete dataset consists of 1 total of 245 data points. Our observed and predicted PGA values (through FFBP modelling) show
high correlation coefficients exceeding 0.90381, except for the validation dataset (correlation coefficient ~ 0.88198). The all, training, and test datasets show a high correlation of 0.90381, 0.92177 and 0.90389, respectively (Fig. 6a, c, and d). Thus, apparently our NN modelling has trained and predicted the data quite well. From Fig. 6a, b, c and d, we get four linear fit lines (Eqs. 8–11) in the form $y = ax + b$, where $x$ and $y$ are observed and predicted (from the FFBP) PGA, respectively, which are mentioned below:
Figure 7a and b shows good agreement between predicted and observed PGA values for training and test datasets, while Fig. 7c and d shows errors in predicted PGA for training and test datasets. A standard deviation of $\pm 0.2554$ is obtained for the errors in PGA prediction for training dataset, while a standard deviation of $\pm 0.2720$ is estimated for the errors in PGA prediction for the test data. Therefore, the PGA prediction from our NN model is found to be good (Fig. 7a, b, c and d).

\begin{align}
  y &= 0.83 \cdot x + 0.12 \\
  y &= 0.76 \cdot x + 0.16 \\
  y &= 0.84 \cdot x + 0.12 \\
  y &= 0.82 \cdot x + 0.13
\end{align}

Fig. 7  
\textbf{a} Predicted log$_{10}$\,(PGA) from the NN modelling for training dataset,  
\textbf{b} Predicted log$_{10}$\,(PGA) from the NN modelling for testing dataset,  
\textbf{c} Error in Predicted log$_{10}$\,(PGA) from the NN modelling for training dataset showing a standard deviation of $\pm 0.2554$, and  
\textbf{d} Error in Predicted log$_{10}$\,(PGA) from the NN modelling for testing dataset showing a standard deviation of $\pm 0.2720$
Fig. 8 The attenuation curves (solid thick red line) for the predicted PGA using the feed-forward data-driven ANN trained network for a Mw5.6, b Mw6.5 and c Mw7.7 events. Observed PGA data (blue open circles) for the 2001 Bhuj mainshock and large Bhuj aftershocks of Mw5.6 and the 1999 Chamoli earthquake have also been plotted to demonstrate the robustness of the prediction. Solid thick grey line marks the predicted attenuation curves from the ground motion relations obtained from the regression analysis of PGA dataset of the 2001 Bhuj earthquake sequence (Mandal et al., 2009). In Fig. 8b, black solid line marks the attenuation curve for the predicted PGA (using ANN model) for an Mw6.0 earthquake. In Fig. 8c, predicted curves for the ground motion attenuation models of Frankel et al. (1996) (dotted green line), Somerville et al. (2001) (dotted magenta line), and Toro et al. (1997) (dotted black line)
7 Prediction of PGA and correlation with the observed PGA from moderate to large Indian Earthquakes

The predicted PGA estimates from our trained ANN model suggest a good agreement with the available observed PGA data for the 2001 Bhuj mainshock (derived from engineering seismographs records) and two 2006 Bhuj aftershocks of $M_w$ 5.6 (Fig. 8a, b and c). Since we do not have observed PGA data from Kachchh events of $M_w$6.0–6.5 events, thus, to validate the PGA predictions from our ANN model we have used available observed PGA data for another Indian earthquake from the Uttarakhand Himalaya namely the 1999 Chamoli earthquake of $M_w$6.4 (Fig. 8b). The predicted PGA values from our ANN model (thick red line in Fig. 8a) show an excellent agreement with the observed PGA data of the 7 March $M_w$5.6 2006 and 6 April $M_w$5.6 2006 Bhuj events. We also plotted the attenuation curve (thick grey line in Fig. 7a) predicted from the ground motion relation of Mandal et al. (2009) that does not show good correlation with the Observed PGA data of two $M_w$5.6 Bhuj events. However, the observed PGA data of the 1999 $M_w$6.4 Chamoli earthquake fall between the predicted PGA curves for $M_w$6.0 and $M_w$6.5 from our ANN model (Fig. 8b). This difference in PGA values could be attributed to the different source processes and tectonics for intraplate Kachchh and interplate Chamoli seismic zones. We also plotted the attenuation curve (thick grey line in Fig. 8b) predicted from the ground motion relation of Mandal et al. (2009) that does not show good correlation with the Observed PGA data of the 1999 Chamoli mainshock of $M_w$6.4. Since there is no observed PGA values that are available for the 2001 Bhuj mainshock except one at 238 km away from the mainshock source zone (i.e. at Ahmedabad). Thus, we have used approximate-corrected PGAs for the 2001 $M_w$7.7 Bhuj mainshock, which were derived from the Engineering Seismoscope (SRR) records (Cramer and Kumar, 2003). Because of this fact, our ANN predicted PGA values for an $M_w$7.7 event show a fair but not very good agreement with the approximate PGA values derived from the SRR data of the 2001 mainshock (Fig. 8c). We also plotted attenuation curves for an $M_w$7.7 event predicted by available ground-motion relations for the eastern North America (Frankel et al. 1996; Toro et al. 1997; Somerville et al. 2001) and Kachchh, Gujarat, India (Mandal et al. 2009). From Fig. 8c, it is quite apparent that the predicted ground motion curve for an $M_w$7.7 event from our ANN model (shown by thick red line) shows a relatively better agreement with the observed data of the 2001 Bhuj earthquake as compared to other predicted curves from other attenuation relations. Thus, we can infer that the predictive PGA estimates from our ANN model suggest a fair to excellent correlation with the observed PGA values of earthquakes of $M_w$ ranging from 5.6 to 7.7, which are occurring in Kachchh, Gujarat, India.

8 Conclusions

We derive a data-driven feed-forward back propagation ANN model (with 8 hidden nodes) for predicting peak ground motions using observed PGA dataset of earthquakes occurring in Kachchh, Gujarat. The dataset consists of thirteen engineering seismoscope (SRR) records of the 2001 $M_w$7.7 Bhuj earthquake and 232 strong-motion records of 32 significant Bhuj aftershocks of $M_w$3.3–5.6 (during 2002–2008) with epicentral distances ranging from 3.3 to 289 km. Our ANN predictive PGA values are validated by correlating with the observed PGA values of Kachchh events of $M_w$5.6 and 7.7, and 1999 Chamoli earthquake.
of Mw 6.4. The standard deviation of PGA prediction error estimates in log10 units is found to be ±0.2554. Our derived ANN model for the Kachchh, Gujarat, India, is applicable within the range of Mw [3.3–7.7], focal depths [3–32 km], epicentral distances [1–288 km] and Vs30 [328–760 m/s].

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s11069-022-05504-7.

Acknowledgements Authors are grateful to the Director, Council of Scientific and Industrial Research—National Geophysical Research Institute (CSIR-NGRI), Hyderabad, India, for his support and permission to publish this work. Authors are grateful to Matlab software of MathWorks Inc. for providing the in-built NNSTART script for the artificial neural network (ANN) modelling. The Matlab script used for our modelling has been provided in the Supplementary data.

Funding Funding was provided by Ministry of Earth Sciences (Grant No. GAP-237-28(PM)).

Data availability Most of the data that support the findings of this study are available in the supplementary material. Figures were plotted using the Generic Mapping Tool (GMT) software (Wessel et al. 2019; https://doi.org/10.1029/2019GC008515). All software and support data related to GMT software are freely accessible and available from this site (www.generic-mapping-tools.org). The elevation data used in generating GMT plots are obtained from the open source Digital Elevation Model (DEM) (https://asterweb.jpl.nasa.gov/gdem.asp). The artificial neural network (ANN) modelling code (NNSTART) in-built in Matlab software developed by MathWorks Inc. is used here.

Declarations

Conflict of interest The authors acknowledge that there are no conflicts of interest recorded.

References

Abrahamson N, Silva W (2008) Summary of the Abrahamson & Silva NGA ground-motion relations. Earthq Spectra 24(1):67–97
Ahmad I, El Naggar MH, Khan AN (2008) Neural network based attenuation of strong motion peaks in Europe. J Earthquake Eng Taylor Francis 12(5):663–680
Alavi AH, Gandomi AH (2011) Prediction of principal ground-motion parameters using a hybrid method coupling artificial neural networks and simulated annealing. Comput Struct Elsevier Ltd 89(23–24):2176–2194
Ambraseys NN, Simpson KA, Bommer JJ (1996) Prediction of horizontal response spectra in Europe. Earthquake Eng Struct Dynam 25(4):371–400
Anbazhagan P, Kumar KA, Sitharam TG (2013) Ground motion prediction equation considering combined dataset of recorded and simulated ground motions. Soil Dyn Earthq Eng 53:92–108. https://doi.org/10.1016/j.soildyn.2013.06.003
Boore DM, Atkinson GM (2008) Ground-motion prediction equations for the average horizontal component of PGA, PGV, and 5%-damped PSA at spectral periods between 0.01 s and 10.0 s. Earthq Spectra 24(1):99–138
Campbell KW (1997) Empirical near-source attenuation relationships for horizontal and vertical components of peak ground acceleration, peak ground velocity, and pseudo-absolute acceleration response spectra. Seismol Res Lett 68(1):154–179
Campbell KW, Bozorgnia Y (2008) NGA ground motion model for the geometric mean horizontal component of PGA, PGV, PGD and 5% damped linear elastic response spectra for periods ranging from 0.01 s to 10 s. Earthq Spectra 24(1):39–171
Castellaro S, Mulargia F, Rossi PL (2008) VS30: Proxy for seismic amplification? Seismol Res Lett 79(4):540–543
Cybenko G (1989) Approximation by superpositions of a sigmoidal function. Math Control Signals Systems 2(4):303–314
Derras B, Bard PY, Cotton F (2016) Site-conditions proxies, ground-motion variability and data-driven GMPEs: Insights from NGA-West 2 and RESORCE data sets. Earthq Spectra 32(4):2027–2056
Derras B, Bard PY, Cotton F, Bekkouche A (2012) Adapting the neural network approach to PGA prediction: an example based on the KiK-net data. Bull Seismol Soc Am 102(4):1446–1461
Frankel A, Mueller C, Barnhard T, Perkins D, Leyendecker EV, Dickman N, Hanson S, and Hopper M (1996) National seismic hazard maps: Documentation June 1996, Open-File Report (United States Geological Survey) 96–532, 41 pp.
Güllü H, Erçelebi E (2007) A neural network approach for attenuation relationships: An application using strong ground motion data from Turkey. Eng Geology Elsevier 93(3):65–81
Gunaydin K, and Gunaydin A (2008). Peak Ground Acceleration Prediction by Artificial Neural Networks for Northwestern Turkey, Hindawi Publishing Corporation Mathematical Problems in Engineering Volume 2008, Article ID 919420, 20 pages. doi:https://doi.org/10.1155/2008/919420
Haykin S (1999) Neural Networks: A Comprehensive Foundation. Prentice-Hall, Englewood Cliffs, NJ, USA
Irshad A, Naggar MHEI, Khan AN (2008) Neural network based attenuation of strong motion peaks in Europe. J Earthq Eng 12:663–680
Joyner WB, Boore DM (1993) Methods for regression analysis of strong-motion data. Bull Seismol Soc Am 83(2):469–487
Kokusho T, Sato K (2008) Surface-to-Base amplification evaluated from KiK-net vertical array strong motion records. Soil Dynam Earthq Eng 28(9):707–716
Levenberg K (1944) A method for the solution of certain non-linear problems in least squares. Q Appl Math 2:164–168
Mandal P, Kumar N, Satyamurthy C, Raju IP (2009) Ground-motion attenuation relation from strong-motion records of the 2001 Mw 7.7 Bhuj Earthquake Sequence (2001–2006), Gujarat, India. Pure Appl Geophys 166:1–19
Marquardt D (1963) An algorithm for least-squares estimation of nonlinear parameters. J Soc Ind Appl Math 11:431–441
McCulloch WS, Pitts W (1943) A logical calculation of the ideas immanent in nervous activity. Bull Math Biophys 5:115–133
Nath SK, Raj A, Thingbaijam KK, Kumar A (2009) Ground motion synthesis and seismic scenario in Guwahati city—a stochastic approach. Seismol Res Lett 80(2):233–242. https://doi.org/10.1785/gssrl.80.2.233
Perlovsky LI (2001) Neural networks and intellect: Using model-based concepts. Oxford University Press, New York
Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by back-propagating errors. Nature 323:533–536
Somerville P., Collins N., Abrahamson N., Graves R., and C. Saikia (2001). Ground motion attenuation relations for the central and eastern United States, Final report to the U.S. Geological Survey, 30 June 2001, URS Group, Inc., Pasadena, California, 36 pp.
Toro G, Abrahamson N, J. Schneider J. (1997) Model of strong ground motions from earthquakes in central and eastern North America: Best estimates and uncertainties. Seismol Res Lett 68:41–67

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

Springer Nature or its licensor holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.