Pure Appl. Geophys. 178 (2021), 1959–1976
© 2021 The Author(s), under exclusive licence to Springer Nature Switzerland AG
https://doi.org/10.1007/s00024-021-02745-8

Pure and Applied Geophysics

Single-Component/Single-Station–Based Machine Learning for Estimating Magnitude and Location of an Earthquake: A Support Vector Machine Approach

Sarit Chanda[1,2] and Surendra Nadh Somala[1]

Abstract—Traditional approaches require a velocity model to compute travel times for estimating the location of earthquakes. Moreover, the velocity model typically assumed is layered in nature, ignoring the perturbations around the background velocity model. In this study, we propose a novel method that does not require any assumption on a velocity model. Also, triangulation estimates of location require three-station recordings. Even if the single station is used, three components of recordings are needed. In this work, we propose a machine learning method, based on support vector machines (SVM), that can predict earthquake location and magnitude with just one component (vertical) of a single station. Machine learning methods are typically used for either regression or classification. Literature shows that the SVM algorithm is promising for the classification of an arbitrary signal. This research demonstrates that, by using complementary input features from a seismic recording, a comparable performance can be obtained, with a substantial reduction in detection time. Furthermore, only the vertical component of a single recording station data is used to train the network. The SVM algorithm is applied on synthetic seismograms from 400 earthquakes of different focal mechanisms and magnitudes. This falls under supervised machine learning, as we give the input features and use them for training the model. Overfitting is avoided by tenfold cross-validation, i.e., the algorithm is repeated ten times, each time giving a different 10% as a testing portion. The classifier's performance is quite good, on the synthetic noise-free data, in predicting the magnitude, elevation angle, and hypocentral distance, but not the azimuthal angle. The trained model in the present work can be readily used where noise levels are quite low without sophisticated ray-tracing or Green's function computation. The performance of the algorithm is checked by adding additive noise to the data. The novelty of this work is that knowledge of the velocity model is not required, and the estimation of magnitude and the origin of the earthquake are done using the single-station/single-component earthquake records.

Keywords: Machine learning, support vector machine, magnitude estimation, earthquake location estimation, cross-validation.

1. Introduction

Earthquake magnitude and location are important to earthquake engineers to derive expected peak ground motion the magnitude and distance information. The location depends on knowledge of the travel time difference between P and S waves. Traditionally, by observing the interval between the P and S wave arrival, the earthquake's location and magnitude have been estimated (Wiejacz & Wiszniowski, 2006). A minimum of three seismic recording stations is used to estimate the location and the earthquake's origin time (Karasözen & Karasözen, 2020). Observed phase arrival times and first-movement polarities of seismic P waves are also useful in deciding hypocenters and the focal mechanisms (e.g., Bergen et al., 2019; Hardebeck, 2002; Jiao & Alavi, 2020; Yang et al., 2012). These quantities, typically, are estimated by human specialists in manual mode. However, as seismic recording systems have been increasingly deployed around the world, such calculations have progressively been automated (Gentili & Bragato, 2006). Still, there are many uncertainties in identifying the P and S wave arrival in the seismogram.

P wave and S wave phase picking is an arduous task. The detection of teleseismic events, using P-wave onset, was done by Tiira (1999) using an artificial neural network (ANN), a machine learning (ML) algorithm. Deep learning (DL) algorithms were used on millions of seismograms from Southern California recordings to pick the P phase and polarity of first motion (Ross et al., 2018). Similarly, Zhou et. al. (2019) have used both a convolutional neural network (CNN) and recurrent neural network (RNN)

[1] Indian Institute of Technology, Hyderabad, India. E-mail: ce16resch11006@iith.ac.in; surendra@iith.ac.in

[2] IcfaiTech, ICFAI Foundation for Higher Education, Hyderabad, India.

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

for phase picking post-2008 Wenchuan earthquake for aftershocks.

Earthquake magnitude-related works that used ML or DL were mostly to predict the expected magnitude, a priori. Audretsch et. al. (2020) are probably one of the very few works that considered CNN to predict the magnitude of small events obfuscated by background noise. Asim et. al. (2019) have proposed to assess the earthquake magnitude based on the fault parameters. Until now, log-linear regression equations are individually used for each fault parameter. Nevertheless, this may lead to uneven magnitude forecasts as the nonlinear parameter correlations are ignored, and those parametric functions cannot consider potential deviations from log-linear scaling. To respond to the deficiencies, they employ ANN to assess the magnitude of earthquakes while at the same time using all the available fault parameters, thereby excluding the possibility of conflicting assessments. Asencio-Cortés et. al. (2017) have used ANN to predict the magnitude of the earthquake in Tokyo.

With the computational power advances, ML and DL algorithms are gaining popularity in seismology (Bellagamba et al., 2019; Bianco et al., 2019; DeVries et al., 2018; Kong et al., 2019; Yoon et al., 2019; Mignan & Broccardo, 2019; Tibi et al., 2019; Xie et al., 2020). As discussed earlier in the literature, several ML algorithms are employed to estimate the earthquake magnitude (Lomax et al., 2019; Mousavi & Beroza, 2020; Tian et al., 2020), but most of the research works use all three components of seismic recordings. The seismic recording sensors, such as accelerometers or seismometers, are quite expensive. In this study, we will use only a single component of the seismogram (i.e., vertical component) to estimate the magnitude and the location of the earthquake. We have extracted six features from the seismic signal and estimated the magnitude and location of the earthquake using a machine learning approach. The feature vector looks like {average amplitude, peak absolute amplitude of the signal, root mean square of the seismogram, P-wave onset, onset of S-wave, and P–S interval}. The support vector machine (SVM) algorithm (Fig. 1) is employed as a primary classifier. Conventionally, a minimum of three seismic stations is required to locate an earthquake using the triangulation method. In our study, we have trained an SVM model to estimate the location of the earthquake using just one component (vertical) of a single station.

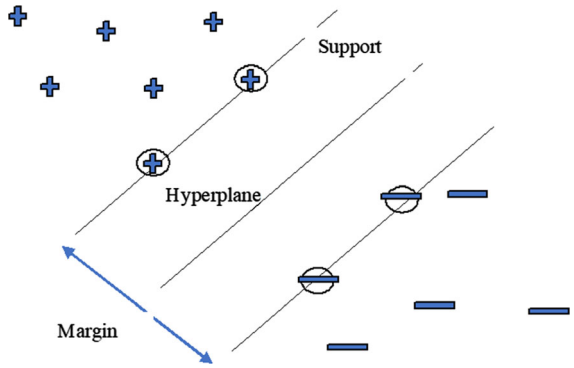


Figure 1
Support vector machine schematic illustrating how the algorithm separates data points of two distinct kinds using a couple of support vectors separated by some margin

Moreover, the actual earthquake signal consists of noises. Deep learning techniques such as ANN and CNN are used to classify noisy seismic signals (Kislov & Gravirov, 2017). To evaluate the SVM with the noisy data, we have added the different levels of additive noise to the simulated data and applied the algorithm to predict the location of the earthquakes.

2. *Data and Methodology*

In this research work, the goal is to develop a machine learning model with 400 Mw $\geq$ 4 earthquakes that are synthesized using SPECFEM3D (Komatitsch & Tromp, 2002a, 2002b). The earthquake data are recorded at a single seismic station in a hypothetical domain of 100 km $\times$ 100 km $\times$ 50 km (Fig. 2). The station is considered as the origin and serves as a reference point. The station is at the free surface and center of the domain for all the simulated earthquakes. The topside of the domain is a free surface, which is the natural boundary condition in finite element methods. The lateral and bottom sides of the domain use absorbing boundary conditions (Clayton & Engquist, 1977). We have restricted our analysis to noise-free data. The earth's crust is

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

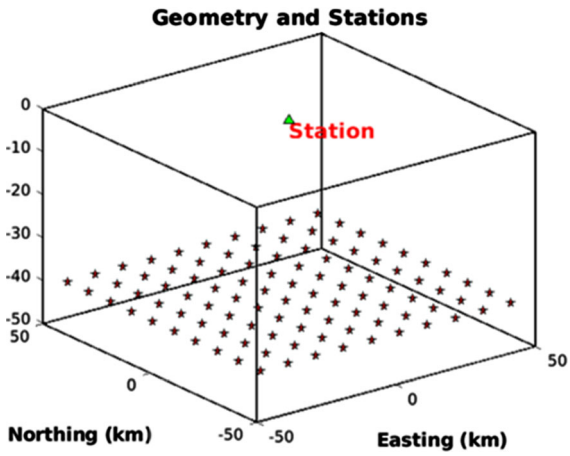


Figure 2
A hypothetical domain used in this study. The domain is 100 km × 100 km × 50 km. Earthquakes are considered to be located 40 km deep, with a single station at the center of the free surface, which is also the domain's origin. Earthquake locations are separated by 10 km, which is roughly the spacing used by many agencies for the first estimation of source position

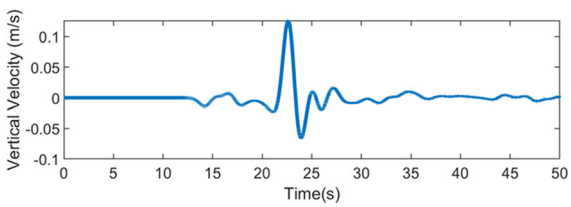


Figure 3
A sample 3-component synthetic seismogram for a Mw 7 earthquake, located at 75 km hypocentral distance from the single station at the origin

heterogeneous, with material properties following von Karman distribution. A two-layered velocity model has been assumed for generating the synthetic seismogram. P-wave velocity, S-wave velocity, and density for the top 25 km of the domain are 6 km/s, 3.5 km/s, and 2690 kg/m^3, respectively, and for the second layer, they are 7.2 km/s 4.15 km/s and 3070 kg/m^3, respectively. This limits the results to bedrock level analysis. A 1-km element size with five Gauss Lobatto Legendre nodes is used, resulting in a timestep of 0.01 s resolving frequencies up to 2.86 Hz.

Hypocenters of the earthquakes are spaced 10 km apart, at a depth of 40 km, and the moment magnitude of the earthquakes is varied from 4 to 7. In this

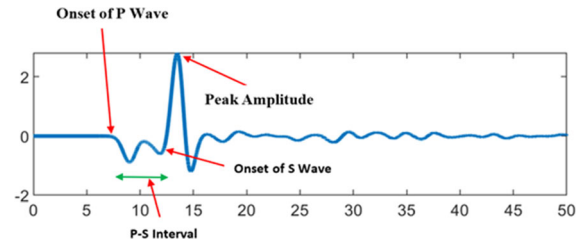


Figure 4
Feature extraction from a seismogram. Some of the features based on one channel of a single station are shown here

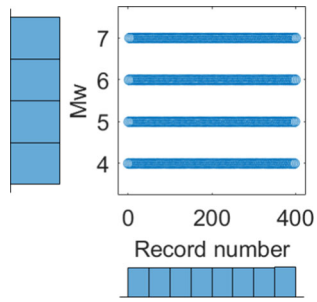


Figure 5
Magnitude in the training data. All the simulated earthquakes belong to one of the four categories of magnitude, Mw 4, 5, 6, and 7

study, we have considered a constant depth of 40 km for all the 400 earthquakes. We are interested in shallow depth earthquakes (less than 70 km) whose mid-depth is roughly 40 km. Moreover, the source location is in the 2nd layer of the two-layered velocity model. The seismogram is complex enough to include the reflections and refractions and the layer transition at 25 km. To simulate the earthquakes, a pure thrust focal mechanism (i.e., Strike: 90, Dip: 45, Rake: 90) is considered in earthquake nucleation point. A total of 400 earthquakes are simulated, and seismograms are computed at the top center of the domain (mid-point of the free surface). A sample synthetic seismogram for Mw 7 at 75 km hypocentral distance is shown in Fig. 3.

We employed an SVM method (Vapnik, 2013) to categorize the earthquakes. For solving the problem, we have chosen to use the MATLAB package to implement the SVM algorithm. The SVM method is intended to categorize the objects between two

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Table 1

Hyperparameters for magnitude, distance, azimuth, elevation estimations

Optimized hyperparameter	Magnitude estimation	Hypocentral distance estimation	Azimuth estimation	Elevation estimation
Kernel function	Gaussian	Quadratic	Gaussian	Gaussian
Epsilon	0.005825	46.7627	12.5933	0.001
Box constraint	326.9508	381.4286	21.3334	0.20323

Table 2

Classifier performance for magnitude, distance, azimuth, elevation estimations

Performance metric	Magnitude estimation	Hypocentral distance estimation	Azimuth estimation	Elevation estimation
RMSE	0.0412	485.53	68.85	0.0056422
R-Squared	1.00	1.00	0.58	1.0
Mean squared error (MSE)	0.00169	235,700	4741.5	0.00003
Mean absolute error (MAE)	0.009419	268.64	58.94	0.0015

classes, i.e., SVM is a binary classifier. The SVM classification involves two phases:

(1) Training of the classifier, i.e., using the known classes, the SVM identifies the boundary line of each class and generates a support vector, (2) Testing of the classifier, i.e., the trained model determines the class of an unknown vector. In the current case, an object is an earthquake source parameter, i.e., magnitude, location of the hypocenter, etc. The different features in the seismic signal are combined as a feature vector.

2.1. SVM Method

The outline of SVM is shown in Fig. 1. The SVM initially transforms every object of the training data set into a higher dimension feature vector with each discriminate feature in the vector as one of that space's dimensions. A kernel function is employed to weigh each discriminant feature vector before the transformation, which converts a nonlinear classification statement into a linear problem. The SVM then discovers the two best support vectors that distinguish between the two classes and gives the largest space between them in the training data. The training vectors which lie in the two best hyperplanes are called the support vectors. These best hyperplanes are often called the local classifiers, and the area between the local classifiers is called the margin. The SVM categorizes the classifier as a hyperplane in the middle of the margin. After the training process, the optimal classifier model is obtained. Then the unknown objects are transformed in the same space, and finally, the model predicts the class of the unknown objects in the testing phase.

The goal is to identify a hyperplane that separates the classes apart, while maximizing the margin. The equation of any plane passing through a point b with normal vector n, can be written as:

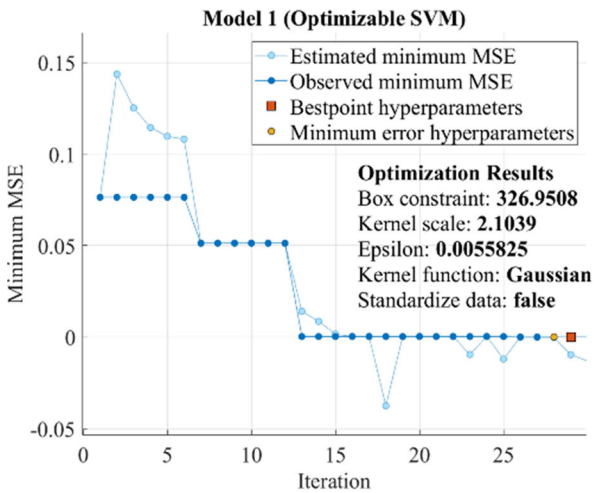
$$n.(r - b) = 0$$

where r is the position vector i.e., $r = (x,y)$ in 2-D. The equation of the plane, in words, is the dot product of normal vector with any vector on the plane is zero, as they are perpendicular to each other. This can be further simplified as:

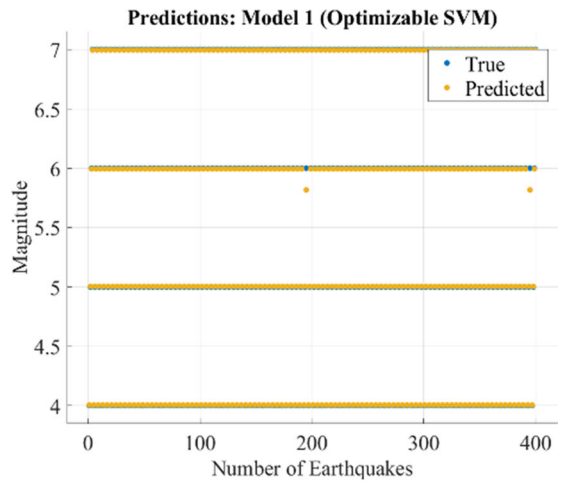
$$n.r + c = 0$$

where c is negative of the dot product of n and b. This plane is described by just two parameters n and c. So, the goal of finding the optimal hyperplane amounts to solving for these two parameters. Note that the hyperplane is flat for Euclidean coordinates and is just a line, when the space is 2-D. The primal–dual (Christianini & Shawe-Taylor, 2000; Hastie et al., 2008) way is one approach that can help to identify these parameters.

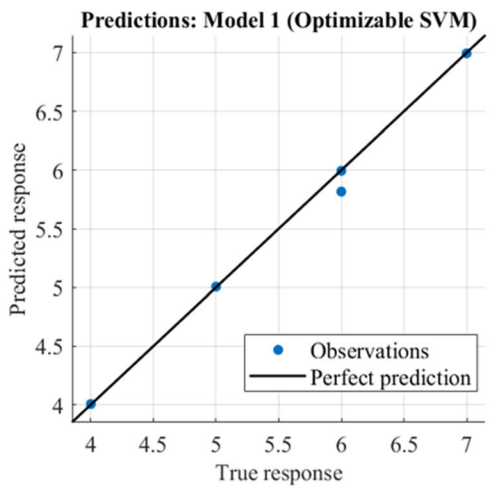
Content courtesy of Springer Nature, terms of use apply. Rights reserved.



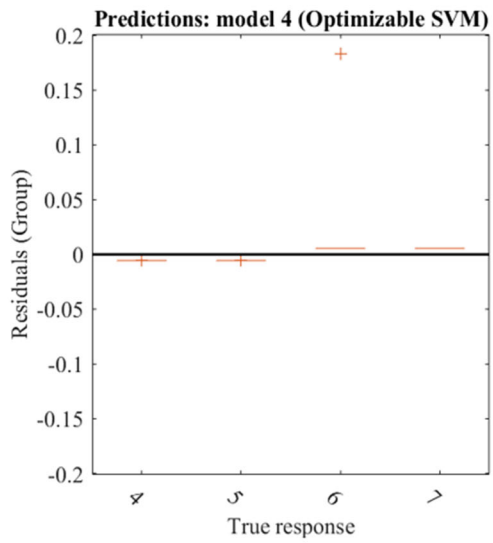
(a) Training Process



(b) Response Plot



(c) Predicted vs Actual Plot



(d) Residual Plot

Figure 6
Performance of magnitude estimation by the SVM model. **a** Training model, **b** response plot, **c** predicted vs. actual plot, **d** residual plot. Given the limited number of discrete values tried, the scatter appears a bit high. The misfit decreases drastically in less than five iterations but takes a few more iterations to converge

For the non-separable data points (in lower dimensional space), it is difficult to form a hyper-plane. In such cases, SVM uses a soft margin, which separates many data points but not all. In some cases, a kernel function is used to transform the data point to a higher dimensional space (separable problem) and then determines the hyper-plane.

2.2. SVM Classification for Multiclass Problems

The SVM classification is a binary classifier. To use SVM for multiple class problem, a "one versus all" configuration can be employed. This methodology divides all the classes into multiple binary classification problems. Each binary classifier estimates whether the object in the class or not.

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Table 3

Classifier performance for magnitudes with different level of noises

Performance metric	Magnitude estimation without any noise	Magnitude estimation with 1% noise	Magnitude estimation with 3% noise	Magnitude estimation with 5% noise
RMSE	0.0412	0.14304	0.14174	0.25107
R-Squared	1.00	0.98	0.98	0.95
Mean squared error (MSE)	0.00169	0.0204	0.02	0.06815
Mean absolute error (MAE)	0.009419	0.0469	0.0529	0.07496

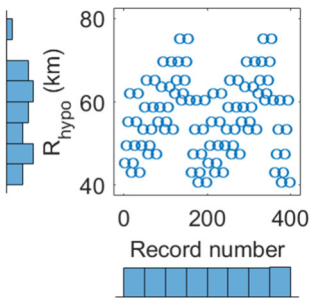


Figure 7
Hypocenter distance in the training data. Earthquakes were 40 km deep, located on a 10 km × 10 km grid, from -50 km $< X <$ 50 km and -50 km $< Y <$ 50 km, below the station at origin. The resultant distances from such earthquake locations and single station combinations are shown in the figure in meters

2.3. Kernels

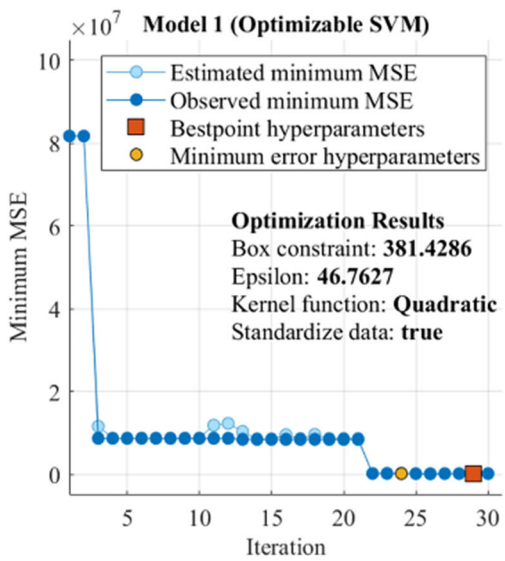
The kernel trick is useful in nonlinear problems i.e., those which are not linearly separable. Let us say if a circle separates the data points in 2-D, rather than a line, then the kernel trick is used to transform the data space in such a way that the partitioning hyperplane is actually a line. The kernel functions possible are 1. Gaussian or Radial Basis Function (RBF), 2. Linear, 3. Polynomial. Their functional form is described below. The kernel functions transform the feature vector to a higher dimension and determine the hyperplane to separate the data.

1. Gaussian or Radial Basis Function (RBF): $K(x_1, x_2) = exp\left(-\frac{\|x_1 - x_2\|^2}{2\sigma^2}\right)$
2. Linear: $K(x_1, x_2) = x_1^T x_2$
3. Polynomial: $K(x_1, x_2) = tanh\left(\beta_0 x_1^T x_2 + \beta_1\right)$

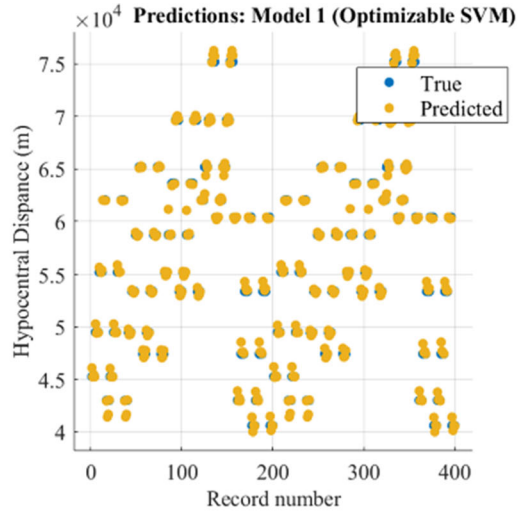
3. Feature Extraction

Feature extraction from the seismic signal is a crucial part of the training process. It is important to find useful information from the data. Seismic signals come together with noise, either induced by the instruments or by the surrounding ambient noise, or a combination of both. The high-frequency noise and the low-frequency noise can be eliminated by appropriate filters. Noise present in the signal can lead to an inaccurate ML model. Low magnitude earthquakes have a lower signal to noise ratio as the noise floor is fixed. Proper care needs to be taken to denoise the signal. The seismic recording is a combination of source information to the path information and local site effects. In this study, we have neglected the site effects by considered homogeneous material properties, and hence the single station at origin is located on the bedrock. As the rupture propagates on the fault and radiates waves, the P wave first reaches the seismic station as it has the highest propagation velocity. The particles move back and forth as the wave travels from the source to the receiver. Thus, in the vertical component of ground motion, the P wave can be observed first. So, the P wave arrival time can be computed from the vertical channel of the seismometer. The initial P wave amplitude is extracted to form one of the entities of the feature vector. Similarly, the P wave onset in all three channels is observed and added to the feature vector. Next, S wave phase picking is marked, and the arrival time is added to the feature vector. The P to S interval gives the hypocenter distance from the recording station. So, the P–S time interval also forms yet

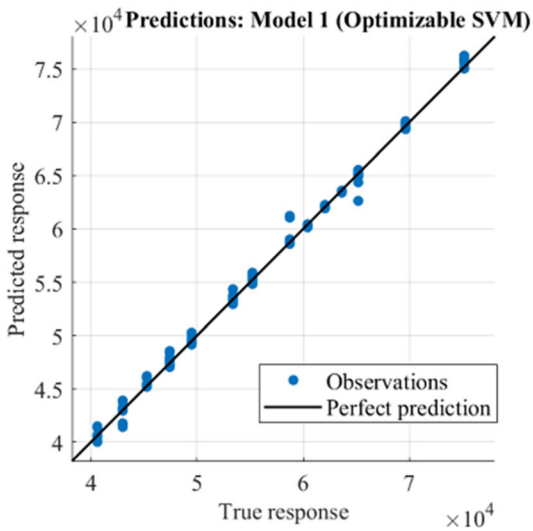
Content courtesy of Springer Nature, terms of use apply. Rights reserved.



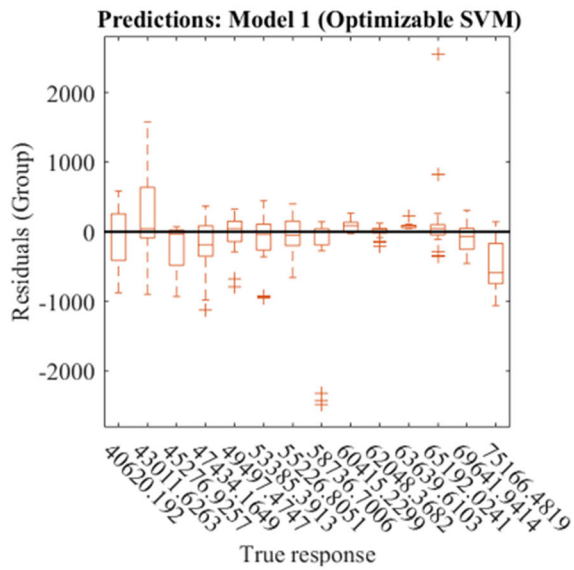
(a) Training Process



(b) Response Plot



(c) Predicted vs Actual Plot



(d) Residual Plot

Figure 8

Performance of hypocentral distance estimation using the SVM model. **a** Training model, **b** response plot, **c** predicted vs. actual plot, **d** residual plot. There is a rapid reduction in misfit in just a few iterations, and the errors are within a few hundred meters, without any trend as a function of hypocentral distance. Residuals are slightly higher for distant earthquakes compared to those close to the single station

another entity in the feature vector. Similarly, the S wave's maximum amplitude and corresponding time are also added to the feature vector. To estimate the direction of the hypocenter, the P wave incident angle needs to be computed. P wave incident angle is also extracted (Herglotz, 1907; Wiechert, 1910) and stored as one of the feature vector's entities. The first motion polarity of the P wave is also included in the feature vector. In this study, we have considered six entities in each

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

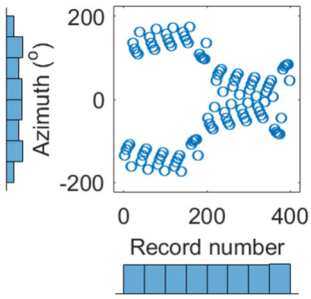


Figure 9
Azimuth in training data. With the single station at the origin and earthquakes distributed within -50 km $< X <$ 50 km and -50 km $< Y <$ 50 km, all azimuths are covered equally well, resulting in an almost uniform distribution over the possible azimuths

feature vector to train the model. Four of the six features are shown in Fig. 4, and the remaining two are the average amplitude and the root-mean-square (RMS) of the seismogram.

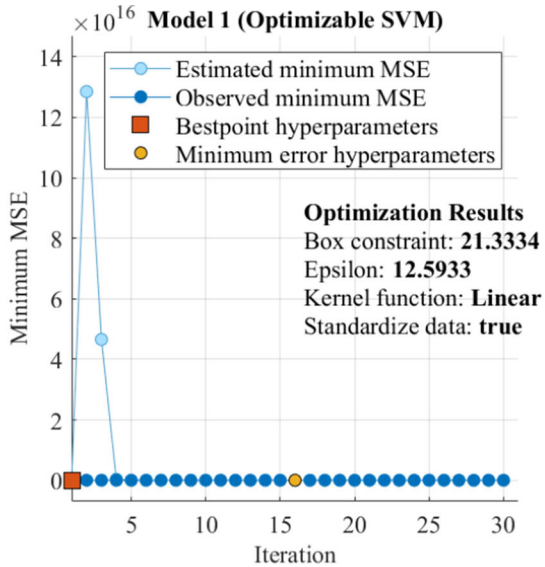
4. Results and Discussion

The SVM classifiers are employed to classify 400 Mw $\geq$ 4 earthquakes that are simulated in the hypothetical domain. A six-entity feature vector is extracted from each seismic signal to train the SVM classifier for magnitude, distance, azimuth, and elevation.

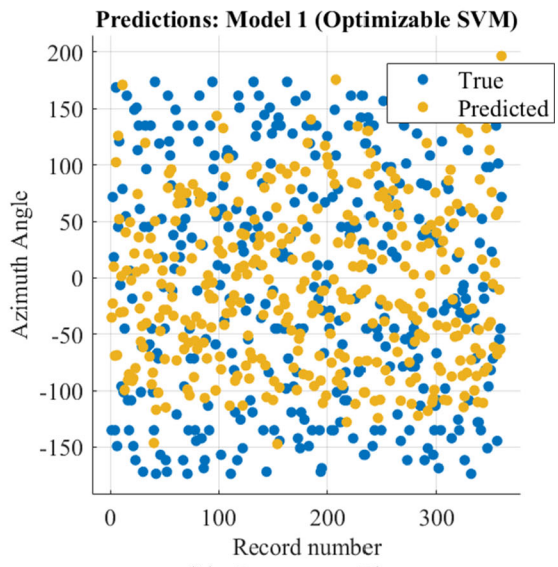
To train the classifier for magnitude prediction, we have used the Gaussian kernel function to transform the feature vector into a higher dimension. Other kernels have also been tried out, and the Gaussian kernel is found to give an optimal prediction in the case of magnitude. The observed data is discrete, ranging from 4 to 7 in the training data (Fig. 5), with an equal number of records for each magnitude. We have used Bayesian optimization to determine the optimal input parameters such as kernel scale, epsilon, and the box constraint. Kernel scale divides the input feature vector to make the maximum separation boundary. Table 1 shows the optimal hyperparameters for magnitude estimation. Tenfold cross-validation is used to avoid overfitting. The accuracy of the classifier is evaluated in terms of root mean squared error (RMSE), R-squared, mean squared error (MSE), and mean absolute error (MAE) are computed. Table 2 shows the statistical parameters conveying the performance of the classifier. The RMSE of the classifier is 0.0412, and the R-squared value is 1 for magnitude estimation. Based on the classifier's performance, it is evident that the trained model is quite accurate in predicting the magnitude of the earthquake. The performance of the SVM algorithm is also shown pictorially using a (a) training model, (b) response plot, (c) predicted vs. actual plot, (d) residual plot. The four plots are hereinafter referred to as performance plots. For the magnitude prediction, performance plots are shown in Fig. 6. Response plot compares the predicted and actual values, along with differences (errors) between them, as a function of samples. A residual plot shows the errors and is useful in figuring out any trend in the errors as a function of the predicted parameter. The trained model plot shows how the error is decreasing with iterations. A plot of the predicted values as a function of actual values helps understand how tightly constrained the prediction is from the $y = x$ line.

The hypocentral distance estimation (whose distribution is shown in Fig. 7), using a quadratic kernel function, gives the optimal model with an R-squared value of 1, RMSE of 485.53 MAE of 268.64. So, the trained model is very efficient in predicting the hypocentral distance from the station. The mean absolute error is hypocentral distance is about 270 km. The performance plots for distance prediction are shown in Fig. 8. Distances are quite accurately estimated within a few hundreds of meters accuracy, using just one component, when earthquakes are in proximity to the single station. As the distance increases, the scatter increases in the predicted vs. actual plot. However, there is no specific trend as can be observed from the residual plot. A point worth noting is that distance estimation training

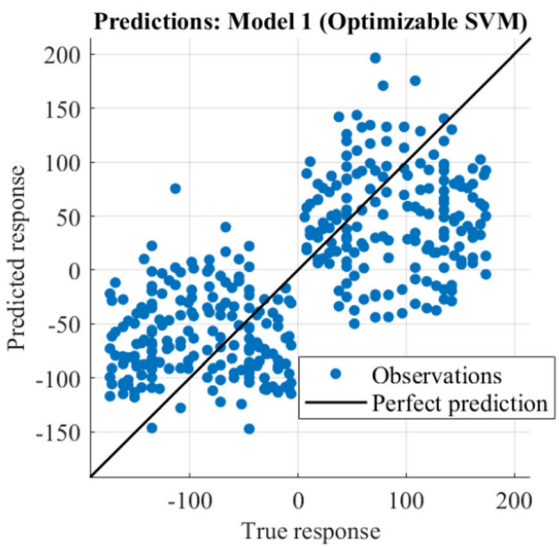
Content courtesy of Springer Nature, terms of use apply. Rights reserved.



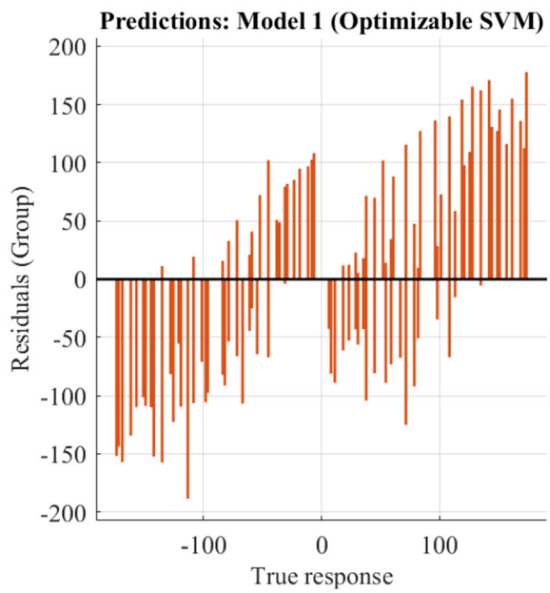
(a) Training Process



(b) Response Plot



(c) Predicted vs Actual Plot



(d) Residual Plot

Figure 10
Performance of azimuth estimation by the SVM model. **a** Training model, **b** response plot, **c** predicted vs. actual plot, **d** residual plot. The reduction in function is only slightly more than 50%, and there is some trend in the residuals

converges quite faster than that for magnitude prediction (Figs. 6 and 8).

To estimate the direction of the source from the recording station, the azimuth angle model is trained. The observed data are well separated (Fig. 9). The Gaussian kernel (found to be optimal among all the kernels) function is deployed to transform the feature vector into a higher dimensional space. The performance plots of the classifier for azimuth prediction are shown in Fig. 10. The *R*-squared value, in this

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

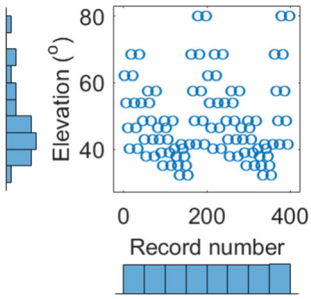


Figure 11
Elevation (in radians) in the training data. With earthquakes at a fixed depth of 40 km and a single station at the origin, a limited number of elevations are covered. Elevation indirectly constrains the depth of the earthquake, having found the hypocentral distance

case, is 0.58. Azimuthal prediction performance of the classifier is not as good as for magnitude and distance. The six features extracted in this work from the input data might not be sufficient to capture the orientation of the source with respect to the station, thereby leading to inaccurate prediction of the azimuth angle of the earthquake source. Inaccurate estimation of azimuth angle can lead to wide variation in the location of the earthquake.

Finally, to predict the location, along with the hypocenter distance and azimuth angle, elevation angle is also required. Indirectly, the elevation angle estimation is identical to depth estimation, having estimated the distance already. In estimating the elevation angle, we have used a Gaussian kernel (found to be optimal among all the kernels) function to train the SVM model. The distribution range of elevation angle in radians is limited in the record number domain (Fig. 11). The performance plots are shown in Fig. 12. No trend is observed in the residuals. The trained model is also highly accurate in predicting the elevation angle. The *R*-squared value is 1, the RMSE value is 0.25107, and the MAE value is 0.07496. Elevation estimation also converges rapidly with iterations, like hypocentral distance estimation (Figs. 8 and 12).

4.1. Feature Importance

The relative score of each feature in predicting the magnitude is computed using an *F*-test and is shown in Fig. 13. In estimating the magnitude, many of the domain expert related features like S wave arrival time, P–S time difference, incident angle, and polarity of P wave have negligible contribution when compared to simple statistical measures like the average amplitude, maximum amplitude, and the root mean squared amplitude of the signal. The only domain expert feature of significant Importance is the P wave onset, which still has a slightly lower contribution than the mean, maximum, and RMS of the signal.

The *F*-test for the hypocentral distance is shown in Fig. 14. The test result shows that the mean amplitude of the signal has the least contribution to the machine learning model. In contrast, the maximum amplitude of the signal, followed by the RMS and onset of P and S waves, has a substantial impact on the model.

For the estimation of elevation angle, the *F*-test plots are shown in Fig. 15. Simple statistics like mean amplitude have maximum impact followed by maximum amplitude, RMS, onsets of P and S waves, P–S time difference, incident angle, and P wave polarity.

In the case of azimuthal angle, the *F*-test results are shown in Fig. 16. Only the mean amplitude of the signal has importance on the machine learning model. All other features have a negligible or null impact on the model. For this reason, the performance of the machine learning algorithm is reduced in predicting the azimuthal angle.

4.2. Noise

Additive noise is dealt with in this study. To check our methodology with the noisy data, we have considered an additive noise of 5% in the synthetic signal. The noisy signal is shown in Fig. 17. In magnitude prediction without any noise, the machine learning model's performance is almost perfect (Fig. 6). The noisy data is used to train the SVM, and the performance plots are shown in Fig. 18. The *R*-Squared value is 0.95, and the RMSE value is 0.25107, and MAE is 0.07496 (Table 3). The classifier's performance has degraded as the data contain noise, but the magnitude prediction is still reasonably good for higher magnitudes (above Mw 5). The higher residuals for Mw 4 case could be because the signal's amplitude is comparable to the

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

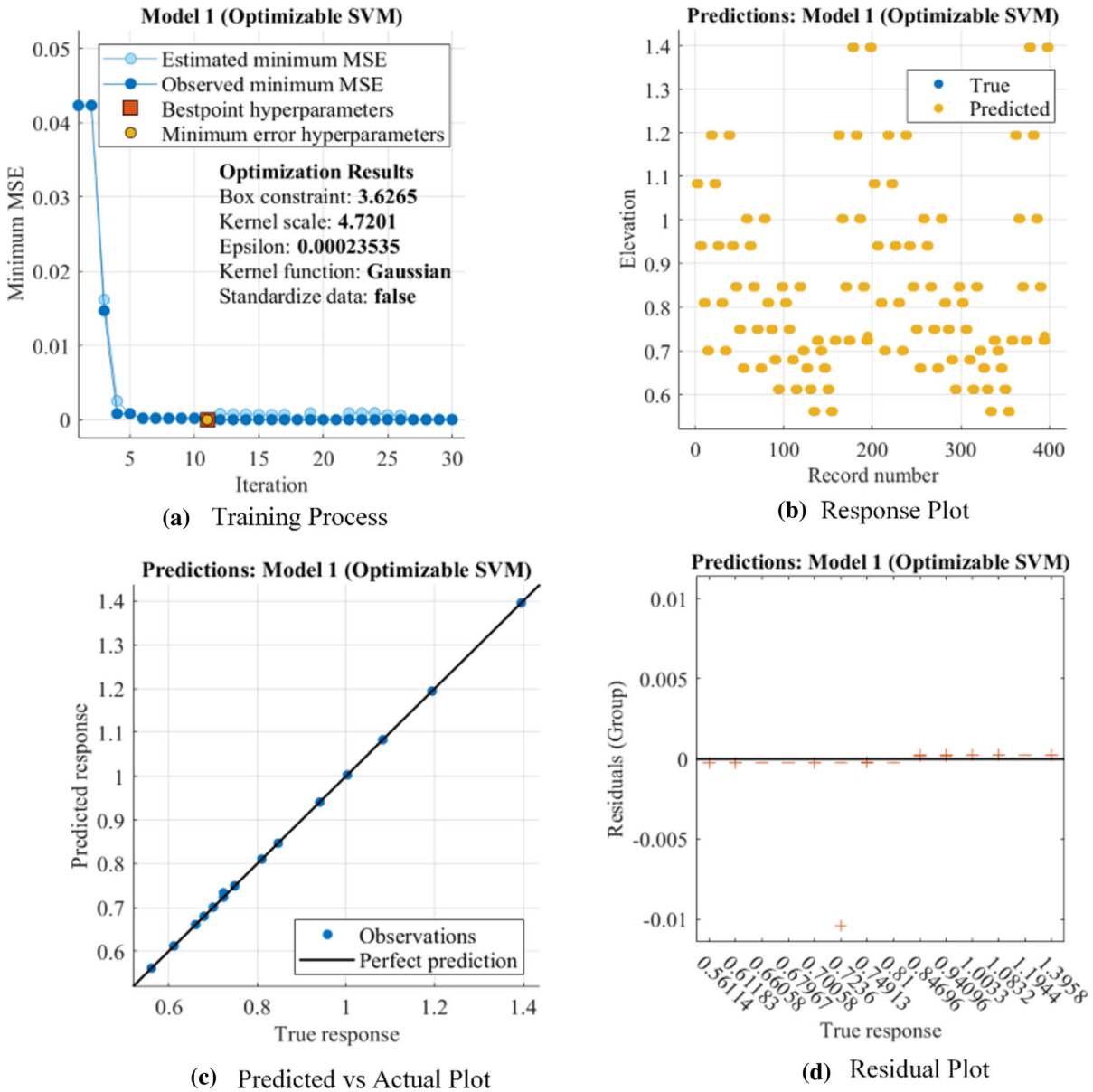


Figure 12
Performance of elevation angle (in radians) estimation by the SVM model **a** training model **b** response plot, **c** predicted vs. actual plot **d** residual plot. The elevation is well constrained, with predicted values following the actual values very closely

noise added, thus decreasing the signal-to-noise ratio significantly. Similarly, the SVM's efficiency with the noisy data is checked for the hypocentral distance (Table 4, Fig. 19) and elevation angle (Table 5, Fig. 20).

5. *Conclusion*

We have used the SVM algorithm to estimate the magnitude and location of the earthquake using a single station/single component of seismic records without knowing the velocity model. SVM classifiers

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Table 4

Classifier performance for hypocentral distance with different level of noises

Performance metric	Hypocentral distance estimation without any noise	Hypocentral distance estimation with 1% noise	Hypocentral distance estimation with 3% noise	Hypocentral distance estimation with 5% noise
RMSE	485.53	2873.1	2651.4	727.16
R-Squared	1.00	0.9	0.91	0.99
Mean squared error (MSE)	2,357,000	8,254,800	7,030,200	5,287,700
Mean absolute error (MAE)	268.64	2287	2119	150.91

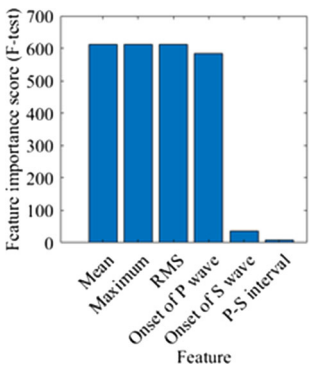


Figure 13
Relative importance of the 8-vector feature set used in this study in predicting the magnitude (*F*-test). Simple statistics including mean, maximum, and RMS have the highest Importance, followed by the P wave's onset

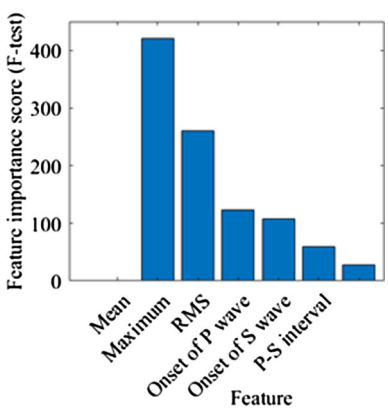


Figure 14
Relative importance of the 8-vector feature set used in this study in predicting the hypocentral distance (*F*-test). Simple statistics including maximum amplitude have maximum impact, followed by RMS and onsets of P and S waves. The mean amplitude of the signals has an almost null impact on the machine learning model

are computed based upon the differentiated features of the training data from the hypothetical domain. The six feature vectors are extracted from the seismogram. The feature vector has been trained with a different kernel function, and the kernel that gave maximum accuracy for each of the source-site parameters is identified. Except for the azimuthal angle, other parameters in this study (hypocentral distance, elevation angle, and magnitude) are quite well estimated using the machine learning algorithm. Hypocentral distance coupled with elevation angle indirectly gives the epicentral distance and hypocentral depth. The exact orientation of the earthquake relative to the station may potentially need more work. It should be highlighted here that no knowledge of the velocity model was assumed in estimating the magnitude and location using SVM, unlike traditional works (those that do not use machine learning), which requires at least an approximate background velocity model.

The trained algorithm in this work can be used on any earthquake recording, where noise is known to be sufficiently low to estimate magnitude and distance. Our rationale is that many noise removal tools are available these days (e.g., based on wavelet decomposition) to get rid of additive noise. The accuracy for the estimation of earthquake magnitude is 100%. Similarly, the accuracy of the elevation angle and hypocentral distance estimation also 100%. Further, the different level of additive noise is added to the signal, and the result shows the algorithm's performance is good. This work can be seamlessly extended to include site effects. The advantage here is that the

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Table 5

Classifier performance for elevation angle with different level of noises

Performance metric	Elevation angle estimation without any noise	Elevation angle estimation with 1% noise	Elevation angle estimation with 3% noise	Elevation angle estimation with 5% noise
RMSE	0.0056422	0.0127	0.0094	0.033
R-Squared	1.0	1.00	1.00	0.97
Mean squared error (MSE)	0.00003	0.0001	0.00008	0.0011
Mean absolute error (MAE)	0.0015	0.011	0.003	0.0175

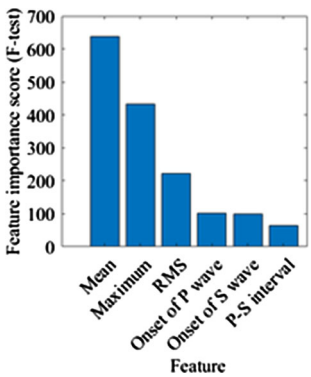


Figure 15
Relative importance of the 8-vector feature set used in this study in predicting the elevation angle (*F*-test). Simple statistics such as mean amplitude have maximum impact, followed by maximum amplitude, RMS, and onsets of P and S waves

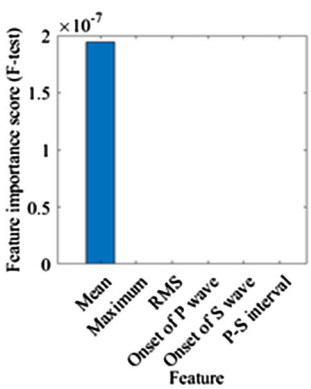


Figure 16
Relative importance of the 8-vector feature set used in this study in predicting the azimuth angle (*F*-test). Only the mean of the signal has an impact on the machine learning model. The remaining seven feature vectors have no impact

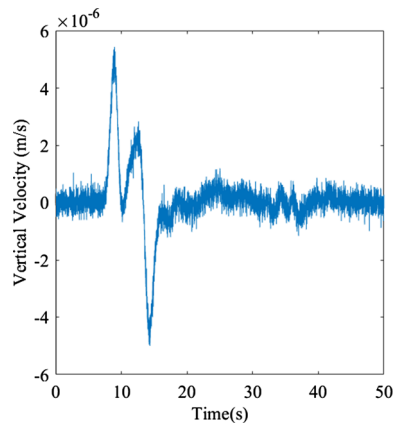
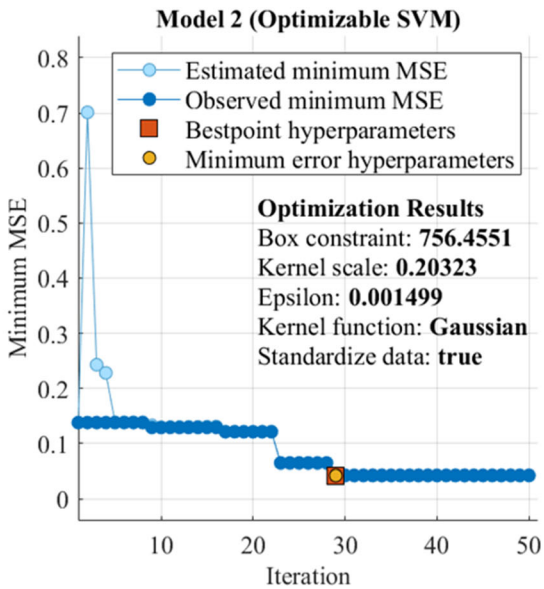


Figure 17
A sample plot of 5% noise added to a vertical component of an earthquake of magnitude 4

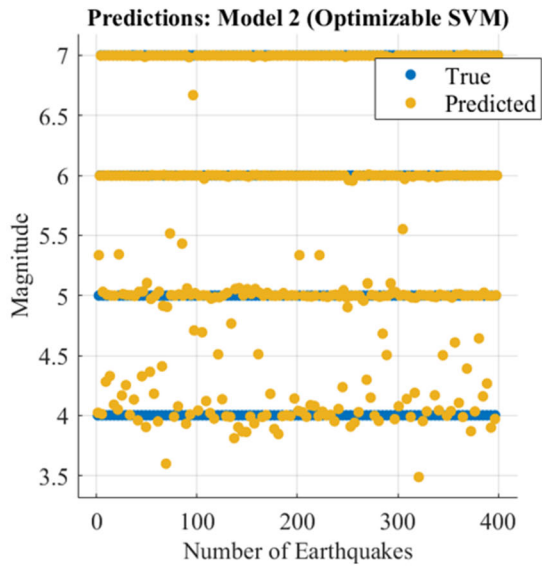
trained model can be used even by those not domain experts, predicting magnitude and location for any available recording.

With the advent of graphical processing unit (GPU) computing, deep learning approaches are being adopted in various fields. The machine learning approach we used in this work requires feature selection and extraction. However, if GPU is available, deep learning-based unsupervised learning can be tried out in the future to automate further the earthquake parameter estimation based on single station recording.

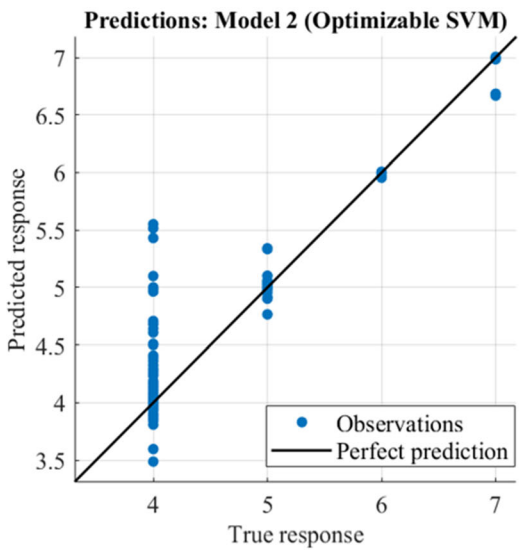
Content courtesy of Springer Nature, terms of use apply. Rights reserved.



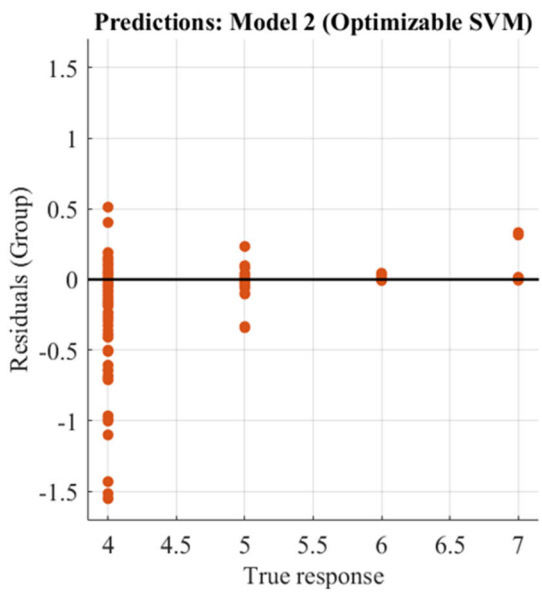
(a) Training Process



(b) Response Plot



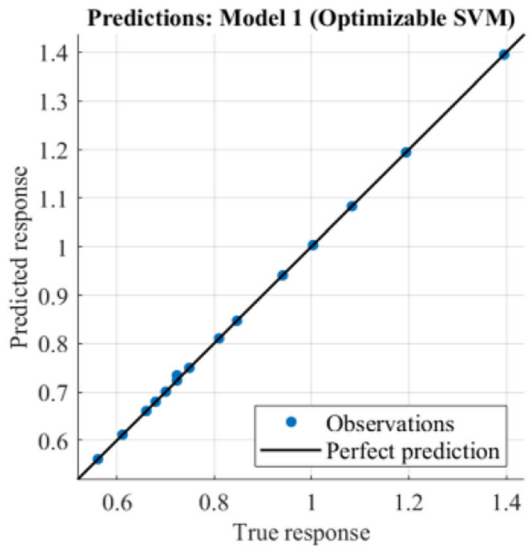
(c) Predicted vs Actual Plot



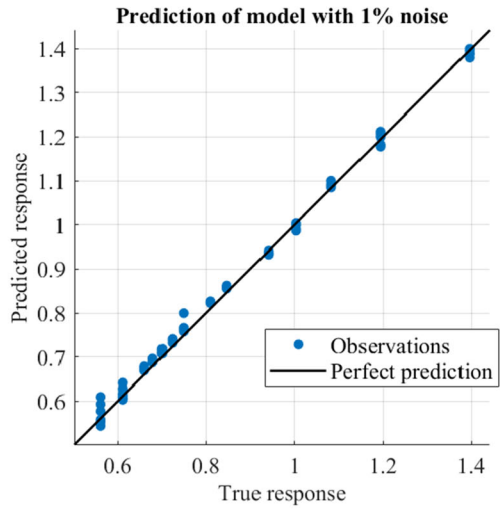
(d) Residual Plot

Figure 18
Performance of magnitude estimation by the SVM model using noisy data (5%). **a** Training model, **b** response plot, **c** predicted vs. actual plot, **d** residual plot. The residuals increase for Mw 4 cases, but Mw 6 and 7 are still quite well predicted

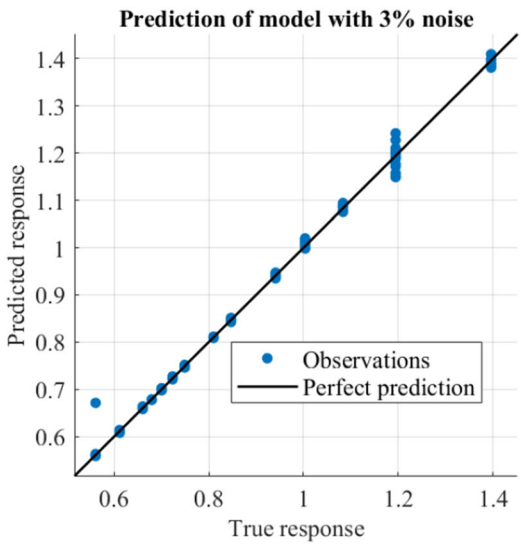
Content courtesy of Springer Nature, terms of use apply. Rights reserved.



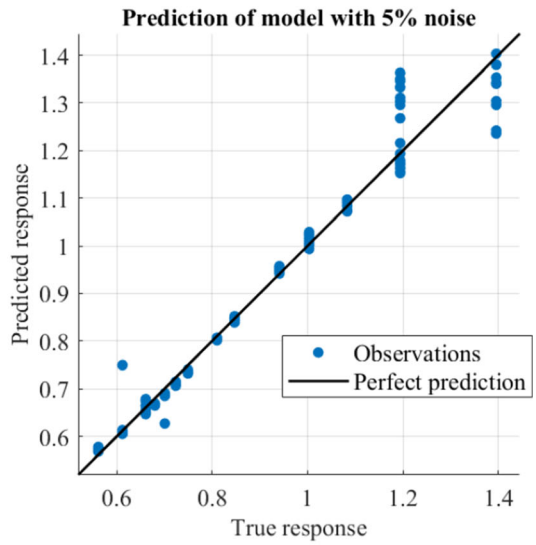
(a) Without noise



(b) 1% noise



(c) 3% noise



(d) 5% noise

Figure 19
Performance of elevation angle by the SVM model using noisy data. **a** Without any noise, **b** 1% noise, **c** 3% noise, **d** 5% noise

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

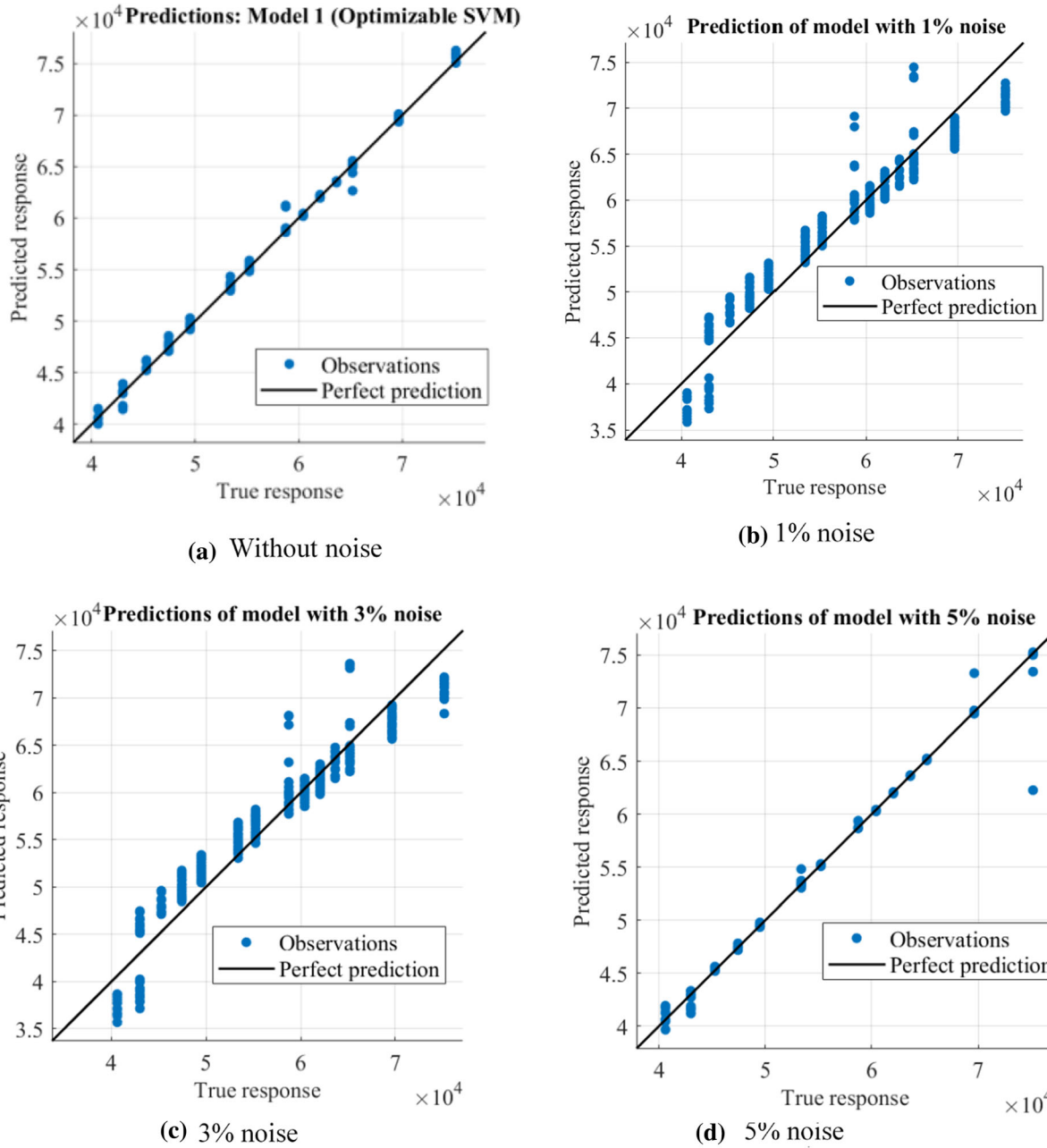


Figure 20
Performance of hypocentral distance estimation by the SVM model using noisy data. **a** Without any noise, **b** 1% noise, **c** 3% noise, **d** 5% noise

Acknowledgements

We thank the two anonymous reviewers for their comments and suggestions, which have significantly improved the manuscript. Funding from the Ministry of Earth Sciences, India through grant MoES/P.O.(Seismo)/1(304)/2016 is gratefully acknowledged.

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Funding

Ministry of Earth Sciences, India.

Data Availability

Not applicable.

Code Availability

SPECFEM3D is an open-source software freely available through GitHub (https://github.com/geodynamics/specfem3d).

Declarations

Conflict of Interest The authors declare no conflict of interest.

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

References

Asencio-Cortés, G., Martínez-Álvarez, F., Troncoso, A., & Morales-Esteban, A. (2017). Medium–large earthquake magnitude prediction in Tokyo with artificial neural networks. *Neural Computing and Applications, 28*(5), 1043–1055. https://doi.org/10.1007/s00521-015-2121-7

Asim, K. M., Javed, F., Hainzl, S., & Iqbal, T. (2019). Fault parameters-based earthquake magnitude estimation using artificial neural networks. *Seismological Research Letters, 90*(4), 1544–1551. https://doi.org/10.1785/0220190051

Audretsch, J. (2020). Earthquake Detection using Deep Learning Based Approaches (Thesis). https://doi.org/10.25781/KAUST-52098

Bellagamba, X., Lee, R., & Bradley, B. A. (2019). A neural network for automated quality screening of ground motion records from small magnitude earthquakes. *Earthquake Spectra, 35*(4), 1637–1661. https://doi.org/10.1193/122118EQS292M

Bergen, K. J., Chen, T., & Li, Z. (2019). Preface to the focus section on machine learning in seismology. *Seismological Research Letters, 90*(2A), 477–480. https://doi.org/10.1785/0220190018

Bianco, M. J., Gerstoft, P., Olsen, K. B., & Lin, F.-C. (2019). High-resolution seismic tomography of Long Beach, CA using machine learning. *Scientific Reports, 9*(1), 14987. https://doi.org/10.1038/s41598-019-50381-z

Christianini, N., & Shawe-Taylor, J. (2000). *An introduction to support vector machines and other kernel-based learning methods*. Cambridge University Press.

Clayton, R., & Engquist, B. (1977). Absorbing boundary conditions for acoustic and elastic wave equations. *Bulletin of the seismological society of America, 67*(6), 1529–1540

DeVries, P. M. R., Viégas, F., Wattenberg, M., & Meade, B. J. (2018). Deep learning of aftershock patterns following large earthquakes. *Nature, 560*(7720), 632–634. https://doi.org/10.1038/s41586-018-0438-y

Gentili, S., & Bragato, P. (2006). A neural-tree-based system for automatic location of earthquakes in Northeastern Italy. *Journal of Seismology, 10*(1), 73–89. https://doi.org/10.1007/s10950-005-9001-z

Hardebeck, J. L. (2002). A new method for determining first-motion focal mechanisms. *Bulletin of the Seismological Society of America, 92*(6), 2264–2276. https://doi.org/10.1785/0120010200

Hastie, T., Tibshirani, R., & Friedman, J. (2008). *The elements of statistical learning*. (2nd ed.). Springer.

Herglotz, G. (1907). Über das Benndorfsche Problem der Fortpflanzungsgeschwindigkeit der Erdbebenstrahlen. *Zeitschrift fúr Geophys, 8*, 145–147

Jiao, P., & Alavi, A. H. (2020). Artificial intelligence in seismology: advent, performance and future trends. *Geoscience Frontiers, 11*(3), 739–744. https://doi.org/10.1016/j.gsf.2019.10.004

Karasözen, E., & Karasözen, B. (2020). Earthquake location methods. *GEM International Journal on Geomathematics, 11*(1), 13. https://doi.org/10.1007/s13137-020-00149-9

Kislov, K. V., & Gravirov, V. V. (2017). Use of artificial neural networks for classification of noisy seismic signals. *Seismic Instruments, 53*(1), 87–101

Komatitsch, D., & Tromp, J. (2002a). Spectral-element simulations of global seismic wave propagation—I. Validation. *Geophysical Journal International, 149*(2), 390–412

Komatitsch, D., & Tromp, J. (2002b). Spectral-element simulations of global seismic wave propagation—II. Three-dimensional models, oceans, rotation and self-gravitation. *Geophysical Journal International, 150*(1), 303–318

Kong, Q., Trugman, D. T., Ross, Z. E., Bianco, M. J., Meade, B. J., & Gerstoft, P. (2019). Machine learning in seismology: turning data into insights. *Seismological Research Letters, 90*(1), 3–14. https://doi.org/10.1785/0220180259

Lomax, A., Michelini, A., & Jozinović, D. (2019). An Investigation of rapid earthquake characterization using single-station waveforms and a convolutional neural network. *Seismological Research Letters, 90*(2A), 517–529. https://doi.org/10.1785/0220180311

Mignan, A., & Broccardo, M. (2019). One neuron is more informative than a deep neural network for aftershock pattern forecasting. *Nature, 574*(7776), E1–E3. https://doi.org/10.1038/s41586-019-1582-8

Mousavi, S. M., & Beroza, G. C. (2020). Bayesian-deep-learning estimation of earthquake location from single-station observations. *IEEE Transactions on Geoscience and Remote Sensing*, *58*(11), 8211–8224. https://doi.org/10.1109/TGRS.2020.2988770.

Ross, Z. E., Meier, M.-A., & Hauksson, E. (2018). P wave arrival picking and first-motion polarity determination with deep learning. *Journal of Geophysical Research: Solid Earth, 123*(6), 5120–5129. https://doi.org/10.1029/2017JB015251

Tian, X., Zhang, W., Zhang, X., Zhang, J., Zhang, Q., Wang, X., & Guo, Q. (2020). Comparison of single-trace and multiple-trace polarity determination for surface microseismic data using deep learning. *Seismological Research Letters, 91*(3), 1794–1803. https://doi.org/10.1785/0220190353

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Tibi, R., Linville, L., Young, C., & Brogan, R. (2019). Classification of local seismic events in the Utah region: a comparison of amplitude ratio methods with a spectrogram-based machine learning approach classification of local seismic events in the Utah region. *Bulletin of the Seismological Society of America, 109*(6), 2532–2544. https://doi.org/10.1785/0120190150

Tiira, T. (1999). Detecting teleseismic events using artificial neural networks. *Computers and Geosciences, 25*, 929–938. https://doi.org/10.1016/S0098-3004(99)00056-4

Vapnik, V. (2013). *The nature of statistical learning theory*. Springer science & business media.

Wiechert, E. (1910). Bestimmung des Weges der Erdbebenwellen im Erdinnern. I. Theoretisches. *Physikalishce Zeitschrift, 11*, 294–304

Wiejacz, P., & Wiszniowski, J. (2006). Moment magnitude determination of local seismic events recorded at selected Polish seismic stations. *Acta Geophysica, 54*(1), 15–32. https://doi.org/10.2478/s11600-006-0003-1

Xie, Y., Ebad Sichani, M., Padgett, J. E., & DesRoches, R. (2020). The promise of implementing machine learning in earthquake engineering: a state-of-the-art review. *Earthquake Spectra*. https://doi.org/10.1177/8755293020919419

Yang, W., Hauksson, E., & Shearer, P. M. (2012). Computing a large refined catalog of focal mechanisms for Southern California (1981–2010): temporal stability of the style of faulting. *Bulletin of the Seismological Society of America, 102*(3), 1179–1194. https://doi.org/10.1785/0120110311

Yoon, C. E., Bergen, K. J., Rong, K., Elezabi, H., Ellsworth, W. L., Beroza, G. C., & Levis, P. (2019). Unsupervised large-scale search for similar earthquake signals. *Bulletin of the Seismological Society of America, 109*(4), 1451–1468. https://doi.org/10.1785/0120190006

Zhou, Y., Yue, H., Kong, Q., & Zhou, S. (2019). Hybrid event detection and phase-picking algorithm using convolutional and recurrent neural networks. *Seismological Research Letters, 90*(3), 1079–1087. https://doi.org/10.1785/0220180319

(Received July 22, 2020, revised March 31, 2021, accepted April 24, 2021, Published online May 12, 2021)

Content courtesy of Springer Nature, terms of use apply. Rights reserved.

Terms and Conditions

Springer Nature journal content, brought to you courtesy of Springer Nature Customer Service Center GmbH (“Springer Nature”).
Springer Nature supports a reasonable amount of sharing of research papers by authors, subscribers and authorised users (“Users”), for small-scale personal, non-commercial use provided that all copyright, trade and service marks and other proprietary notices are maintained. By accessing, sharing, receiving or otherwise using the Springer Nature journal content you agree to these terms of use (“Terms”). For these purposes, Springer Nature considers academic use (by researchers and students) to be non-commercial.
These Terms are supplementary and will apply in addition to any applicable website terms and conditions, a relevant site licence or a personal subscription. These Terms will prevail over any conflict or ambiguity with regards to the relevant terms, a site licence or a personal subscription (to the extent of the conflict or ambiguity only). For Creative Commons-licensed articles, the terms of the Creative Commons license used will apply.
We collect and use personal data to provide access to the Springer Nature journal content. We may also use these personal data internally within ResearchGate and Springer Nature and as agreed share it, in an anonymised way, for purposes of tracking, analysis and reporting. We will not otherwise disclose your personal data outside the ResearchGate or the Springer Nature group of companies unless we have your permission as detailed in the Privacy Policy.
While Users may use the Springer Nature journal content for small scale, personal non-commercial use, it is important to note that Users may not:

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

In line with the restriction against commercial use, Springer Nature does not permit the creation of a product or service that creates revenue, royalties, rent or income from our content or its inclusion as part of a paid for service or for other commercial gain. Springer Nature journal content cannot be used for inter-library loans and librarians may not upload Springer Nature journal content on a large scale into their, or any other, institutional repository.
These terms of use are reviewed regularly and may be amended at any time. Springer Nature is not obligated to publish any information or content on this website and may remove it or features or functionality at our sole discretion, at any time with or without notice. Springer Nature may revoke this licence to you at any time and remove access to any copies of the Springer Nature journal content which have been saved.
To the fullest extent permitted by law, Springer Nature makes no warranties, representations or guarantees to Users, either express or implied with respect to the Springer nature journal content and all parties disclaim and waive any implied warranties or warranties imposed by law, including merchantability or fitness for any particular purpose.
Please note that these rights do not automatically extend to content, data or other material published by Springer Nature that may be licensed from third parties.
If you would like to use or distribute our Springer Nature journal content to a wider audience or on a regular basis or in any other manner not expressly permitted by these Terms, please contact Springer Nature at

onlineservice@springernature.com