Physics of the Earth and Planetary Interiors 327 (2022) 106879

Contents lists available at ScienceDirect

Physics of the Earth and Planetary Interiors

journal homepage: www.elsevier.com/locate/pepi

Forecasting strong subsequent earthquakes in California clusters by machine learning

S. Gentili [a,*], R. Di Giovambattista [b]

[a] National Institute of Oceanography and Applied Geophysics - OGS, Via Treviso 55, 33100, Udine, Italy
[b] Istituto Nazionale di Geofisica e Vulcanologia, Via Di Vigna Murata 605, 00143 Rome, Italy

ARTICLE INFO

Keywords:
Largest aftershock
Seismicity
Machine learning
Bath's law
Aftershock sequence
California seismicity

ABSTRACT

In this paper, we propose an innovative machine learning approach called NESTORE, which analyses seismic clusters to forecast strong earthquakes of magnitudes similar or greater to those of the mainshock. The method analyzes the seismicity in the first hours/days after the mainshock and provides the probability of having a strong subsequent earthquake. The analysis is conducted at various stages of time to simulate the increase in knowledge over time. We address the main problem of statistics and machine learning when applied to spatiotemporal variation of seismicity: the small datasets available, on the order of tens or fewer instances, need a more accurate analysis with respect to the classical testing procedures, where hundreds or thousands of data are available. In addition, we develop a more robust NESTORE method based on a jackknife approach (rNESTORE), and we successfully apply it to California seismicity.

1. Introduction

Many strong earthquakes are followed by one or more subsequent large earthquakes (SLE) of magnitudes similar to the initial quake or even stronger. Repeating earthquakes cause accumulated damage to already weakened buildings and infrastructure; therefore, forecasting their occurrence is a challenging task from the viewpoint of civil protection to stop the continuous loss of lives. Studies have concentrated on the value of *Dm*, defined as the difference in magnitude between the mainshock and the strongest SLE (SSLE). The smaller *Dm* is, the higher is the magnitude of the SSLE, and the more dangerous is the cluster. The reason why the studies on this topic are based on *Dm* instead of on the SSLE magnitude is that the self-similarity theory of seismicity is adopted, which allows similar behavior to be assumed for shocks of different magnitude. While after the end of the cluster the mainshock(s) can be defined as the strongest earthquake(s), in real-time or near real-time applications executed during the cluster, when a strong magnitude is recorded, it is not known whether it is the mainshock or simply the first strong earthquake, and a stronger event will follow. Our method is designed to become a near real-time application, so we need additional definitions. In the following section, we will speak of the o-mainshock (the abbreviation for operative-mainshock), which represents the first shock of the cluster over a given magnitude threshold. For large values of the magnitude threshold, the o-mainshock often coincides with the mainshock of the cluster or with the first mainshock if it is a swarm. In addition, we will use the term "SSLE" instead of "strongest aftershock", to avoid confusion between aftershocks, that can be estimated only a-posteriori as the earthquakes following the mainshock, and the SLEs, that can also include the mainshock itself, if the o-mainshock is a strong foreshock.

Båth (1965) proposed one of the first systematic empirical aftershock studies in which he claimed that *Dm* is approximately constant and equal to 1.2 (the so-called Båth law). However, several successive studies have outlined the large variance of *Dm* (e.g., Shcherbakov and Turcotte, 2004). Shcherbakov and Turcotte (2004, 2014) and Shcherbakov et al. (2018 and 2019) used early information on aftershocks that had already occurred to extrapolate the expected magnitude of the SSLE. In particular, Shcherbakov and Turcotte (2004) estimated it from the b value of the Gutenberg-Richter distribution, while Shcherbakov (2014) and Shcherbakov et al. (2018 and 2019) applied Bayesian analysis and extreme value statistics and used the Omori-Utsu law (Shcherbakov, 2014; Shcherbakov et al., 2018) or ETAS (Shcherbakov et al., 2019) to estimate the rate of aftershocks.

Many studies on *Dm* are based on b-value; Helmstetter and Sornette (2003) showed a dependence of *Dm* on the inverse of the b-value. Gulia and Wiemer (2019) also proposed a method for large SSLEs based on the

* Corresponding author.
E-mail addresses: sgentili@inogs.it (S. Gentili), rita.digiovambattista@ingv.it (R. Di Giovambattista).

https://doi.org/10.1016/j.pepi.2022.106879
Received 21 July 2021; Received in revised form 7 January 2022; Accepted 21 April 2022
Available online 27 April 2022

b value. In particular, they were interested in negative values of *Dm*, i.e., cases in which the SSLE magnitude is greater than that of the o-mainshock. They observed, in clusters with negative *Dm*, a decrease in the b value before the SSLE. In their paper, therefore, small values of b correspond to low values of *Dm*, in contradiction with the Helmstetter and Sornette (2003) model. Brodsky, (2019)outlined some limits of the Gulia and Wiemer (2019) method, such as the small statistical database for clusters with high SSLE and some degree of arbitrariness in b-value estimation. Dascher-Cousineau et al. (2020) outlined the instability of the results of the method on a test on the 2019 Ridgecrest cluster, in which their results and Gulia et al.'s (2020) results were different, but Gulia and Wiemer (2021) show some differences between their data analysis and Dascher-Cousineau et al. (2020) one. Van der Elst (2020) developed a more stable estimator of the b value, the b-positive, partially confirming the results of Gulia and Wiemer (2019, 2020), but he claimed that unbiased b-value changes may be too subtle to be used in a real-time earthquake alarm system. Although the debate on the method is still ongoing, the main problem in the applicability of the method for real clusters of seismicity is the need, for correct estimation of the b-value, for a huge number of earthquake records, which are often not available in seismic catalogs.

Some studies have investigated the dependence of *Dm* on mainshock characteristics. In particular, the dependence of *Dm* on the mainshock magnitude has been investigated by several authors (e.g., Båth, 1965, Vere-Jones, 1969, Tahir et al., 2012, Helmstetter and Sornette, 2003, Gentili and Di Giovambattista, 2017, 2020); other studies have been based on mainshock focal mechanisms (Rodríguez-Pérez and Zúñiga, 2016; Tahir et al., 2012; Gentili and Di Giovambattista, 2017); other works have investigated the depth of the mainshock (Persh and Houston, 2004; Gentili and Di Giovambattista, 2017). The results are region-dependent and may also be very different from one region to the other. In this paper, we investigated the dependence on mainshock parameters for the southern California clusters in Section 3.3.

Another approach to SSLE forecasting is based on machine learning applied to seismicity following the mainshock. Vorobieva and Panza (1993) and Vorobieva (1999) proposed a set of seismicity-based functions (features) representing premonitory phenomena. Features, also referred to as attributes, variables, fields, or predictors, are the items of data that are used in machine learning. They correspond to individual measurable properties or characteristics of a phenomenon.

In seismicity analysis they may be functions of the number of earthquakes in a given magnitude range, or of their energy, or of their focal mechanism, etc. Vorobieva and Panza (1993) and Vorobieva (1999) divided the clusters into two categories (classes): "type A" if, given a mainshock of magnitude Mm, the SSLE in the cluster has a magnitude Ma $\geq$ Mm-1; "type B" otherwise. For each feature, they found a threshold so that if the feature is over the threshold, the feature votes for a type A cluster. The final forecasting is obtained by a voting procedure on the 7 or 8 independent features. The main drawback of the Vorobieva and Panza (1993) and Vorobieva (1999) methods is that forecasting requires a long time span T after the mainshock (10–40 days). In many cases (e.g., 71% in Italy), the SSLE occurs before the end of T, and the method cannot be applied. In addition, the voting procedure supplies a binary classification and not the probability of having an A cluster; a combination of precursors does not always supply more useful forecasting than a simple precursor (Grandori et al., 1984) because lower performance of one precursor can decrease the quality of results.

The results in literature indicate the need to develop a fast forecasting method that can be applied for a large number of clusters, shortly after the mainshock. Seismic catalogs provide data in a short time, and machine learning methods allow relevant information to be extracted quickly and reproducibly. In Gentili and Di Giovambattista (2017) we proposed a decision-tree-based, machine learning approach to forecast if the clusters are of type A or B. Our analysis was performed at different time intervals Ti, ranging from 6 h to 7 days after the mainshock, to simulate the increasing knowledge during the time after the o-mainshock. The features adopted were some new original features and some features existing in the literature (Kossobokov et al., 1999; Vorobieva, 1999; Di Giovambattista and Tyupkin, 2002; Gentili and Bressan, 2008). The algorithm needed a completeness magnitude for the clusters of Mm-3, where Mm is the o-mainshock magnitude, like in Vorobieva (1999). To avoid problems related to the voting procedure, we provided a probability to have an A cluster that was the weighted mean of the binary classification of each feature, using the Informedness of each feature evaluated during the training phase as weight.

In Gentili and Di Giovambattista (2020), we proposed an improvement of the previous machine learning method, named NESTORE (Next STrOng Related Earthquake). The algorithm has two different instances, named NESTORE_M2 and NESTORE_M3, requiring a completeness magnitude of Mm-2 and Mm-3, respectively. The reduction of the difference in magnitude between Mm and Mc allowed us to analyze a larger dataset. Therefore, the results were more stable for NESTORE_M2, despite the smaller number of features used. The estimation of the probability of having an A cluster was further improved, evaluating the probability for each feature from the percentage of A clusters under and over the threshold in the training set. These probabilities were combined using a Bayesian approach (see Section 2.3 for more details).

The main idea in applying NESTORE to different parts of the world is to test a set of features that have been successful elsewhere and take only those that provide reliable information for the area under analysis. Applying it to a different dataset, with different seismicity characteristics and catalog quality, also allows the training procedure to be tested and improved in order to develop a more robust machine learning procedure that can be successfully applied in previously analyzed regions to improve the performance of the classifier.

In this paper, we applied a revised version of the NESTORE_M2 method to California seismicity; the California catalogs are particularly useful for testing the NESTORE method due to the availability of a relatively large number of clusters in a tectonically homogeneous region. In addition, we propose a new, more robust approach, called rNESTORE, that can reduce overfitting problems (see Section 4.3).

2. NESTORE algorithm

NESTORE is a multiparameter machine learning approach analyzing the evolution of seismicity at different time steps. Its main objective is estimating, after a strong earthquake, the probability that the type of the ongoing cluster is A. Machine learning algorithms are usually trained and tested on hundreds (or thousands) of data. NESTORE is specifically tuned for seismicity analysis problems, with few data – the number of available clusters is often on the order of tens - and the characteristics of the seismicity change over time. To take into account the low number of available examples, NESTORE considers a set of features separately, and only after the training it does merge the best features classification. In addition, it uses one-level decision trees (in other words, a simple threshold) to discriminate between the classes. In detail, we used a linear classification decision tree implemented in the "fitctree" function in Matlab. This approach simplifies the problem, reducing overfitting due to the small training set. The analysis is performed on increasing time intervals, starting a short time after the mainshock and ending at the times $T_1 \ldots T_m$ (we will call these intervals $T_1 \ldots T_m$), to simulate the evolution of seismicity over time. Note that due to the particular problem addressed, the number of A clusters decreases for longer time intervals. In fact, as the time after the o-mainshock increases, one or more SLEs with magnitude Ma $\geq$ Mm-1 may be included in the time interval T_i. In this case, the cluster has already been shown to be a type-A cluster; therefore, no further forecasting can be done, and the cluster analysis stops at step T_{i-1}.

NESTORE is composed of two main modules: the *training module*, in which the parameters are adapted by a training procedure to the analyzed seismicity, and the *classification module*, in which new

examples are automatically classified in class A or B by using the parameters tuned in the previous module. In the following, we will describe the features adopted and the two modules of NESTORE. Note that the general architecture of the algorithm is independent from the particular features used, which can be tuned depending on the characteristics of the analyzed area, both in terms of seismicity and of available data.

2.1. Analyzed features

In this version of the algorithm, we analyzed earthquakes with magnitude M ≥ Mm-2 in the selected intervals Ti for the clusters with Mm ≥ 4 (see section 3.3). We adopted the features of the instances NESTORE_M2 in the paper by Gentili and Di Giovambattista (2020) and one of the features of NESTORE_M3, modified to be used in this range of magnitude. A detailed list of references on where these features were first proposed in literature is available in Gentili and Di Giovambattista (2017). To take into account the increase in the completeness magnitude Mc immediately after stronger earthquakes of magnitude Mm, we applied the Helmstetter et al. (2006) equation for Mc estimation in time for California:

$$Mc = Mm - 4.5 - 0.75 log_{10}(t) \tag{1}$$

where t is the time expressed in days. The equation is valid for Mc ≥ 2.

Rearranging the eq. (1) we obtain

$$t = 10^{\frac{Mm-Mc-4.5}{0.75}} \tag{2}$$

For this NESTORE version, to avoid incurring problems of incompleteness of the catalog, we need a difference Mm-Mc ≥ 2. Substituting 2 in place of Mm-Mc in eq. (2), we obtain $t = 4.6 \bullet 10^{-4}$ days corresponding to 0.7 min. This is the shortest amount of time after the mainshock for which we can assume that the catalog is complete and NESTORE can be applied. We set a starting time s_1 for the analysis of one minute after the mainshock. The condition Mc ≥ 2 is satisfied because we set the minimum Mm to 4; therefore, we are not interested in earthquakes of magnitudes less than 2.

The features are evaluated on the events in the cluster with magnitude ≥Mm-2 inside a time interval [s_1, Ti] for Ti ≤ s_2. The features used in this version of NESTORE are as follows:

1. *S*: Normalized event source area

$$S(i) = \sum_i 10^{(m_i - M_m)} \tag{3}$$

where m_i is the magnitude of the i^{th} event.

2. *Z*: Linear concentration of events

$$Z(i) = \frac{mean\left(10^{0.69 m_i - 3.22}\right)}{mean\left(r_{ij}\right)} \tag{4}$$

where r_{ij} is the distance between the generic i^{th} and j^{th} events.

3. *Q*: Normalized radiated energy

$$Q(i) = \frac{\sum_i E_i}{E_m} \tag{5}$$

where E_m is the energy of the mainshock and E_i is the energy of the i^{th} event. The energy E is evaluated from the magnitude by the Gutenberg and Richter, (1956) equation:

$$Log_{10}(E) = \frac{3}{2} M + 4.8$$

4. *SLCum*: Cumulative deviation of S from the long-term trend on increasing length windows

$$SLCum(i) = \sum_i abs\left[S(t_i) - S(t_{i-1}) \frac{i \bullet dt}{(i-1) \bullet dt}\right] \tag{6}$$

where $t_i = s_1 + i \bullet dt$, $S(t_i)$ is *S* calculated on the time interval $[s_1, t_i]$ and dt = 6 h.

5. *SLCum2*: Cumulative deviation of *S* from the long-term trend on sliding windows

$$SLCum2(i) = \sum_i abs\Big[S([s_1 + (i-1) \bullet dt\,, s_1 + i \bullet dt]) - S([s_1 + (i-1) \bullet dt\,, s_1 + (i-1) \bullet dt + d\tau]) \frac{dt}{d\tau}\Big] \tag{7}$$

where *S*[*a*, *b*] is *S* calculated over the generic time interval [*a*, *b*] and $d\tau$ = 1 h. In contrast to *SLCum*, the window does not start at a fixed time close to the mainshock origin time.

6. *QLCum*: Cumulative deviation of *Q* from a long-term trend on increasing length windows

$$QLCum(i) = \sum_i abs\left[Q(t_i) - Q(t_{i-1}) \frac{i \bullet dt}{(i-1) \bullet dt}\right] \tag{8}$$

where $t_i = s_1 + i \bullet dt$, and $Q(t_i)$ is calculated on the time interval $[s_1, t_i]$.

7. *QLCum2*: Cumulative deviation of *Q* from the long-term trend on sliding windows

$$QLCum2(i) = \sum_i abs\Big[Q([s_1 + (i-1) \bullet dt\,, s_1 + i \bullet dt]) - Q([s_1 + (i-1) \bullet dt\,, s_1 + (i-1) \bullet dt + d\tau]) \frac{dt}{d\tau}\Big] \tag{9}$$

where *Q*[*a*, *b*] is *Q* calculated on the generic interval [*a*, *b*].

8. V_m: Cumulative variation of magnitude from event to event

$$V_m(i) = \sum_i |m_i - m_{i-1}| \tag{10}$$

This feature has been modified compared with the old versions of NESTORE because we now investigate earthquakes with magnitudes Ma ≥ Mm-2 and not Ma ≥ Mm-3, as in the previous versions.

9. N2: Number of events

According to Gentili and Di Giovambattista (2017 and 2020), we set the first analysis 6 h (0.25 days) after the mainshock to balance the need to have as many clusters as possible for our analysis (many SLEs occur in the first hours after the mainshock) and the need to have a sufficiently good statistic on the evolution of seismicity. The time intervals {T_1… T_{10}} start at time s_1 after the mainshock, and they end at the following times: {0.25, 0.50, 0.75, 1, 2, 3, 4, 5, 6, 7} days, as in Gentili and Di Giovambattista (2017).

S, Z, Q and N2 features are evaluated for all the time intervals. QLCum, SLcum and Vm, conversely, due to their definition, require an earlier time step, and are analyzed in the [T_2, T_{10}] time intervals. QLCum2 and SLcum2 can be evaluated in the interval T_2, but they are coincident with QLCum and SLcum; for this reason, we evaluate them only in the intervals [T_3, T_{10}].

The main idea for using these features is to detect a change in earthquakes flow in terms of increased intensity and irregularity in space, time and magnitude. Keilis-Borok and Rotwain, (1990) and

Keilis-Borok and Kossobokov, (1990) observed a similar pattern before strong mainshocks, while Vorobieva (1999) and Vorobieva and Panza (1993) before stronger SLEs. This change has been interpreted as a symptom of instability of a nonlinear system corresponding to earthquake-generating faults (Vorobieva, 1999).

2.2. NESTORE training

NESTORE is a supervised training algorithm. In other words, it takes in input a set of examples, together with the desired output class (the training set). Fig. 1 shows the flowchart of the training of the algorithm for one feature.

The algorithm is divided into several blocks: feature extraction, decision tree training, good interval identification, inheritance and validation.

2.2.1. Feature extraction block

The algorithm extracts the required feature from the input training clusters. The procedure is repeated for each time interval T_1, …T_m after the mainshock. The feature extraction module outputs, for each time period T_i, a vector with the value of the feature for each cluster of the training set in the interval T_i: $f(T_i)$. In addition, it supplies the class of each cluster (not shown in the figure, for simplicity).

2.2.2. Decision tree training block

The training is performed by binary decision trees. The tree depth is set to 1 (one decision node with two leaves) to avoid any overfitting of the data. A one node decision tree is simply a threshold. For the features adopted, the output class is A if the feature is greater than or equal to the threshold and B otherwise. If no tree is found solving the problem, the value of the threshold is set to NaN (Not a Number).

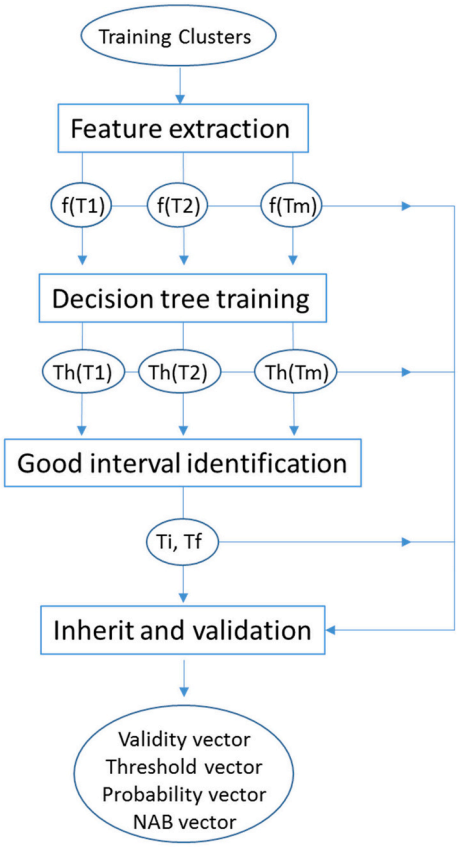

Fig. 1. Flowchart of the NESTORE algorithm for one feature training.

2.2.3. Good interval identification block

Even when a threshold is identified by the previous block, the performance of the decision tree may be poor. For this reason, the algorithm evaluates the feature performance by the LOO (Leave one Out) method applied to the training set. In particular, the following performance evaluators are estimated: Accuracy, Precision, Recall and Informedness; for further details on the parameters, see Gentili and Di Giovambattista (2020). The first three evaluators are defined between 0 and 1 (1 is the best and 0 the worst), and the last evaluator is defined between −1 and 1 (1 is the best and − 1 the worst). The typical trend of these performance evaluators in NESTORE applications is an increase in the evaluator value as the observation time Ti increases, until a maximum, followed by a constant value or a decrease if the observation time is longer. The algorithm selects the intervals Ti that satisfy the following conditions:

1. Accuracy, Precision, and Recall should be >0.5.
2. Accuracy should be greater than or equal to the one we can obtain by a constant response corresponding to the most populated class (B class).
3. Informedness should be >0.

The first and last conditions guarantee good performance of the feature, eliminating the less stable feature from the analysis. The second condition is accuracy, a measure of the percentage of correct classifications. It is used to avoid overestimations of the feature performance in the case of unbalanced classes.

The "good interval" starts with the first Ti satisfying the previous conditions (1–3) and ends with the Ti corresponding to the maximum of Informedness ($Ti = s_2$). Informedness is a measure of how informed the classifier is about the class (for further details, see Gentili and Di Giovambattista, 2020).

2.2.4. Inherit and validation block

The last block is modified with respect to previous versions of the algorithm by Gentili and Di Giovambattista (2020). In previous versions, once a feature reached the maximum of Informedness, the threshold was fixed to the current value. Analyses at longer time periods "inherited" the feature values and the threshold of the interval s_2. In other words, if the maximum of Informedness was reached at a given Ti, for a given threshold Thi, for $Tj > Ti$ the features of each cluster were set to the value they had for $T = Ti$ and the thresholds were set to Thi. This allowed feature with good-performance to be used even for times longer than when performance was the best. Sometimes we observed, as Ti increased, a drop in performance for inherited features and thresholds, but we ascribed it to fewer clusters being available for longer times and thus to lower test reliability. Further tests demonstrated that this is not always correct because there is a selection effect on the clusters. In other words, for some features, the type A clusters with later SSLEs belong to a different population with respect to most A clusters. As time passes, the percentage of these clusters increases in the dataset, causing a decrease in feature performance.

For this reason, in this new version of NESTORE, the algorithm again verifies the performances by applying an LOO method to the training set. For all inherited thresholds and features, it checks whether the percentage of A correctly classified (hit rate) is greater than the percentage of B incorrectly classified as A (false alarm rate). If it does not occur for some interval T_i, T_i is eliminated from the set of intervals for which the feature can be evaluated.

The training algorithm outputs the following vectors for each feature:

- Validity vector: A vector of the values of T_i for which the feature can be used.
- Threshold vector: The values of the thresholds for each T_i.
- Probability vector: The probability of being an A cluster estimated as the ratio between the number of A clusters and the total number of

clusters. The probability is estimated separately under (p_u) and over (p_o) the threshold.
- NAB vector: The numbers N(A) and N(B) of clusters A and B in the training sets for each Ti.

2.3. NESTORE classification

Fig. 2 shows the classification stage for generic Ti.

Given the set of clusters to be classified at a given time period Ti, the features are extracted. For each feature, the validity vector, threshold vector, and probability vector estimated in the training phase are used to classify the input. If a feature f is reliable for that time period, its value is compared with the corresponding threshold Th_f. Instead of a simple binary classification in the two classes A or B (respectively, if the $f \geq Th_f$ or $f < Th_f$), the probability vector is used. This vector is a key point in classification procedure, because it provides important information on how to combine the output of different feature classifiers in the final classification step. The probability of being an A cluster are calculated separately under and over the threshold. While this probability would ideally be 0 for $f < Th_f$ and 1 for $f \geq Th_f$, some features provide very good performances above (or below) the threshold, where all clusters belong to the same class, while the performances are poorer below (or above) the threshold, where the classes are more mixed: in these cases the probability can be close to 0.5. In order to score the features depending on their performances, the feature classification block supplies the probability $p_n = P(A|D_n)$, that is the probability of having a type A cluster given a value D_n of the n^{th} feature: p_n corresponds to p_u if the feature is under the threshold and p_o if it is over.

The overall probability of having an A cluster is evaluated by using a Bayesian approach (see Gentili and Di Giovambattista, 2020) to combine different feature-based probabilities:

$$P(A|D_1 \ldots D_N) = \frac{[N(B)]^{N-1} \prod_{n=1}^{N} p_n}{[N(B)]^{N-1} \prod_{n=1}^{N} p_n + [N(A)]^{N-1} \prod_{n=1}^{N} (1-p_n)} \quad (11)$$

where N(A) and N(B) are the number of A and B clusters in the training set, respectively, for that particular time interval, stored in the NAB vector.

By using this approach, the number of A and B clusters is taken into account. This is very important for unbalanced classes, as in our case (see section 3.3).

3. Earthquake catalog and cluster detection

3.1. Catalog and selected area

We adopted the SCSN earthquake catalog (Hutton et al., 2010) available from 1932 to the present, where data are listed in M_L units. During these almost 90 years, the sensitivity and accuracy of location and magnitude determination of the catalog have changed; changes are related to the network geometry, instrument characteristics, and monitoring and processing strategies over time. In Hutton et al. (2010) paper, several periods with different catalog characteristics were identified. For our analysis, we selected data from 1981 to the present (corresponding to the last two periods in Hutton et al.'s paper) due to the large increase in seismic stations in 1981, which produced an increase in sensitivity and accuracy in parameter determination. Several authors have pointed out the differences in the quality of the catalog, not only in time but also in space. Unsurprisingly, for example, offshore data have a higher completeness magnitude (Nandan et al., 2019), i.e., the network is less sensitive offshore. To estimate the completeness magnitude of the analyzed clusters, we proceeded in two ways: for clusters with a number of events greater or equal to threshold Th_{Mc}, we evaluated the

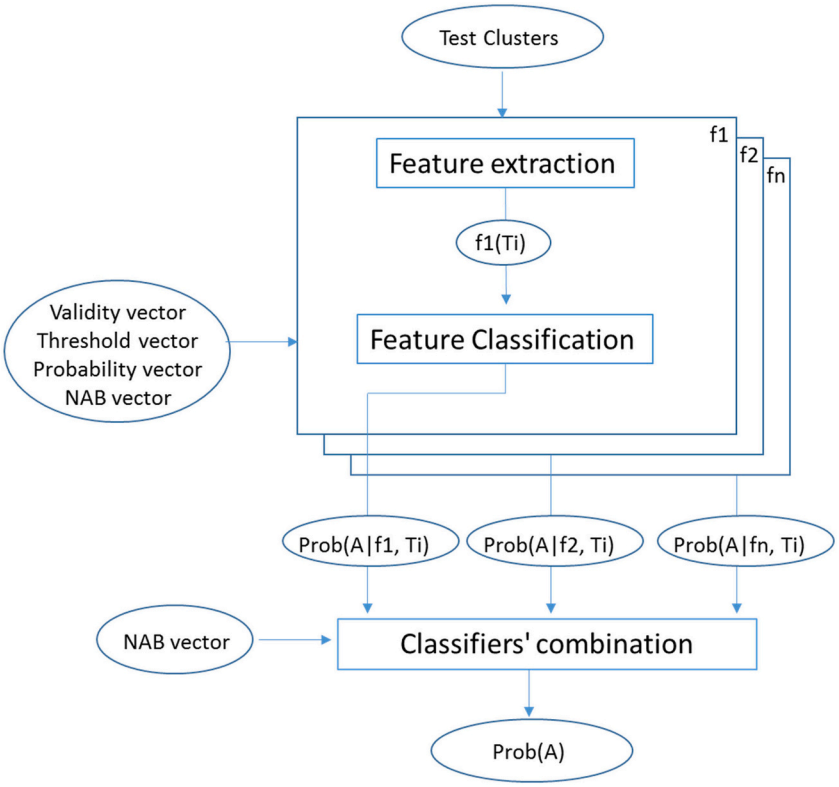

Fig. 2. Classification for a given interval T_i.

completeness magnitude by the maximum curvature method, to which, as recommended by Woessner and Wiemer (2005), we added a correction factor of 0.2. For a lower number of event clusters, we set a maximum completeness magnitude considering the results in the literature. In particular, we considered the completeness maps by Hutton et al. (2010) and Lippiello et al. (2012) and the estimates of Godano (2016) and Nandan et al. (2017).

In addition, we analyzed the completeness of southern California in the years from 1981 to 2007, thanks to the data service available at the CompletenessWeb website (http://www.completenessweb.gfz-potsdam.de/doku.php). Based on CompletenessWeb outputs and considering the lower quality of offshore data (Nandan et al., 2019), we selected the area shown in Fig. 3 (in green) for our analysis. Inside the area, we set a conservative value of Mc = 3 when the completeness magnitude cannot be evaluated due to too few earthquakes in the cluster. In Fig. 3, red points correspond to the o-mainshocks of the A clusters, and blue points correspond to the o-mainshocks of the B clusters.

With a completeness magnitude that is not uniform across time and space, we used the local completeness magnitude whenever possible instead of the higher regional completeness magnitude. By doing so, we obtained a higher number of clusters that met our completeness requirements (see introduction) and thus more reliable results in cluster classification. On the one hand, a small Th_{Mc} would have allowed us to perform the analysis based on the local completeness magnitude of a large number of clusters, even the smallest ones. On the other hand, too low a Th_{Mc} would have caused inaccurate completeness estimates due to too few events in the cluster. In order to achieve a compromise between these two different requirements, we chose a threshold $Th_{Mc} = 80$ events by a trial and error approach.

3.2. Cluster identification

Cluster identification is a non-unique procedure, and several methods exist in the literature, supplying different results. We will cite here the most popular methods that have been applied to California. The simplest methods available are window-based methods, in which the cluster is identified as all events within a time and space window around the mainshock, whose extent is generally a function of the mainshock magnitude. Two examples of the functions adopted in California are those of Gardner and Knopoff (1974) and Kagan (2002). Reasenberg (1985) proposed a popular link-based method in which the cluster is seen as a set of events connected by links in space and time, whose extent is a function (also) of the seismic moment of the mainshock and of the last earthquake. Other methods are based on the probability of earthquakes belonging or not belonging to a cluster; one example is the ETAS-based stochastic model of Zhuang et al. (2002). More recently, Zaliapin and Ben-Zion (2013) proposed a cluster identification method based on the analysis of space-time and energy distribution of the entire catalog (nearest-neighbor method). In this paper, we do not need a cluster identification method only but also a method that, given an o-mainshock, forecasts if an SSLE with magnitude ≥Mm-1 will follow, supplying the space and time interval in which it will occur. For this reason, we decided to adopt a window-based method, in which the window size is a function of the o-mainshock magnitude. For the time window, we used the Gardner and Knopoff functions:

$$t = 10^{0.032 \bullet Mm + 2.7389} \text{ if } M \geq 6.5 \quad (12)$$

$$t = 10^{0.5409 \bullet Mm - 0.547} \text{ else}$$

The time window is not particularly important in our analysis because the SSLE often occurs from a few hours to a few months after the mainshock; therefore, the cluster usually ends a long time after the occurrence of the SSLE.

For the space window, we compared the Gardner and Knopoff function

$$d = 10^{0.1238 \bullet Mm + 0.983} \quad (13)$$

and to the Kagan (2002) function

$$d = 0.02 \bullet 10^{0.5 \bullet Mm} \quad (14)$$

where d is the distance from the mainshock (in our case, the o-mainshock) and Mm is its magnitude. In the range of magnitude 4–7.5, where the o-mainshocks occur, the Kagan function supplies a radius of the cluster generally smaller than that of Gardner and Knopoff (15 times smaller for Mm = 4). To choose the best function, we compared the circular area of the clusters obtained by the two methods with the one obtained by Zaliapin and Ben-Zion (2013) using the nearest-neighbor (NN) method; the NN method is not a window-based method and automatically adapts to the space-time characteristics of the area. Fig. 4 shows an adaptation from Zaliapin and Ben-Zion (2013); in particular, the figure shows the area of the clusters as a function of the mainshock magnitude found by the NN method in California. In the image, we

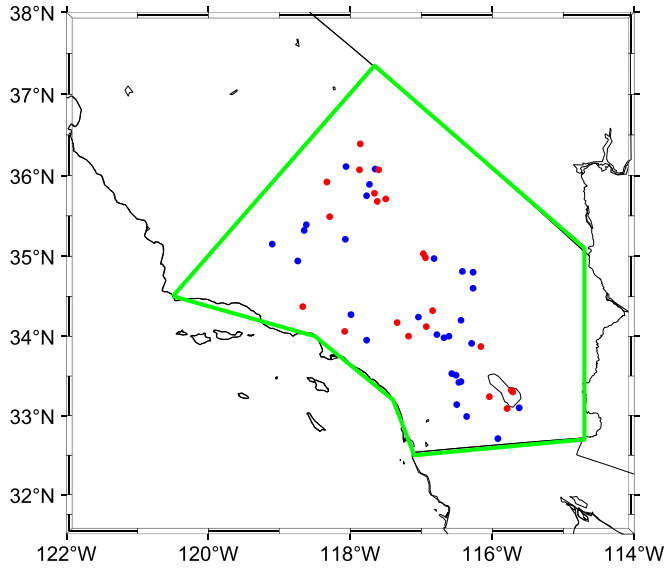

Fig. 3. Area selected for the analysis. Blue = B class red = A class. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

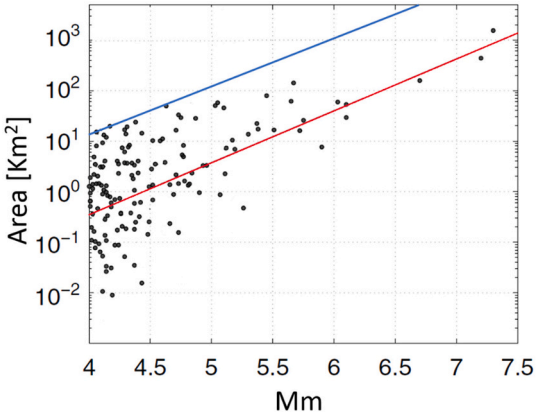

Fig. 4. Aftershock area for clusters in California according to Zaliapin and Ben-Zion (2013). Red line: slope of the distribution by Zaliapin and Ben-Zion (2013). Blueline: Kagan (2002) equation. Circles: aftershock area according to Zaliapin and Ben-Zion's (2013) paper. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

superimposed the circular area calculated by using the radius obtained by the Kagan (2002) function (blue line). The Kagan function coincides with an upper limit of the Zaliapin and Ben Zion distribution of the cluster areas.

In addition, both in the Zaliapin and Ben-Zion (2013) results and in Kagan (2002), the area of the cluster region scales with a magnitude of $A \approx 10^{\gamma Mm}$ with $\gamma = 1$. For this reason, we adopted the Kagan (2002) equation for space-window evaluation.

3.3. Cluster dataset for NESTORE analysis

The choice of the minimum magnitude for cluster analysis is challenging. On the one hand, it is important to maintain this magnitude as small as possible to enlarge the available dataset. On the other hand, by selecting a magnitude that is too small, a large amount of noisy background data may be added to the dataset. Considering that the NESTORE algorithm needs a completeness magnitude at least equal to Mm-2 and that most of the analyzed area has a completeness magnitude equal to 2, we selected as o-mainshocks all earthquakes with magnitudes ≥4 not preceded by other earthquakes with magnitudes ≥4. We eliminated from our analysis the clusters with *Dm* in the interval [0.8–1.2], where due to uncertainty in magnitude estimation, the cluster type is uncertain. The algorithm for cluster identification extracted 79 (50 A and 29 B) clusters inside the selected area. After 6 h (the end of T1), we have a total of 50 clusters, of which 21 are type A and 29 are type B. Of the 50 clusters, in 45 cases, the o-mainshock coincides with the mainshock, while in 5, it is a foreshock of a following mainshock. Similar to Gentili and Di Giovambattista (2017 and 2020), our analysis is performed every 6 h on the first day and every day for the first week after the mainshock. During this period, while the number of B clusters remains unchanged, the number of A clusters decreases from 21 to 7 because we do not analyze the clusters at times Ti, in which the SSLE has already occurred. Fig. 5a shows the number of A and B clusters in time: the dataset for $T =$ 6 h is enough balanced and becomes unbalanced for longer times (for $T \geq 5$, the number of B clusters is 4 times the number of A clusters).

The total number of recorded events in the clusters in California depends on mainshock magnitude, on the cluster productivity and on the completeness magnitude in the area. In our database it ranges from 1 to over 40,000 events. In the magnitude range Mm-2, useful for the analysis, for clusters with completeness magnitude Mc ≤Mm-2, it ranges from 0 to 22 events in the analyzed time period.

Fig. 5b, c and d show the magnitude of the o-mainshock for each

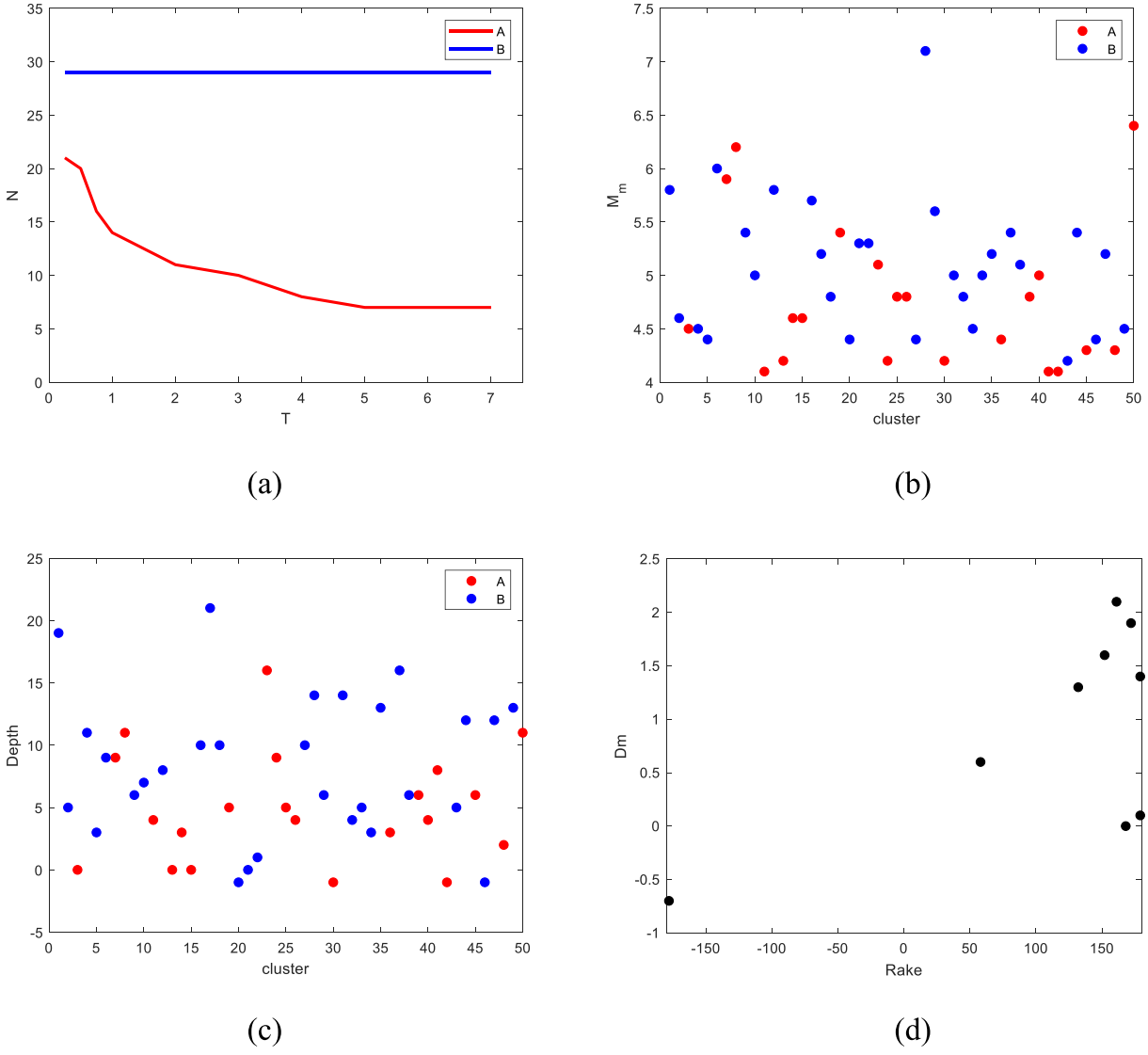

Fig. 5. Blue: B clusters; Red A clusters (a) number of A and B clusters for different time intervals T_i (b) Mm of the analyzed clusters. (c) Depth of the cluster mainshock (d) *Dm* vs mainshock rake angle. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

cluster, its depth and the rake angle, respectively. The rake angles were obtained for the o-mainshocks for which focal mechanisms were available in the SCSN Focal Mechanism Catalog (Hutton et al., 2010). For California, we did not find any relationship between the o-mainshock magnitude or depth and the cluster class (see Fig. 5b and c). The rake data show a general increase in *Dm* with the rake angle, but the data are too few to be statistically relevant.

4. Results

To test NESTORE on an independent test set, we applied Leave One Out (LOO) method to the 50 cluster dataset (see section 4.1). The performances are evaluated on all data, testing each cluster after training obtained by all other clusters. The performances of well-known clusters in California are shown in detail (section 4.2). The LOO method results allow us to evaluate the most successful features and their stability. In section 4.3, we show how we integrated the results on different datasets to obtain for each feature a more accurate threshold and a more reliable domain of applicability (rNESTORE). Finally, section 4.4 shows the test of both NESTORE and rNESTORE on an independent database.

4.1. Leave One Out method

The LOO method is a particular case of K-fold cross validation in which each element of the dataset is tested separately by a classifier trained by all other elements. In this way, the training set dimensions are maximized. This approach is very useful for small dataset testing. Fig. 6 shows the performance of NESTORE obtained by the LOO method through the Receiving Operating Characteristics (ROC) (Egan, 1975; Sweets et al., 2000; Fawcett, 2006) and Precision-Recall (PR) (Fawcett, 2004; Davis and Goadrich, 2006; Saito and Rehmsmeier, 2015) graphs. Both graphs are used to illustrate the performance of a binary classifier,

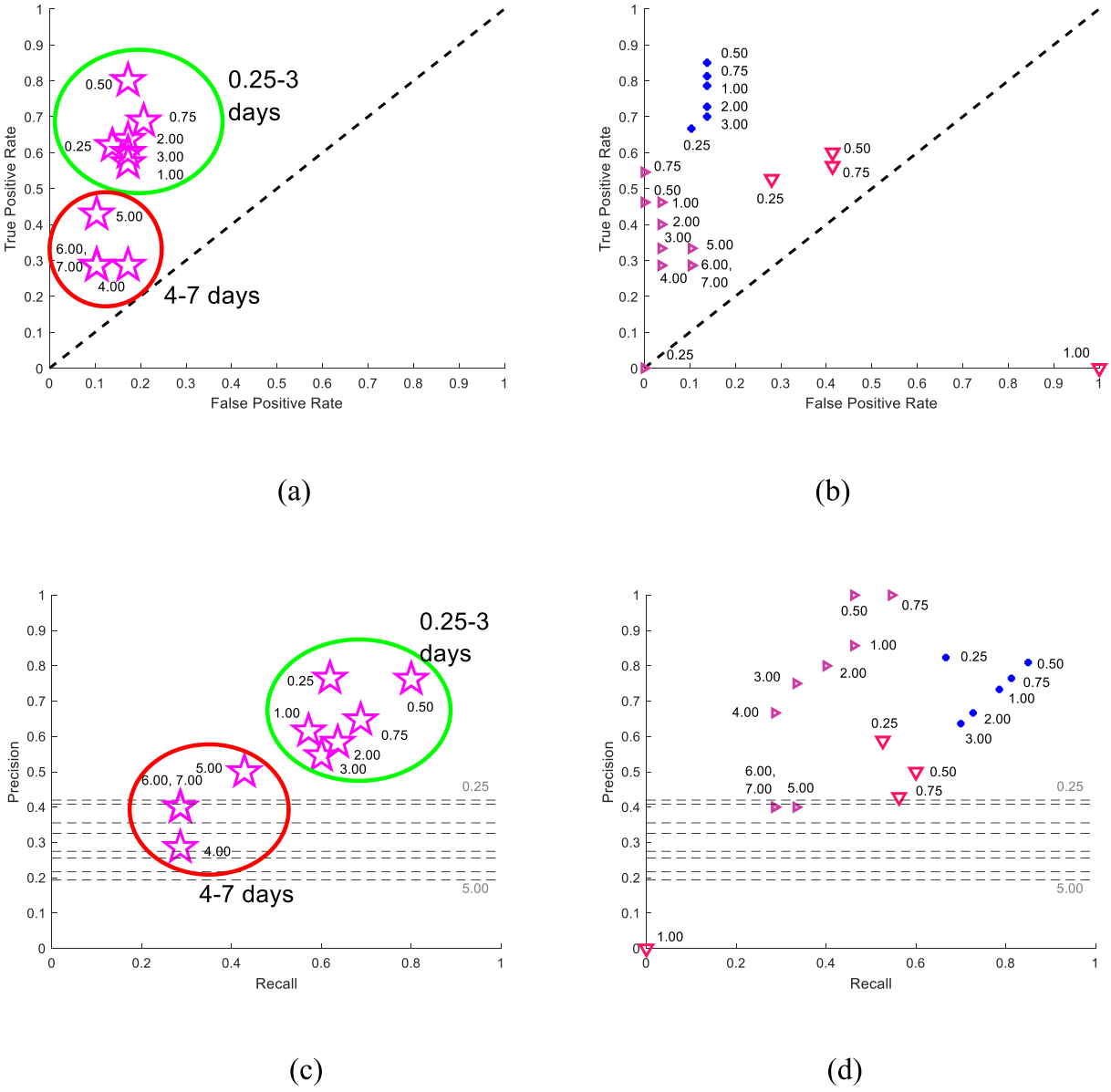

Fig. 6. NESTORE performances for different Ti values using the LOO method. Time periods Ti are listed close to the corresponding star. Magenta stars: NESTORE performances; blue dots: N2; violet small triangles: Q; and red larger triangles: Z. (a) ROC graph for NESTORE; (b) ROC graph for selected features; dashed line correspond to random classifier; red and green circles outline performances in different time periods (c) Precision-Recall (PR) graph for NESTORE (b) PR graph for selected features; dashed lines: random classifiers for different time periods: from up to bottom 0.25, 0.5, 0.75, 1, 2, 3, 4, 5 days; longer time period coincide with the ones of 5 days. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

where one class is considered positive (in our case, we set class A as positive) and the other negative. The ROC graph shows the True Positive Rate (TPR - in our case, the ratio of the number of A clusters classified as A and the total number of A clusters) versus the False Positive Rate (FPR - in our case, the ratio between the B incorrectly classified as A and the total number of B clusters). In other words, it shows relative trade-offs between benefits (true positives) and costs (false positives). The diagonal of the graph (see Fig. 6a, b) represents random guessing. Any classifier that appears in the lower right triangle performs worse than random guessing and should be discarded. Conversely, a classifier in the upper left triangle supplies reliable results; the closer the classifier is to the upper left corner, the better its performance. The upper left corner corresponds to the ideal classifier, in which no B is erroneously classified as A, while all As are correctly forecasted. The use of ROC graphs for unbalanced datasets is debated; ROC graphs are insensitive to changes in class distribution because both TPR and FPR are obtained by analyzing only the positive class (class A in our case) or the negative class (our class B), respectively, and are independent of the ratio of the number of items in the two classes. While some authors consider this an advantage (e.g., Fawcett, 2004) other authors (e.g., Saito and Rehmsmeier, 2015) disagree, because large changes in the number of false positive FPs (in our case, B misclassified as A) may not affect the FPR if the number of correctly classified elements in the negative class is large. For this reason, they suggest using the Precision-Recall plot, where instead of FPR, Precision is shown. Precision is defined as the ratio of the number of correctly classified positives to the total number of items classified as positives, or in our case, the ratio of the number of clusters A to the number of clusters classified as A. Since the number of clusters classified as A includes FP, the position on the Precision-Recall graph depends on the relative abundance of examples in each class (class skew). TPR and Recall are synonyms but, for consistency with previous literature, we will denote them as FPR in ROC graphs and as Recall in PR graphs. Random guessing in the Precision-Recall graph (see Fig. 6c, d) depends on the relative abundance of classes and is determined by the fraction of positives in the dataset; in our case, the number of As decreases for longer time periods and thus the random guessing line (dashed lines in Fig. 6c, d), parallel to the x-axis, has a y-intercept that decreases as T increases (the 0.25- and 5-day random guessing lines are specified in gray in Fig. 6c, d). A classifier above the random guessing line provides reliable results; the closer the classifier is to the upper right corner, the better its performance. For the PR graph, the upper right corner corresponds to the ideal classifier, in which no Bs are misclassified as As, while all As are correctly predicted.

In Fig. 6a and c, magenta stars correspond to the position in the ROC diagram of the NESTORE classification for each time period T.

For all time periods, NESTORE's performance is always in the upper left triangle for ROC graphs, and above the corresponding random hypothesis line in PR graphs, i.e., in areas corresponding to reliable classifiers. Comparing different T_i performances, the TPR (i.e. the Recall in PR graphs) decreases for $T > 3$ days. This may be partially due to the unbalancing of the classes in a small dataset: for $T > 3$ days, the number of A clusters becomes less than 1/3 of the B clusters. The small dataset causes training based on a few A examples: 7 or 8 (depending on Ti) if a B cluster is tested and 6–7 if the test is performed on an A cluster. Whereas the TPR, based on A clusters, the performances generally decrease as the number of A clusters in the database decreases, the FPR, based on B clusters, is generally not affected by changes in the training set and fluctuates from 0.1 to 0.2. The PR graph shows a decrease in Precision for $T > 3$ days related to the increase in B clusters classified as A compared to correctly classified A clusters. In other words, due to the higher number of B clusters in the training set, the decision trees tend to classify most of the clusters as B, even though they are A. In Fig. 6b and d the details for some features are shown. We selected these features because they are representative of different performances of features in our analysis. N2 (blue dots) is the higher performance feature, with TPR ranging from 0.66 to 0.85, the Precision ranging from 0.63 to 0.82 and the FPR ranging from 0.10 to 0.14. Q, accordingly with the results of LOO method, is a low-hit rate and low false alarm feature, reaching FPR $= 0$ for some interval $T_i = 0.5$–0.75 days; for the same time intervals, the precision is high, reaching the value of 1. If this feature for some of the clusters is under the threshold, this information is not particularly relevant, but if it is over the threshold, the cluster is very likely to be a type A. This information is quantitatively stored in the probability vector that concurs in NESTORE classification. For the interval T_1 (from 1 min to 0.25 days after the mainshock), the feature is unreliable (FPR $=$ TPR $= 0$). The Precision cannot be shown, because no element is classified as A. In other words, the decision tree classifies all clusters as B and the feature is useless. The Z feature supplies performances in the range 0.25–0.75 days close to the diagonal of the ROC graph. For $T = 1$ day, the performances are the worst and correspond to the bottom right corner of the ROC graph (FPR $= 1$ and TPR $= 0$). This is also confirmed by the PR graph.

4.2. Case studies

In this section, some examples of the well-known clusters in California with mainshock magnitudes greater than 5.8 are shown. To be shown, the clusters, if of type A, need to have the SSLE at least 6 h after the mainshock.

Table 1 shows the selected clusters. The last column of the table shows NESTORE forecasting; A/B represents the classification changes depending on the time interval Ti (see Fig. 7).

For Superstition Hill and Ridgecrest clusters (1995 and 2019), the o-mainshock does not coincide with the mainshock, but it is a foreshock. In this case, NESTORE correctly forecasts the stronger earthquake(s) following the foreshock.

Fig. 7 shows the performances of the method on the clusters in Table 1. The symbols are red for type A clusters and blue for type B clusters. The algorithm correctly identifies all A clusters, starting 6 h after the mainshock. Note that the algorithm stops classifying when an event with magnitude $M_a \geq$ Mm-1 occurs. The classification of Ridgecrest 2019 stops after 12 h because an earthquake of magnitude 5.4 occurred on July 5, 2019–16 h after the o-mainshock. Regarding the B clusters, NESTORE correctly classifies the Westmoreland and Sierra Madre clusters. The Hector Mine cluster (blue circles) is incorrectly classified (Prob(A) > 0.5) for short time periods T_i but becomes correct one day after the mainshock when more information is available. The algorithm fails in the classification of North Palm Spring 1986. In both the Hector Mine and North Palm Spring clusters, the algorithm shows values of probability in the range 0.4–0.6 for some Ti, which, accordingly with Gentili and Di Giovambattista (2017), are bounds of uncertain classification.

Regarding the North Palm Spring cluster (blue triangles in Fig. 7), the classification is always A, even if it decreases in time. The extremely high value of all features (for example, N2 is two times the threshold for $T = 6$ h) makes this cluster an outlier with respect to the behavior of the other clusters. The presence of some outliers in the dataset may influence the values of the threshold and therefore the classifier reliability. For this reason, a comparison among different training set outputs is necessary to define more reliable parameters.

Ridgecrest 2019, a cluster correctly classified as A, can be considered a retrospective forecast because the training set is composed only of clusters occurring before July 2019. In other words, if we trained NESTORE a few days before the cluster, by using all available data, we would have had a correct forecast.

4.3. A robust classifier – rNESTORE

We investigated differences in the thresholds found by different training sets in the training phase of the LOO method. The training sets are very similar: there is a difference of two clusters from one training set to the other for $T = 0.25$ days. Therefore, what is interesting is not

Table 1
Case studies for southern California. The date, latitude, longitude and magnitude are listed, followed by the date and magnitude of the SSLE. Abbr: abbreviation of cluster's name, in Fig. 7 legend. Type: class of the cluster. NESTORE forecasting: NESTORE classification starting 6 h after the mainshock. A/B: if classification changes in time.

Cluster	Abbr.	o-Main date yyyy/mm/dd	Lat	Lon	M_m	SSLE date yyyy/mm/dd	M_a	Type	NESTORE forecasting
Westmorland	W	1981/04/26	33.1	−115.62	5.8	1981/04/26	3.7	B	B
North Palm Springs	NPS	1986/07/08	34.00	−116.61	6.0	1986/07/17	4.5	B	A
Whittier Narrows	WN	1987/10/01	34.06	−118.08	5.9	1987/10/04	5.3	A	A
Superstition Hill	SH	1987/11/24	33.09	−115.79	6.2	1987/11/24	6.6	A	A
Sierra Madre	SM	1991/06/28	34.27	−117.99	5.8	1991/06/28	4.3	B	B
Ridgecrest	R95	1995/08/17	35.78	−117.66	5.4	1995/09/20	5.8	A	A
Hector Mine	HM	1999/10/16	34.60	−116.27	7.1	1999/10/16	5.8	B	A/B
Ridgecrest	R19	2019/07/04	35.71	−117.50	6.4	2019/07/06	7.1	A	A

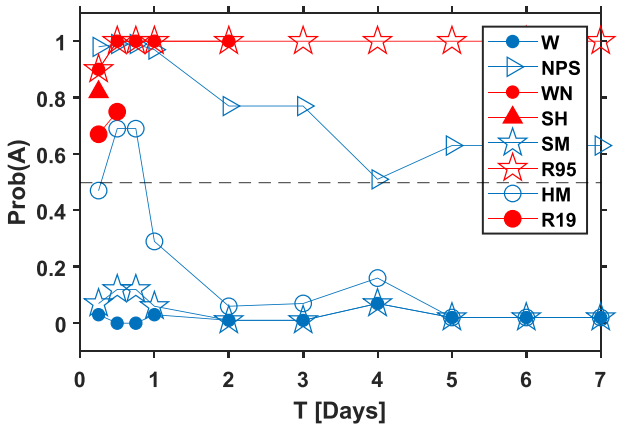

Fig. 7. Tests of NESTORE classification on clusters with Mm ≥ 5.8. Each cluster is tested using all other clusters for training. Red: A clusters. Blue: B clusters. Legend: W=Westmorland; NPS = North Palm Springs; WN=Whittier Narrows; SH = Superstition Hill; SM = Sierra Madre; R95 = Ridgecrest 1995; HM = Hector Mine; R19 = Ridgecrest 2019. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

what is stable but what it is not stable. If the difference of two in 50 examples may influence the feature, it indicates that the feature is not stable. The instability may be of two types: (1) the feature threshold is defined by NESTORE but is (slightly) different from one training set to the other, and (2) the threshold is not defined for some training sets. The first instability may be related to (a) a finite and (relatively) small number of clusters with a feature value close to the edge between the two classes or to some local minima in the choice of the threshold confused with the absolute minimum because of the small dataset or (b) to an interval of feature values in which the classes are truly mixed. For instability of type (1), however, most clusters are well classified and are not affected by the change in threshold value. In case (a), perhaps a larger training set may reduce the fluctuations, adding constraints to the threshold value. In case (b), a three-class classifier is needed, the classes "A" "B" and "unknown" for intermediate values of the feature. For this classifier, at least two different thresholds are necessary. However, this approach is beyond the scope of this paper due to the small number of clusters available for training. Instability of type (2) causes the failing of the classifier in obtaining reliable results; it occurs when, due to a low amount of data for some training sets, an overfitting occurs. Table 2 shows for each feature and each time interval the percentage of training sets for which a reliable threshold has been found.

We developed a more stable classifier for future classification, called rNESTORE, also considering the small number of available data and the effect of outliers. To do this, we selected only the values of T for which at least 95% of the classifiers found a reliable threshold (in boldface in the table). Table 3 shows the mean and standard deviation (in parentheses) of the values of the thresholds for different training sets in the selected time intervals. For feature N2, which can assume only integer values, the standard deviation is 0. SLcum and QLcum2 features have been eliminated from the table because they do not show reliable results for any value of T. The value of the mean of the thresholds of Table 3 is used as threshold vector for rNESTORE. In other words, rNESTORE adopts a technique known in statistics as "Jackknife resampling" for estimating the threshold value (Efron and Stein, 1981). The probability vector values are estimated from the original complete dataset, as the ratio of A clusters and total number of clusters under (p_u) and over the threshold (p_o) and are shown in Table 4. Boldface numbers correspond to intervals T_t for which the threshold has been calculated for the current interval Ti feature. For longer time periods, the same threshold is used, inherited from the last interval T_t; the probability vector, vice-versa, is calculated for every time interval, to take into account the different number of A clusters available.

Figs. 8a, c show the performances rNESTORE on the complete original dataset. It is important to note that this is a test on internal coherence of the dataset (self-test) and not on the entire classifier, because, in this case, the training set and test set are coincident. Testing a classifier by the same dataset used for the training can provide biased performances because it does not consider overfitting issues. However, such a test can be useful in understanding whether feature values are mutually consistent.

rNESTORE performances (Figs. 6a and c) are very stable when

Table 2
Percentage of training sets in which each feature was considered reliable for each time period. Bold face: over 95%.

T [days]	S	Z	SLcum	QLcum	SLcum2	QLcum2	Q	Vm	N2
0.25	0	88	0	0	0	0	62	78	**100**
0.50	0	**100**	0	**100**	0	0	76	**100**	**100**
0.75	0	**100**	0	67	0	0	78	**100**	**100**
1	0	28	14	0	14	0	93	**100**	**100**
2	0	3	13	0	**100**	0	93	**100**	**100**
3	0	0	13	0	**100**	0	92	**97**	**100**
4	0	0	14	0	72	0	94	0	11
5	**100**	0	11	0	8	0	**97**	0	0
6	**100**	0	11	0	8	0	**100**	0	0
7	**100**	0	11	0	8	0	**100**	0	0

Table 3
Mean and the standard deviation (in parentheses) for the thresholds for the cells in Table 2 with value >95%. Boldface numbers correspond to intervals Tt.

	S	Z	QLcum	SLcum2	Q	Vm	N2
6 h							**2.5(0)**
12 h		**0.020(0.007)**	**0.18(0.02)**			**0.45(0.02)**	**2.5(0)**
18 h		0.020(0.007)				0.45(0.02)	2.5(0)
1 d						0.45(0.02)	2.5(0)
2 d				**0.0916(0.004)**		0.45(0.02)	2.5(0)
3 d				0.0916(0.004)		0.45(0.02)	2.5(0)
4 d							
5 d	**0.0885(0.0006)**				**0.0154(0.0004)**		
6 d	0.0885(0.0006)				**0.016 (0.002)**		
7 d	0.0885(0.0006)				0.016 (0.002)		

Table 4
p_u and p_o values for the cells in Table 2 with value >95%. Boldface numbers correspond to intervals Tt.

	S		Z		QLcum		SLcum2		Q		Vm		N2	
	p_u	p_o	p_u	p_o	p_u	p_o	p_u	p_o	p_u	p_o	p_u	p_o	p_u	p_o
6 h													**0.21**	**0.82**
12 h			**0.14**	**0.61**	**0.06**	**0.58**					**0.23**	**0.80**	**0.10**	**0.81**
18 h			0.14	0.61							0.21	0.75	0.11	0.76
1 d											0.19	0.72	0.11	0.73
2 d							**0.10**	**0.72**			0.19	0.62	0.11	0.67
3 d							0.10	0.7			0.19	0.67	0.11	0.63
4 d														
5 d	**0.04**	**0.06**							**0.04**	**0.67**				
6 d	0.04	0.06							**0.04**	**0.67**				
7 d	0.04	0.06							0.04	0.67				

changing Ti, and they are all in the upper left triangle close to the upper left corner for ROC and well above the random guessing line for PR. In particular the quality of the classifier increases from 0.25 to 0.50 decrease in the following 3 days and again increases starting with 5 days, thanks to the use of Q and S features, they supply the best performances. The best performances are obtained for ROC graph at 5–7 days, while for PR graph at 0.5 days with the following similar values respectively: TPR = Recall = 0.86 FPR = 0.07 Precision = 0.75 at 5–7 days and TPR = Recall = 0.85 FPR = 0.14 Precision = 0.81 at 0.5 days. Figs. 8b, d show the performances of features N2, Q and Z for different time intervals T. N2 performances in Figs. 8 and 6 are coincident due to the good stability of the feature. Conversely, the Q and Z performances are enhanced in Figs. 8b and d with respect to the corresponding Figs. 6b and d due to the elimination of incorrect thresholds found in a few databases and to the better choice of reliability intervals. In particular, unreliable performances have been eliminated because the features are no longer used for the corresponding intervals (see Fig. 8 and Tables 2 and 3). Of particular interest is the elimination of the Q feature for Ti < 6 days. The seemingly good performance of the feature, in terms of Precision, Recall, and FPR was evaluated by the LOO method using only trainings in which the feature was considered reliable; however, there were training in which the results were so poor that the feature was eliminated as unreliable. rNESTORE was able to highlight this instability and remove it from the list of reliable features for small periods of time.

To understand how feature thresholds fluctuate throughout the different training sets, Fig. 9a shows the threshold found by the corresponding trainings for the Z feature for $T = 12$ h. Even if most experiments found a threshold of approximately 0.02, in a few cases, the values are different; in particular, in four cases, the threshold is ~0.04. Fig. 9b shows the fluctuations of the threshold of the Q feature for $T = 6$ days. The Q feature in most cases is 0.0155. However, in 10 cases it is 0.015. In only one case, the threshold is set to 0.027.

Comparing Figs. 6 and 8, the choice of the thresholds of 0.02 for Z and of 0.0154 for Q improves the performances of the features.

4.4. Test on an independent dataset

To test rNESTORE on an independent database, we used for northern California the Comprehensive Earthquake Catalog (ComCat - available at the web-site: https://earthquake.usgs.gov/earthquakes/search/), expressed also in M_L magnitude (like SCEC catalog) and the SCEC catalog itself from 2020 to present. Fig. 10a shows the map of the analyzed area and the positions of the o-mainshock of the clusters, while Fig. 10b shows the completeness magnitude of the ComCat catalog in the selected region (green upper polygon). The completeness magnitude was obtained by using Zmap software (Wiemer, 2001) and the maximum curvature method.

For the test set California data, we evaluated the completeness magnitude directly on the cluster when at least 80 earthquakes were available by using the maximum curvature method; otherwise, we used a completeness magnitude of 3.5 for norther California to be compatible with Fig. 10b and of 3 for Southern California, like for the training data set. We added a correction factor of 0.2 to both magnitudes. Due to the high completeness magnitude of the catalog, this test set is small: it is composed of 11 clusters, of which 3 are of class A and 8 are of class B for T = 6 h. Some clusters are on the coast, while others are in the eastern California or on the California-Nevada border.

Table 5 shows the selected clusters. Fig. 11 shows the comparison of the performance of NESTORE (Fig. 11a, c, e) and rNESTORE (Fig. 11b, d, f) for the 11 clusters. In particular, Fig. 11a and b show the estimated probability of being a cluster A for the two methods in time. Red symbols correspond to type A clusters and blue symbols to type B clusters. For T = 6 h, an A cluster is misclassified, but the classification becomes correct for longer times. The main difference between the classifiers is for Ti = 2 and 3 days, when rNESTORE correctly classifies Ocotillo Wells clusters as type B, with a Prob(A) < 0.5, while NESTORE fails in the classification. Note that while the ROC graph, being insensitive to the relative abundance of A and B clusters, has an identical response regarding NESTORE results for Ti of 2 and 3 days (Fig. 11c), the PR graph provides a different representation because the Precision depends on this abundance (Fig. 11 e). Neither the ROC graph nor the PR provides any information for $T > 3$ days because no more A clusters are available in the

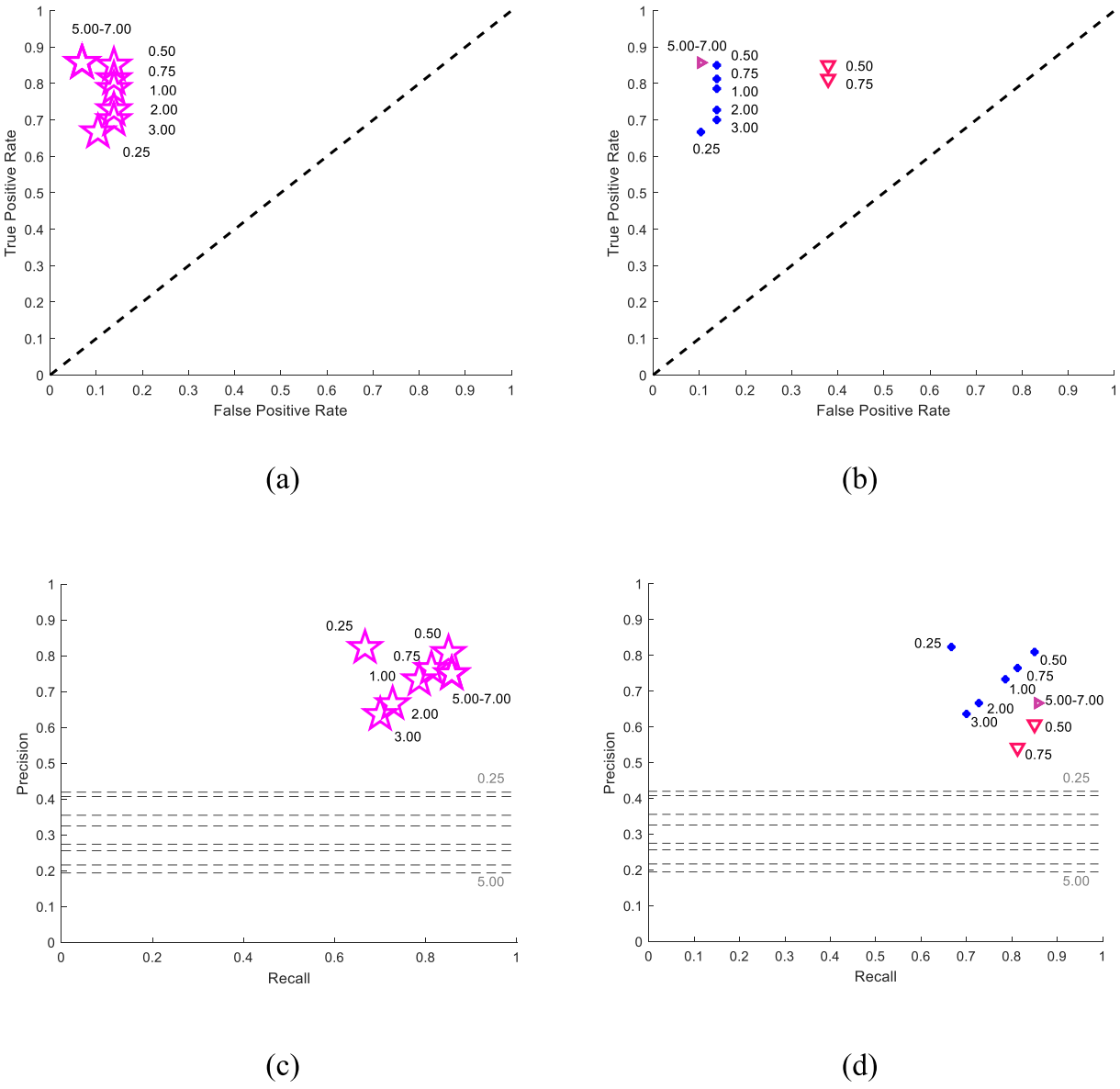

Fig. 8. rNESTORE performances for different Ti values (self-test). Time periods Ti are listed close to the corresponding star. Magenta stars: NESTORE performances; blue dots: N2; violet small triangles: Q; and red larger triangles: Z. (a) ROC graph for NESTORE; (b) ROC graph for selected features; dashed line correspond to random classifier (c) PR graph for NESTORE (d) PR graph for selected features; dashed lines: random classifiers for different time periods: from up to bottom: 0.25, 0.5, 0.75, 1, 2, 3, 4, 5 days longer time period coincide with the ones of 5 days. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

test set for those periods. Both methods correctly classify all clusters for time intervals 0.5–1 day, which corresponds to Recall = 1, Precision = 1, and FPR = 0.

5. Discussion and conclusions

In this paper, the NESTORE machine learning algorithm has been applied to California seismicity for type A clusters forecasting. The algorithm analyzes several features of the cluster seismicity at increasing time intervals from the o-mainshock and uses, for each time interval, only features that provide best performances. Each feature is used to train a decision tree and estimate the probability to be an A cluster; the probabilities estimated from different features are merged using a Bayesian approach to obtain the NESTORE response. The best performances were obtained for time periods $\leq$3 days. In particular, the best performances are obtained for a time interval of 0.5 days (12h) after the mainshock, which makes the algorithm very attractive for an early warning application, because the probability of a new dangerous earthquake after a first strong event may be estimated in advance. A careful analysis of the features and the time periods in which they are relevant for classification can be helpful in understanding the SSLE preparation process. In particular, at short mainshock timescales (6–12 h) the N2, Z, QLcum, and Vm features provide good results (see Table 2). This result for California can be compared with those for northeastern Italy-western Slovenia (Gentili and Di Giovambattista, 2020), corresponding to Alpine and Dinaric seismicity, and with those for all Italy, dominated by Apennine seismicity (Gentili and Di Giovambattista, 2017). The N2, Vm, and Z features showed good short-term performance after o-mainshock for all three studies. N2 is the number of events with magnitude $\geq$Mm-2 after mainshock, Vm corresponds to the cumulative change in magnitude between successive events, and Z represents the linear concentration of events. The three studies, while corresponding to very different seismotectonic characteristics, show that high cluster productivity, irregularity in the magnitude distribution over time, and

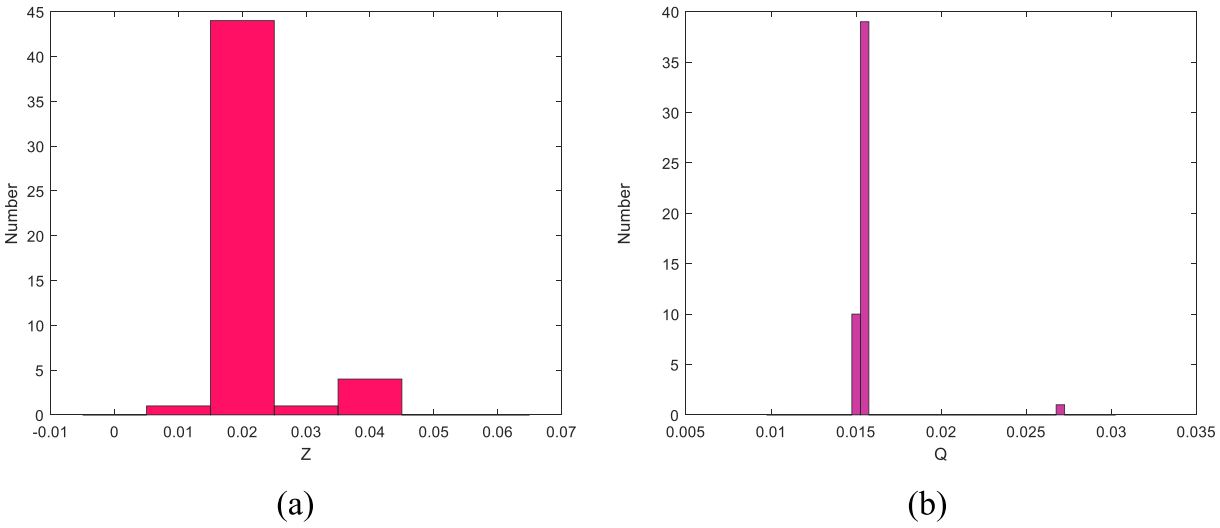

Fig. 9. Feature thresholds for (a) the Z feature for $T_2 = 12$ h and (b) the Q feature for $T_9 = 6$ days.

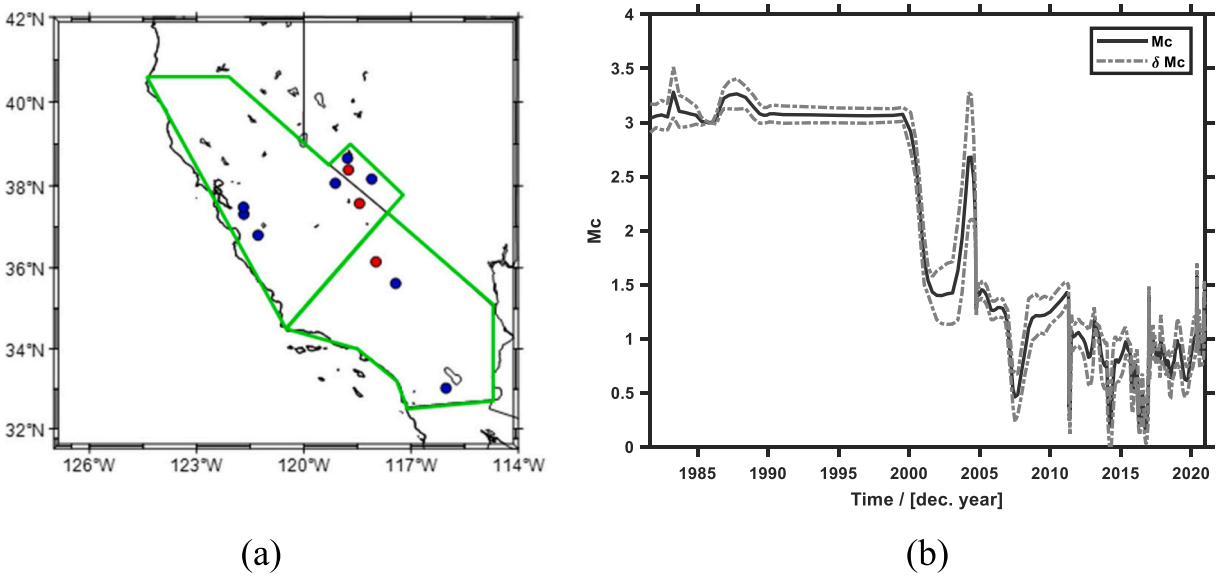

Fig. 10. (a) Green: polygons of analyzed regions; upper polygon: Comcat catalog; and downer polygon: SCEC catalog. Positions of the o-mainshocks of the analyzed clusters; dots: training set; circles: test set; blue symbols: B class; and red symbols: A class. (b) Completeness magnitude of the ComCat catalog over time in the corresponding polygon. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 5
Case studies for the independent test set. The date, latitude, longitude and magnitude are listed, followed by the date and magnitude of the SSLE. Abbr: abbreviation of cluster's name, in Fig. 11 legend. Type: class of the cluster. NESTORE forecasting: NESTORE classification starting 6 h after the mainshock: A/B if the classification changes in time.

Cluster	Abbr.	o-Main date yyyy/mm/dd	Lat	Lon	M_m	SSLE date yyyy/mm/dd	M_a	Type	NESTORE forecasting
Morgan Hill	MH	1984/04/24	37.31	−121.68	6.2	1984/09/26	4.4	B	B
Central California	CC	1986/01/26	36.8	−121.28	5.5	1986/01/26	4.0	B	B
Alum Rock	AR	1986/03/31	37.48	−121.69	5.7	1986/12/11	4.1	B	B
Chalfant Valley	CV	1986/07/20	37.57	−118.44	5.9	1986/07/21	6.4	A	A
Mono County	MC	1990/10/24	38.06	−119.12	5.8	1990/11/05	4.3	B	B
Hawthorne, Nevada	HN	2011/04/11	38.38	−118.75	4.1	2011/04/17	4.6	A	A
Walker Lake	WL	2016/03/22	38.66	−118.78	4.1	2016/03/22	2.8	B	B
Mina, Nevada	MN	2020/05/18	38.16	−118.10	4.3	2020/05/22	2.7	B	B
Ocotillo Wells	OC	2020/05/10	33.02	−116.02	3.3	2020/05/12	1.2	B	A/B
San Bernardino	SB	2020/06/04	35.62	−117.43	5.5	2020/06/08	4.1	B	B
Coso Junction	CJ	2020/06/04	36.15	−117.98	4.2	2020/06/07	3.5	A	A/B

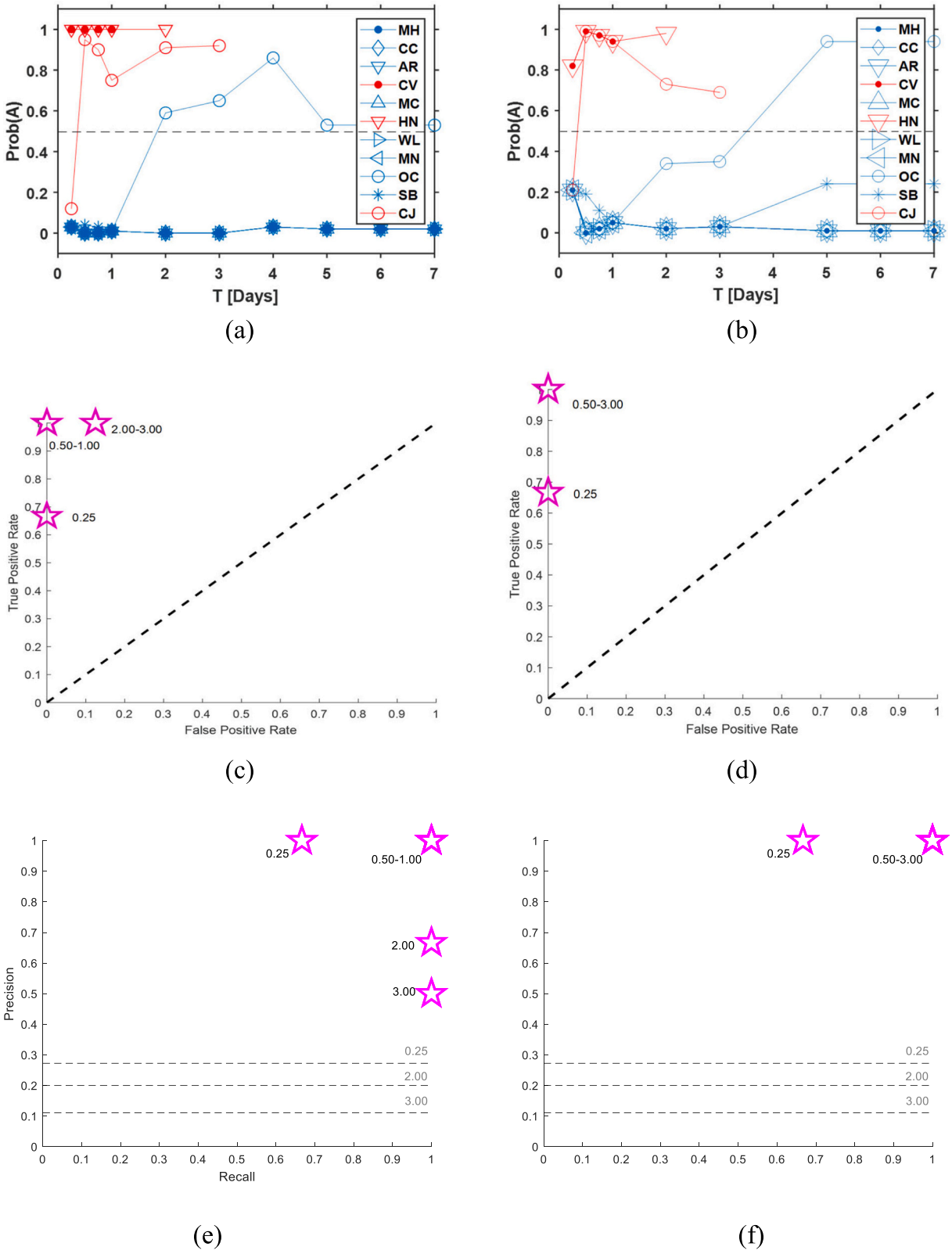

Fig. 11. Tests of the robust classifier on the independent test set for NESTORE and rNESTORE. Red symbols: A clusters, Blue symbols: B clusters. MH = Morgan Hill; CC=Central California AR = Alum Rock; CV=Chalfant Valley; MC = Mono County; HN=Hawthorne Nevada; WL = Walker Lake; MN = Mina Navada; OC=Ocotillo Wells; SB=San Bernardino; CJ = Coso Junction (a) Probability to be A vs time for NESTORE (b) Probability to be A vs time for rNESTORE (c) ROC graph showing NESTORE performances (d) ROC graph showing rNESTORE performances. (e) PR graph showing NESTORE performances (f) PR graph showing rNESTORE performances. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

event concentration are clues to the process of preparing for a strong subsequent earthquake (SSLE). Note that in previous work Vm was evaluated in the magnitude range [Mm-3, Mm], while in this work the smaller range [Mm-2, Mm] is used, without affecting the performance of the feature.

Other features are characterized by zone-dependent performance or have been more rigorously evaluated through NESTORE enhancements. The Q feature, for example, which depends on the cumulative radiated energy, in previous works was considered reliable even for small time intervals. rNESTORE allows us to highlight its instability in California for short time periods and then use this feature only for longer time periods. QLcum, on the other hand, which corresponds to the deviation of Q from the long-term trend, provides good results for both California and NE Italy-Western Slovenia, while it needs longer time intervals for Italian seismicity (Apennines).

Summarizing, we obtained the following results:

1. We tested the performance of the algorithm using the LOO method, on fifty clusters recorded in the SCEDC earthquake catalog from 1981 to 2020 and showed the results on the ROC and PR graphs. All tests showed that the performance of NESTORE was reliable for all time periods because it was in the upper left triangle of the ROC graph, and above the corresponding random guess line for the PR graph. In particular, we found the best performance of FPR = 0.17, TPR = Recall = 0.80 and Precision = 0.76 for Ti = 0.5 days (12 h), while for time periods Ti > 3 days, the performance worsened due to instability of some features.
2. For 8 known clusters with Mm > 5.8, we showed the performances of NESTORE over time: we had 6 correct classifications for all Ti, one classification starting to be correct after one day, and one outlier. The successful classification of Ridgecrest 2019 can be considered a retrospective forecast because the training was performed using only information from previous clusters.
3. Using a jackknife approach, we developed a more robust classifier, called rNESTORE, whose performance was compared with NESTORE one on a small independent set: rNESTORE supplies better performances for time intervals of 2–3 days. Both algorithms provide the best performances for intervals of 0.5–1 day. Although the results should be verified in the future on a larger database, they are encouraging because all the clusters are classified correctly. In particular, the good performances for the 0.5 days test are in good agreement with the test of point 1.
4. Accurate analysis with rNESTORE, together with results from previous applications on different seismotectonic regimes, allows us to indicate high cluster productivity, irregularity in magnitude distribution over time, and event concentration as precursors of an SSLE.

Considering these results, after careful validation with rigorous statistical testing on a larger database, rNESTORE could lead to an operationally qualified and regularly updated early warning system for SSLEs.

CRediT authorship contribution statement

S. Gentili: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Project administration, Resources, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing. **R. Di Giovambattista:** Conceptualization, Methodology, Supervision, Writing – original draft, Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

Funded by a grant from the Italian Ministry of Foreign Affairs and International Cooperation.

We would like to thank Robert Shcherbakov for his useful information on California catalogs and for supplying us with a code for downloading the ComCat catalog in ML units. We also wish to thank Francesco Giannitrapani for NESTORE's icon we used in Graphical Abstract.

We thank the anonymous reviewers for their comments and suggestions.

The southern California earthquake catalog and the focal mechanisms were provided by the SCEDC (2013): Southern California Earthquake Center. Caltech Dataset. doi:https://doi.org/10.7909/C3WD3xH1 and was downloaded from the following website: https://service.scedc.caltech.edu/eq-catalogs/date_mag_loc.php (last accessed in August 2021).

https://service.scedc.caltech.edu/eq-catalogs/FMsearch.php (last accessed in January 2021).

The CompletenessWeb website is available at address http://www.completenessweb.gfz-potsdam.de/doku.php (last accessed in September 2020).

The Comprehensive Earthquake Catalog (ComCat) was downloaded from the U.S.G.S. website https://earthquake.usgs.gov/earthquakes/search/ (last accessed in January 2021).

References

Båth, Markus, 1965. Lateral inhomogeneities of the upper mantle. Tectonophysics 2 (6), 483–514.

Brodsky, Emily E., 2019. Determining whether the worst earthquake has passed. Nature 574, 185–186. https://doi.org/10.1038/d41586-019-02972-z.

Dascher-Cousineau, K., Lay, T., Brodsky, E.E., 2020. Two foreshock sequences post Gulia and Wiemer (2019). Seismol. Res. Lett. 91, 2843–2850. https://doi.org/10.1785/0220200082.

Davis, J., Goadrich, M., 2006. The relationship between precision-recall and ROC curves ICML '06: Proceedings of the 23rd international conference on Machine learning, June 2006, pp. 233–240.

Di Giovambattista, R., Tyupkin, Yu.S., 2002. Burst of aftershocks as a manifestation of instability of the earth crust in an area of strong earthquake preparation. European seismological commission (ESC). XXVIII General Assembly, 1–6 September 2002 Book of Abstracts, p. 228.

Efron, B., Stein, C., 1981. The jackknife estimate of variance. Ann. Stat. 9 (3), 586–596. https://doi.org/10.1214/aos/1176345462.

Egan, J.P., 1975. Signal Detection Theory and ROC Analysis, Series in Cognition and Perception. Academic Press, New York.

Fawcett, T., 2004. ROC graphs: notes and practical considerations for researchers. Pattern Recogn. Lett. 31 (8), 1–38.

Fawcett, T., 2006. An introduction to ROC analysis. Pattern Recogn. Lett. 27, 861–874.

Gardner, J.K., Knopoff, L., 1974. Is the sequence of earthquakes in Southern California, with aftershocks removed, Possionian? Bull. Seismol. Soc. Am. 64 (5), 1363–1367.

Gentili, S., Bressan, G., 2008. The partitioning of radiated energy and the largest aftershock of seismic sequences occurred in the northeastern Italy and western Slovenia. J. Seismol. 12, 343–354.

Gentili, S., Di Giovambattista, R., 2017. Pattern recognition approach to the subsequent event of damaging earthquakes in Italy. Phys. Earth Planet. Inter. 266, 1–17.

Gentili, S., Di Giovambattista, R., 2020. Forecasting strong aftershocks in earthquake clusters from northeastern Italy and western Slovenia. Phys. Earth Planet. Inter. 303, 106483.

Godano, C., 2016. A new method for the estimation of the completeness magnitude. Phys. Earth Planet. Inter. 263 (2017), 7–11.

Grandori, G., Guagenti, E., Perotti, F., 1984. Some observations on the probabilistic interpretation of short-term earthquake precursors. Earthq. Eng. Struct. Dyn. 12, 749–760.

Gulia, L., Wiemer, S., 2019. Real-time discrimination of earthquake foreshocks and aftershocks. Nature 574, 193–199.

Gulia, L., Wiemer, S., 2021. Comment on "Two Foreshock Sequences Post Gulia and Wiemer (2019)" by Kelian Dascher-Cousineau, Thorne Lay, and Emily E. Brodsky. Seismol. Soc. Am. 92 (5), 3251–3258.

Gulia, L., Wiemer, S., Vannucci, G., 2020. Prospective evaluation of the foreshock traffic light system in Ridgecrest and implications for aftershock hazard assessment. Seismol. Res. Lett. https://doi.org/10.1785/0220190.

Helmstetter, A., Sornette, D., 2003. Båth's law derived from the Gutenberg-Richter law and from aftershock properties. Geophys. Res. Lett. 30, 2069.

Gutenberg, B., Richter, C.F., 1956. Earthquake magnitude, intensity, energy, and acceleration: (Second paper). Bull. Seismol. Soc. Am. 46 (2), 105–145.

Helmstetter, A., Kagan, Y.Y., Jackson, D., 2006. Comparison of short-term and time-independent earthquake forecast models for southern California. Bull. Seismol. Soc. Am. 96 (1), 90–106.
Hutton, Kate, Woessner, Jochen, Hauksson, Egill, 2010. Earthquake monitoring in Southern California for seventy-seven years (1932–2008). Bull. Seismol. Soc. Am. 100, 423–446. https://pubs.geoscienceworld.org/ssa/bssa/article/100/2/423/349275/Earthquake-Monitoring-in-Southern-California-for.
Kagan, Y.Y., 2002. Aftershock zone scaling. Bull. Seismol. Soc. Am. 92, 641–655.
Keilis-Borok, V.I., Kossobokov, V.G., 1990. Premonitory activa- tion of earthquake flow: algorithm M8. Phys. Earth Planet. Inter. 61 (1–2), 73–83.
Keilis-Borok I., V., Rotwain M., I., 1990. Diagnosis of Time of Increased Probability of strong earthquakes in different regions of the world: algorithm CN. Phys. Earth Planet. Inter. 61 (1–2), 57–72.
Kossobokov, V.G., Maeda, K., Uyeda, S., 1999. Precursory activation of seismicity in advance of Kobe, 1995 M=7.2 earthquake. Pure Appl. Geophys. 155, 409–423.307.
Lippiello, E., Godano, C., de Arcangelis, L., 2012. The earthquake magnitude is influenced by previous seismicity. Geophys. Res. Lett. 39, L05309. https://doi.org/10.1029/2012GL051083.
Nandan, S., Ouillon, G., Wiemer, S., Sornette, D., 2017. Objective estimation of spatially variable parameters of epidemic type aftershock sequence model: application to California. J. Geophys. Res. Solid Earth. https://doi.org/10.1002/2016jb013266.
Nandan, S., Ouillon, G., Sornette, D., Wiemer, S., 2019. Forecasting the rates of future aftershocks of all generations is essential to develop better earthquake forecast models JGR solid. Earth 124 (8), 8404–8425.
Persh, S.E., Houston, H., 2004. Strongly depth-dependent aftershock production in deep earthquakes. Bull. Seismol. Soc. Am. 94, 1808–1816.
Reasenberg, P., 1985. Second-order moment of Central California seismicity, 1969–82. J. Geophys. Res. 90, 5479–5495.
Rodríguez-Pérez, Q., Zúñiga, F.R., 2016. Båth's law and its relation to the tectonic environment: a case study for earthquakes in Mexico. Tectonophysics 687, 66–77.
Saito, T., Rehmsmeier, M., 2015. The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. PLoS One 10 (3), e0118432, 2015.
SCEDC, 2013. Southern California earthquake center. Caltech Dataset. https://doi.org/10.7909/C3 WD3xH1.
Shcherbakov, R., 2014. Bayesian confidence intervals for the magnitude of the largest aftershock. Geophys. Res. Lett. 41, 6380–6388.
Shcherbakov, R., Turcotte, D.L., 2004. A modified form of Båth's law bull. Seismol. Soc. Am. 94 (5), 1968–1975.
Shcherbakov, R., Zhuang, J., Ogata, Y., 2018. Constraining the magnitude of the largest event in a foreshock–mainshock–aftershock sequence. Geophys. J. Int. 212, 1–13.
Shcherbakov, R., Zhuang, J., Zöller, G., Ogata, Y., 2019. Forecasting the magnitude of the largest expected earthquake. Nat. Commun. 10, 4051.
Sweets, J.A., Dawes, R.M., Monahan, J., 2000. Better decisions through science. Sci. Am. 283, 82–87.
Tahir, M., Grasso, J.R., Amorese, D., 2012. The largest aftershock: how strong, how far away, how delayed? Geophys. Res. Lett. 39, L04301. https://doi.org/10.1029/2011GL050604.
van der Elst, N.J., 2020. B-positive: a robust estimator of aftershock magnitude distribution in transiently incomplete catalogs. J. Geophys. Res. Solid Earth 126 e2020JB021027.
Vere-Jones, D., 1969. A note on the statistical interpretation of Båth's Law. Bull. Seismol. Soc. Am. 59, 1535–1541.
Vorobieva, I.A., 1999. Prediction of a subsequent large earthquake. Phys. Earth Planet. Inter. 111, 197–206.
Vorobieva, I.A., Panza, G.F., 1993. Prediction of the occurrence of related strong earthquakes in Italy. Pure Appl. Geophys. 141, 25–41.
Wiemer, S., 2001. A software package to analyze seismicity: ZMAP. Seismol. Res. Lett. 72 (2), 373–382.
Woessner, J., Wiemer, S., 2005. Assessing the quality of earthquake catalogues: estimating the magnitude of completeness and its uncertainty. Bull. Seismol. Soc. Am. 95, 684–698.
Zaliapin, I., Ben-Zion, Y., 2013. Earthquake clusters in southern California, I: identification and stability. J. Geophys. Res. 118, 2847–2864. https://doi.org/10.1002/jgrb.50179.
Zhuang, Jiancang, Ogata, Yosihiko, Vere-Jones, David, 2002. Stochastic Declustering of Space-Time Earthquake Occurrences. J. Am. Stat. Assoc 97, 369–380.