Journal of Geophysical Research: Solid Earth

Supporting Information for

Earthquake declustering using the nearest-neighbor approach in space-time-magnitude domain

Ilya Zaliapin[1] and Yehuda Ben-Zion[2]

[1] Department of Mathematics and Statistics, University of Nevada, Reno

[2] Department of Earth Sciences, University of Southern California, Los Angeles

Contents of this file

Text Sections S1 to S3; Figures S1 to S9

Introduction

The Supporting Information discusses theoretical motivation for the proposed declustering algorithm, outlines the main steps of its numerical implementation, and includes figures with additional information about declustering in synthetic and real data. It also includes a version of declustered catalog of *Hauksson et al.* [2012].

S1. Motivation of the proposed declustering algorithm

Here we provide motivation and justification for the proposed declustering algorithm. It is based on the distribution analysis for the nearest-neighbor proximities and thinning theory of point processes. We discuss the case $w = 0$ (no magnitude component), which corresponds to the main version of our analysis. The magnitude-dependent case can be examined in a similar fashion. The discussion below explains why the proposed algorithm works in selected basic models of clustered fields, and why one can expect it to work in more general situations. We also discuss specific conditions under which the algorithm gives biased results.

S1.1 Weibull approximation to the nearest-neighbor proximity distribution

The basic model that we use in this analysis is a Poisson space-time point field that is stationary in time and homogeneous in d-dimensional space, with independent space and time components. We refer to the process by its counting measure [*Daley and Vere-Jones*, 2003]

$$H(A) = \text{number of events within space-time region } A.$$

The first moment measure of the process

$$M(A) = E[H(A)] = \lambda \int_A \mathrm{d}t\, \mathrm{d}x_1 \ldots \mathrm{d}x_d = \lambda |A|$$

is completely specified by the process intensity λ [$\mathrm{yr}^{-1}\mathrm{km}^{-d}$]. The number of events that occurred within a space-time region A with volume $|A|$ is a Poisson random variable with intensity $\lambda|A|$. We define the *earthquake proximity sphere* centered at event i with radius x as the space-time region

$$\mathrm{S}(i,\eta) = \{(t,\mathbf{x})\text{: the proximity from event } i \text{ to } (t,\mathbf{x}) \text{ is less than } \eta\}.$$

The nearest-neighbor proximity η_i of Eqs. (1,3) of the main text calculated for event i signifies that there are no events in the sphere $\mathrm{S}(i,\eta_i)$. The Poisson distribution for the number of events in space-time volumes implies

$$\mathrm{Prob}[\eta_i > x] = \mathrm{Prob}[H(\mathrm{S}(i,x))=0] = exp\{-\lambda|\mathrm{S}(i,x)|\}.$$

This allows one to find the distribution of the nearest-neighbor proximities. A complete analysis, which involves some additional technical requirements and auxiliary parameters to prevent spheres of infinite volumes, leads to the following approximate distribution [see *Zaliapin et al.*, 2008; *Hicks*, 2011]:

$$\mathrm{Prob}[\eta_i > x] \approx exp\{-\lambda\xi x^k\}. \tag{S1}$$

Here ξ, k are functions of dimension d and the auxiliary parameters; these functions are constants with respect to x. We assume that the values of ξ, k are constants for a given examined catalog.

The approximation (S1) is the Weibull distribution with shape parameter k and scale parameter $s = (\lambda\xi)^{-1/k}$. It provides a close fit to the proximities in the observed earthquake data and synthetic catalogs [*Zaliapin and Ben-Zion*, 2013a], and can be used for both integer and fractional dimensions d (see also Fig. S9).

The numerical values of the parameters ξ, k depend on the analysis assumptions (including possible errors in determining the fractal dimension of the epicenters); they are best estimated from the data. Analyses of multiple observed catalogs suggest that the background field corresponds to an approximate range $0.75 \le k \le 1.25$, and often the estimated values of k are close to unity. Recall that the case $k = 1$ in (S1) corresponds to the exponential distribution; the same as the distribution of interevent times in a homogeneous Poisson process [*Daley and Vere-Jones*, 2003].

S1.2 Gumbel approximation for the log-proximities

We start with a result that connects the Weibull and Gumbel distributions.

Lemma 1. Suppose a random variable X has the Weibull distribution with scale parameter $s > 0$ and shape parameter $k > 0$:

$$\mathrm{Prob}[X > x] = exp\{-(x/s)^k\}, \; x \ge 0. \tag{S2}$$

Then, the random variable $Y = \log_{10}(X)$ has the Gumbel (minimum) distribution

$$\mathrm{Prob}[Y > y] = exp\{-exp\{(y-\mu)/\beta\}\}, \; -\infty < y < \infty, \tag{S3}$$

with location parameter $\mu = \log_{10} s$ and scale parameter $\beta = (k \ln 10)^{-1}$. In particular,

$$\mathrm{E}[Y] = \log_{10} s - \gamma(k \ln 10)^{-1} \quad \text{and} \quad \mathrm{Var}[Y] = 1/6\, \pi^2 (k \ln 10)^{-2},$$

where $\gamma = 0.5772\ldots$ is the Euler-Mascheroni constant. Inversely, if random variable Y has the Gumbel distribution (S3), then the random variable $X = 10^Y$ has the Weibull distribution (S2).

Proof. By transforming the cumulative distribution functions of the Weibull and Gumbel distributions.

Consider now a point field with space-time intensity λ [$\mathrm{yr}^{-1}\mathrm{km}^{-d}$] and suppose that its nearest-neighbor proximity η_i is given by the Weibull distribution (S1) with shape parameter k and scale parameter $s = (\lambda\xi)^{-1/k}$. An example of such process is given by the homogeneous Poisson model of **Sect. S1.1.** Lemma 1 implies that the logarithm $\log_{10}\eta_i$ of the nearest-neighbor proximity has the Gumbel (minimum) distribution, with mean

$$\mathrm{E}[\log_{10}\eta_i] = -1/k \log_{10}(\lambda\xi) - \gamma(k \ln 10)^{-1}$$

and variance

$$\mathrm{Var}[\log_{10}\eta_i] = 1/6\,\pi^2\,(k\,\ln 10)^{-2}.$$

Here, the mean depends on the process intensity λ and the parameters ξ, k; and the variance is independent of the process intensity λ and is completely determined by the parameter k. Accordingly, the random variable

$$\log_{10}\alpha_i = \log_{10}\eta_i - \mathrm{E}[\log_{10}\eta_i]$$

has the Gumbel distribution with zero mean and variance that is independent of the process intensity λ. Lemma 1 also implies that the random variable α_i has the Weibull distribution with shape parameter k and scale parameter $exp(-\gamma/k)$. Importantly, the distribution of the random variable α_i does not depend on the process intensity λ.

Next, we apply the distribution results of **Sects. S1.1, S1.2** to each step of the declustering algorithm (**Sect. 4.1** of the main text).

S1.3 Step 1: Identifying the most clustered events

This step takes advantage of the well-documented bimodality in the distribution of the nearest-neighbor proximities. Figure S8 illustrates this in the global NCEDC catalog (panel a) and *Hauksson et al*. [2012] catalog for Southern California (panel b). A sharper separation between the modes can be achieved by considering a 2D space-time representation of the proximity, see **Sect. 3**, Eq. (4), as discussed by *Zaliapin et al.* [2008] and *Zaliapin and Ben-Zion* [2013a]. Independently of whether the bimodality is present or not, we expect the right part of the distribution (large proximities) to correspond to the background seismicity. The left part (short proximities) is expected to be a mixture of background and clustered events. Application of the cutoff proximity η_0 is intended to sample the long proximities, which quantify the (location-dependent) background event distribution. The randomized-reshuffled catalogs of Step 2, constructed with these sampled events, are used to approximate the distribution of nearest-neighbor proximities at each location in the absence of clustering. This estimation is necessarily biased (unless $\eta_0 = 0$ and the catalog is unclustered, which is not the case in most interesting practical situations), since it only uses a fraction of background events (those with parent proximity above η_0) and hence underestimates the background intensity as each location (i.e., produces a higher fraction of large proximity values). The better is the separation of the clustered and background modes (see Fig. S8), the smaller is the bias. Even in presence of the bias, the resulting estimation should reasonably approximate the relative background intensity. This is confirmed by the analysis of synthetic ETAS seismicity in **Sect. 6**.

S1.4 Step 2: Estimation of relative background intensity

According to **Sects. S1.1, S1.2**, the empirical distribution of the elements in the proximity vector $\boldsymbol{k}_i = (\kappa_{1,i},\ldots,\kappa_{M,i})$ is approximated by the Weibull distribution with scale parameter that is proportional to $(\lambda_i)^{-1/k}$, where λ_i denotes the estimated background intensity at location i, and k is the shape parameter close to unity.

We notice that one can closely estimate the relative location-dependent background intensity only in cases when the separation of the background intensity from the cluster intensity is comparable at different locations. For instance, the location-dependent background intensity may substantially vary from place to place, but if it is always substantially lower that the cluster intensity, our heuristics works. Furthermore, if the location-specific background intensity substantially overlaps the cluster intensity, but the degree of overlap is approximately the same at all locations; the heuristics is still valid. The situation when our estimation may give substantially biased results is when the location-dependent background intensity varies in such a way that in some locations it overlaps with the cluster intensity, and in other locations it does not. In this case, the proposed estimation may distort the relative background intensity levels. This is why we suggest to apply the technique to regions where the expected background intensities do not vary over an order of magnitude.

S1.5 Step 3: Normalized nearest-neighbor proximities

At this step, we obtain the normalized nearest-neighbor proximities α_i by rescaling the observed proximities η_i according to the mean of the proximity vector $\boldsymbol{k}_i$. The goal is to obtain distribution of α_i that is independent of the estimated location-specific background intensity λ_i. The proposed normalization of Eq. (7) uses logarithmic representation of the proximity vector, and hence is less sensitive to possible outliers.

In a catalog with constant background intensity λ, no clustering, and using $\eta_0 = 0$, the normalized proximities α_i have the Weibull distribution, with parameters independent of the intensity λ; see **Sect. S1.2**. One can expect that a similar argument is heuristically applied to a catalog with space-varying intensity $\lambda(\mathbf{x})$, no clustering, and using $\eta_0 = 0$. Finally, in presence of clustering and with $\eta_0 > 0$, the right tail of the distribution of α_i is approximately Weibull with intensity-independent parameters, while the left tail might be heavier (a larger proportion of small values) depending on the cluster intensity.

S1.6 Step 4: Thinning by the observed value of normalized proximity

The main component of the declustering procedure is <u>Step 4</u>, which applies thinning with the retention probability of event *i* being proportional to its normalized proximity α_i. The motivation for this procedure comes from the general theory of thinning for point processes [*Schoenberg*, 2003; *Daley and Vere-Jones*, 2008]. As a simple motivation example, consider a (possibly multidimensional) Poisson point process with intensity $\lambda(\mathbf{x})$ and apply thinning independently to every event with the retention probability $p(\mathbf{x})$. Then the thinned process is Poisson with intensity $p(\mathbf{x})\lambda(\mathbf{x})$. For instance, if the retention probability is

$$p(\mathbf{x}) = \lambda_0/\lambda(\mathbf{x}), \qquad \text{(S4)}$$

then the thinned process is homogeneous Poisson with constant intensity λ_0.

Application of this general idea to thinning by *estimated* process intensity is a delicate problem; see *Schoenberg* [2003], *Moeller and Schoenberg* [2010], and *Clements et al.* [2012] for a comprehensive discussion and further references. Notably, in one-dimensional case one can avoid complicated estimation of the process intensity, and use a process-dependent thinning to still obtain a homogeneous point process. Specifically, it can be shown (see Lemma 14.2.7 in Chapter 14 of *Daley and Vere-Jones*, [2008]) that thinning of a point process with intensity $\lambda(t) > \lambda_0$ using process-dependent retention probability $\min\{\lambda_0(t_i - t_{i-1}),1\}$ results in a point process with intensity $\lambda_0 + \varepsilon(t)$, where the deviation term $\varepsilon(t)$ decreases as $\lambda(t)/\lambda_0$ increases. In other words, the process-dependent thinning results in an almost-homogeneous point process, even if the process intensity is unknown. If one interprets the quantity $(t_i - t_{i-1})^{-1}$ as a single-point estimation of the process intensity $\lambda(t)$ at time t_i, then the process-dependent thinning is a natural extension to the general thinning result (S4).

This theoretical background motivates us to suggest a process-dependent earthquake thinning procedure. Recall that the shape parameter of the Weibull approximation to the nearest-neighbor proximity η_i is close to unity. This means that the distribution of η_i is close to exponential, the same as the interevent time distribution in the above result. We use thinning with retention probability proportional to the observed normalized nearest-neighbor proximity α_i. In the Weibull model (S1), the MLE of the inverse intensity λ^{-1} based on a single observation x is given by

$$\xi\,[x/\Gamma(1+1/k)]^k \approx \xi x \qquad (\text{since } k \approx 1),$$

where $\Gamma(x)$ is the gamma function. This allows one to expect that thinning with retention probability $\min\{A_0\alpha_i,1\}$ results in a point field with approximate intensity A_0/ξ.

Figure S9 shows a Weibull approximation to the normalized nearest-neighbor proximities α_i after thinning of Step 4 for the global and southern California catalogs. The fit, although not perfect, is very close. This may serve as an indication that the above heuristics does work in the examined data. This is inspiring, given the enormous variety of seismic regimes, background intensities, and cluster forms that has been analyzed in each examined case. We finally mention that the fit is even closer when examining local regions that are characterized by more uniform background and cluster properties.

S2. Numerical implementation

The numerical implementation of the declustering algorithm (**Sect. 4.1**) is described below:

1. Set parameters
   d (fractal dimension of epicenters/hypocenters);
   w (parameter of the proximity of Eq. (1));
   η_0 (initial cutoff threshold);
   α_0 (cluster threshold);
   M (number of reshufflings).

2. Calculate the nearest-neighbor proximity η_i for each event in the catalog using Eqs. (1),(3).

3. Select N_0 events that satisfy $\eta_i > \eta_0$.

4. Create M randomized-reshuffled catalogs and calculate the proximity vectors $\boldsymbol{k}_i$ for each event i. Specifically, for each $k = 1,\ldots,M$:

   a. Create N_0 independent and uniformly distributed time instants within the examined time interval;
   b. Reshuffle the locations of N_0 earthquakes selected in Step 3 using a random uniform permutation of $\{1,\ldots,N_0\}$. Independently, reshuffle the magnitudes of these events.
   c. Find the nearest-neighbor proximity $\kappa_{k,i}$ from each event i in the original catalog to the events of the randomized-reshuffled catalog k comprised of the random times from step (a) and reshuffled locations and magnitudes from step (b).

5. Calculate the normalized nearest-neighbor proximity α_i for each event in the catalog using Eq. (7).

6. Calculate the retention probability $P_{back,i}$ for each event i in the original catalog according to Eq. (8).

7. Identify background events according to the retention probabilities of Step 6.

Some practical comments are in order:

1. In Step 4c, the reshuffled catalog may include the event with the same location as event i from the original catalog. This happens if event i satisfies the condition $\eta_i > \eta_0$ and is used in reshuffling. Such a duplicate location should not be used in computing the proximity $\kappa_{k,i}$, as this leads to severe artifacts. Accordingly, for each event i that satisfies the condition $\eta_i > \eta_0$, the proximity $\kappa_{k,i}$ is computed using $N_0 - 1$ events of the k-th reshuffled catalog, excluding the event with the same location as event i.

2. For several initial events in the original catalog, a reshuffled catalog k may contain no earlier events. This leads to an infinite value of $\kappa_{k,i}$. Such infinite values should be excluded from calculating the average $mean[\log_{10}(\boldsymbol{k}_i)]$ in Eq. (7). Formally speaking, we calculate the conditional nearest-neighbor proximity $\kappa_{k,i}$ given that a randomized-reshuffled catalog k has events prior to event i of the original catalog.
3. The first event in the catalog has undefined η_i (no earlier events), and hence an undefined α_i. We use the convention that the first event does not satisfy the background condition (equivalently, $P_{\mathrm{back},1} = 0$).
4. As we mentioned in the main text, the parts 6 and 7 of the numerical algorithm are implemented via Eq. (9).
5. In part 4c, it is enough to only reshuffle events' magnitudes and use the original locations. Assigning random times to the original event locations serves as location reshuffling.
6. The value of the initial cutoff threshold η_0 can be selected using the bimodal distribution of the nearest-neighbor proximities η_i. Hence, one may first to calculate the proximities (part 2 of the numerical algorithm above), use them to select the value of η_0, and then run the other parts of the numerical algorithm.

S3. Sample declustered catalog

We include a version of declustering for the catalog of *Hauksson et al.* [2012] examined in this work. The catalog is in the file `2018JB017120-01.txt` and the format description is in the file `2018JB017120-02.txt.`

The sample declustering file refers to 123,275 events with magnitudes $m \geq 2.0$ during 1981 – 2018. The file reports (in column 13) the values of the logarithmic normalized proximities, $\log_{10}(\alpha_i)$, which allows one to produce declustering with different thresholds α_0 and create alternative stochastic realizations of declustering for a fixed α_0. In Matlab, this can be done with the following commands, which assume that the logarithmic proximities are stored in the variable `logalpha` and produces a vector `I` of background event indicators (logical `1` or `0`)

```
>> p = 10.^(logalpha-alpha0);
>> I = p>rand(size(p));
```

These commands identify background events that are the first events in the respective clusters (see **Sect. 4.1**). Identification of the largest events from each cluster can be done using the information of the spanning time-oriented tree, which is also provided in the file in the form of parent links (column 16).

As a specific example of declustering, the file also reports background event indicators for a single stochastic realization of the algorithm with the cluster threshold $\alpha_0 = 0$. Two types of the background indicators are given: the largest cluster event (column 14) and the first cluster event (column 15).

The file reports the SCSN event id (cuspid) in column 8. This allows one to get additional information about the examined events reported in the original catalog.

Example 1: Line 3 refers to event with the SCSN cuspid `3301566`; this event forms a cluster of a single event, and is identified as a background event. Accordingly, it has background index 1 in both column 14 (the largest cluster event indicator) and column 15 (the first cluster event indicator).

Example 2: Line 2 refers to event with the SCSN cuspid `3301565`; this event is a first event in a larger cluster and is identified as a background event. Accordingly, it has background index 0 in column 14 (the largest cluster event indicator) and index 1 in column 15 (the first cluster event indicator). The largest event in this cluster has index 59 (id `3316358`), that event has index 1 in column 14 and index 0 in column 15.

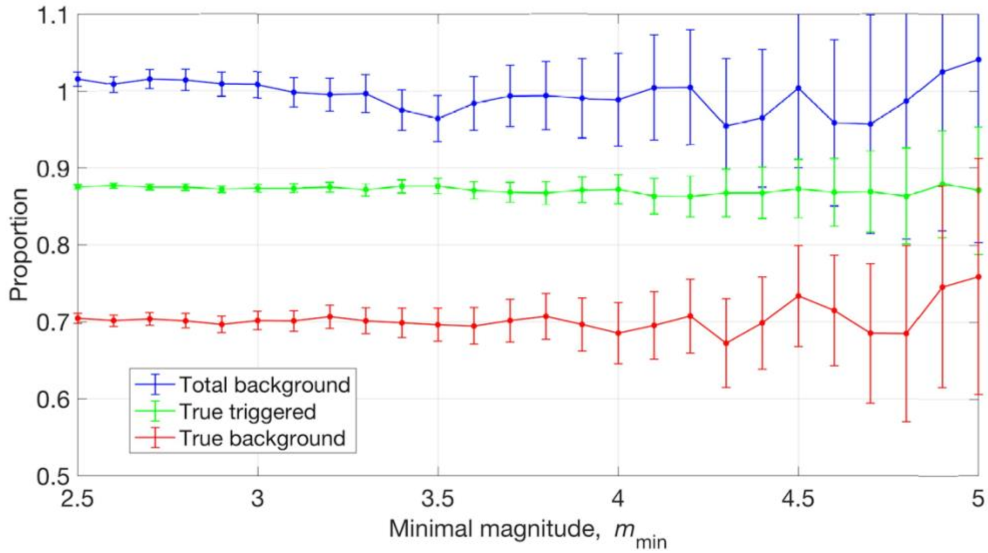

Figure S1: Declustering results for ETAS catalog of *Gu et al.* [2013]. Quality of event identification among earthquakes with magnitude equal to or above $m_{\min}$. Blue (top): proportion of the total estimated background events with respect to the true number of background events. Green (middle): proportion of correctly identified triggered events. Red (bottom): proportion of correctly identified background events. The error bars are 95% prediction intervals (*not* the errors of the mean). The analysis is done for 10,000 independent realizations of declustering with $\alpha_0 = 0.1$ at every examined value of $m_{\min}$. The figure summarizes the results for 210,000 declustered catalogs.

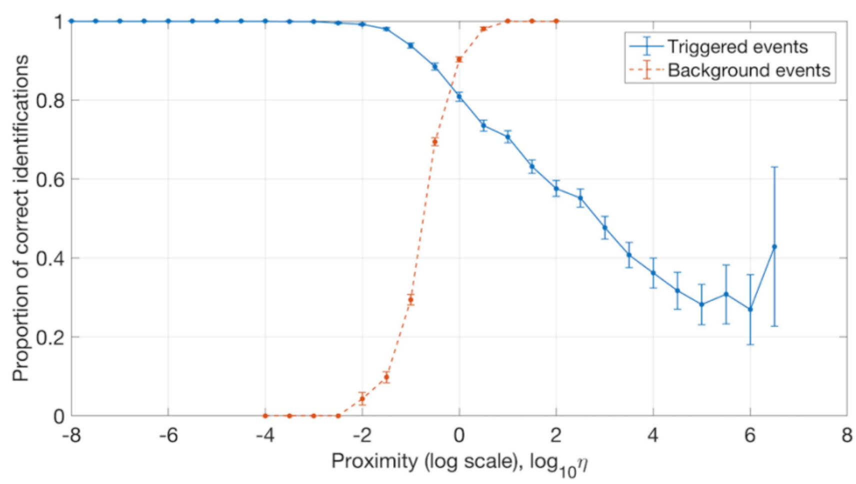

Figure S2: Declustering results for ETAS catalog of *Gu et al.* [2013]. Proportion of correctly identified triggered (solid blue line) and background (dashed red line) events, as a function of the proximity to the true parent or nearest neighbor, respectively. The analysis refers to a single realization of declustering.

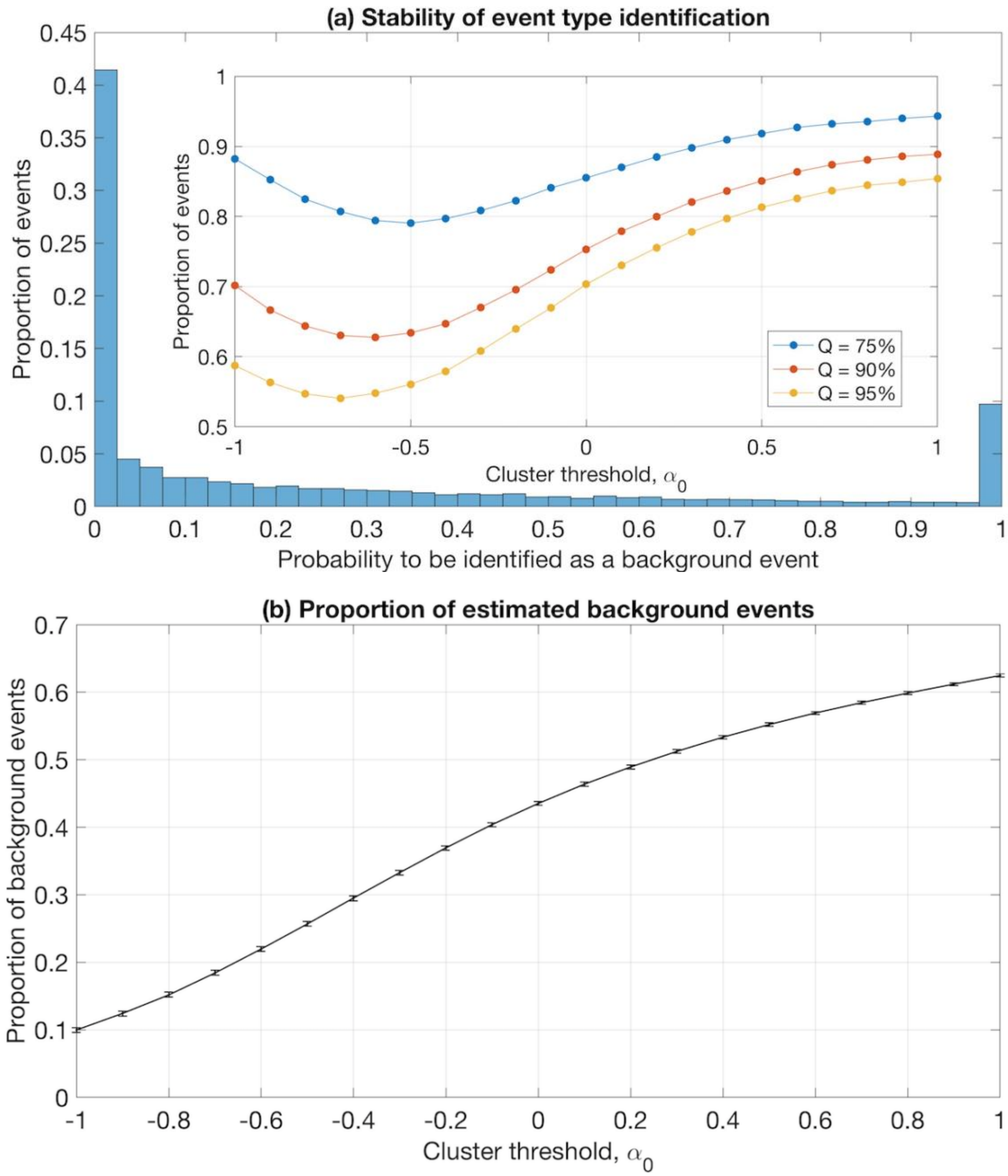

Figure S3: Declustering results in the global NCEDC catalog, $m \geq 5$. Stability of declustering. The analysis is done for 10,000 independent realizations of a declustered catalog for each value of cluster threshold α_0. (a) The main panel refers to $\alpha_0 = -0.5$. The rest of notations as in Fig. 5. The actual proportion of events that have the same estimated type in all 10,000 realizations is 9.3% for background and 11.8% for clustered events. This is hidden because of a finite bin width (0.025).

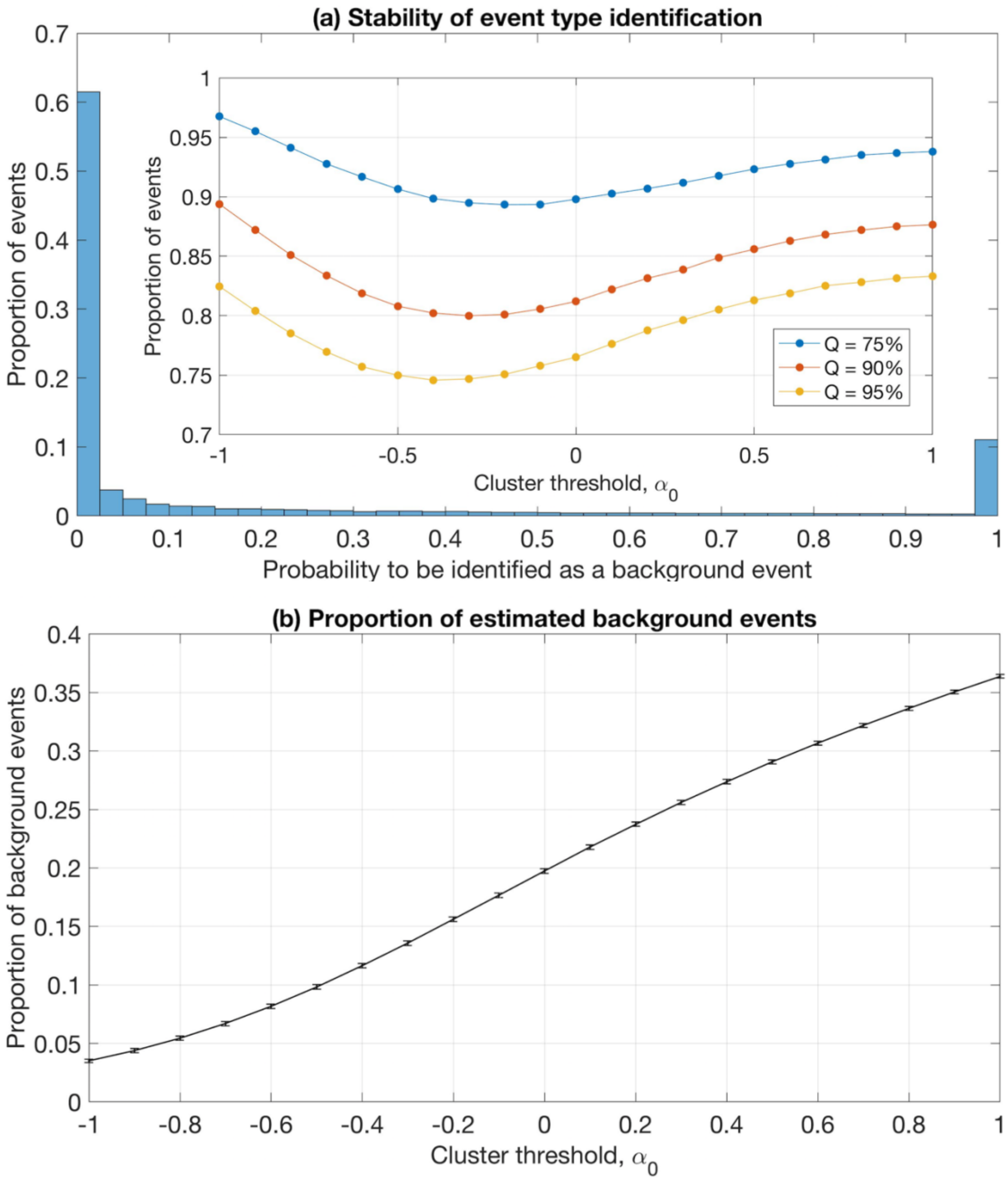

Figure S4: Declustering results for Southern California, $m \geq 2.5$, catalog of *Hauksson et al.* [2012]. Stability of declustering. The analysis is done for 10,000 independent realizations of a declustered catalog for each value of cluster threshold α_0. (a) The main panel refers to $\alpha_0 = 0$. The rest of notations as in Fig. 5.

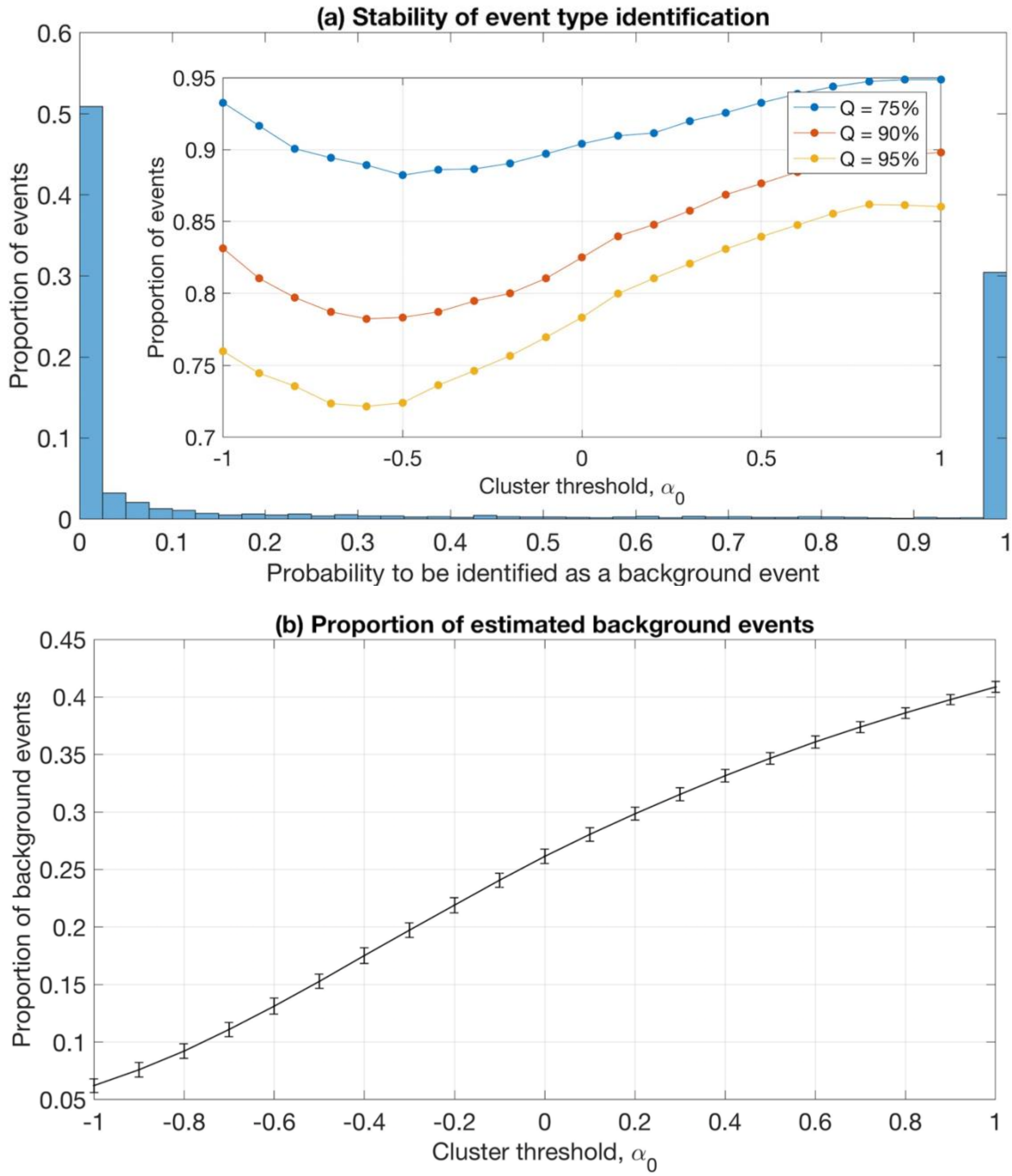

Figure S5: Declustering results for Southern California, $m \geq 3.5$, catalog of *Hauksson et al.* [2012]. Stability of declustering. The analysis is done for 10,000 independent realizations of a declustered catalog for each value of cluster threshold α_0. (a) The main panel refers to $\alpha_0 = 0.6$. The rest of notations as in Fig. 5.

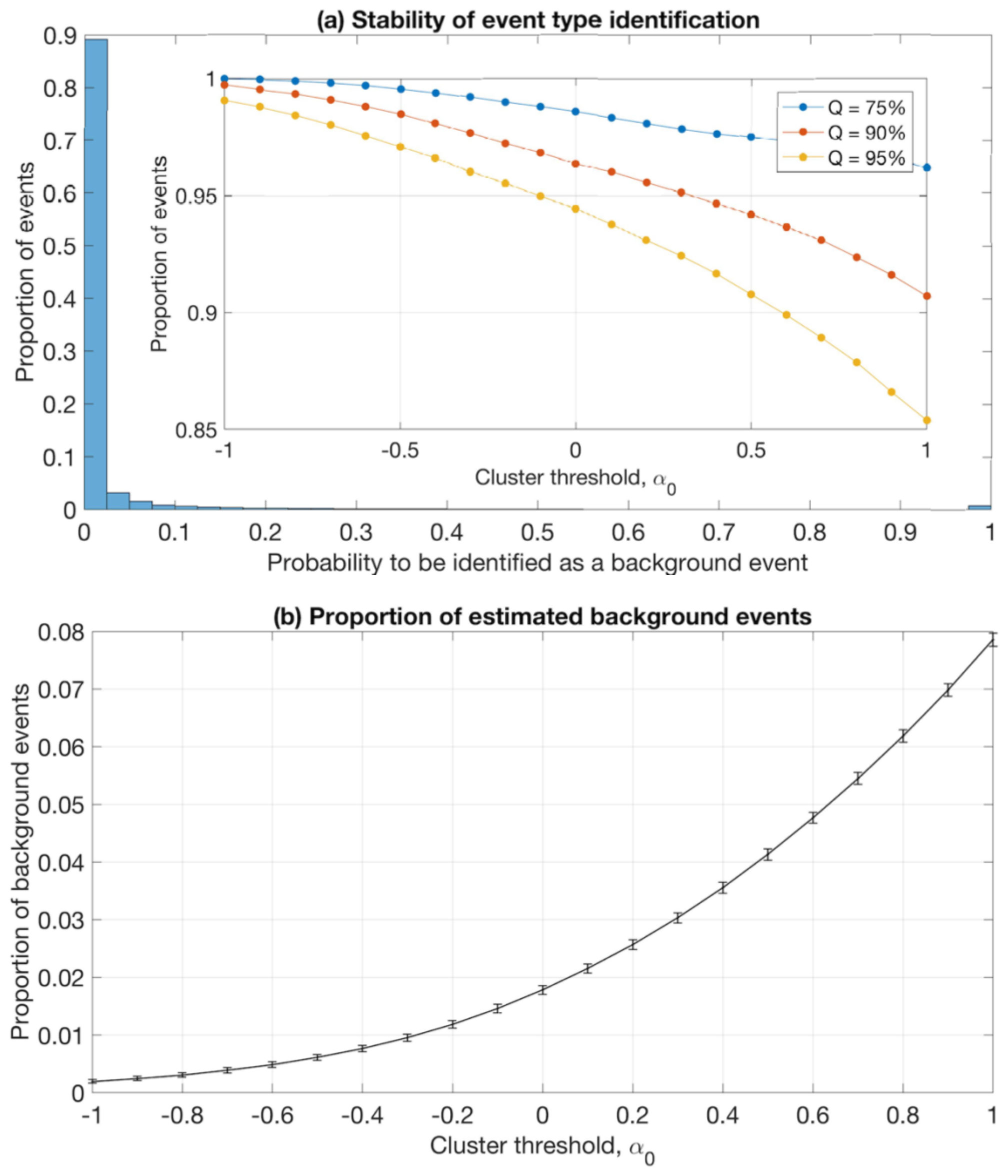

Figure S6: Declustering results for Landers (1992, M7.3) sub-catalog of *Hauksson et al.* [2012]. Stability of declustering. The analysis is done for 10,000 independent realizations of a declustered catalog for each value of cluster threshold α_0. (a) The main panel refers to $\alpha_0 = 0.2$. The rest of notations as in Fig. 5.

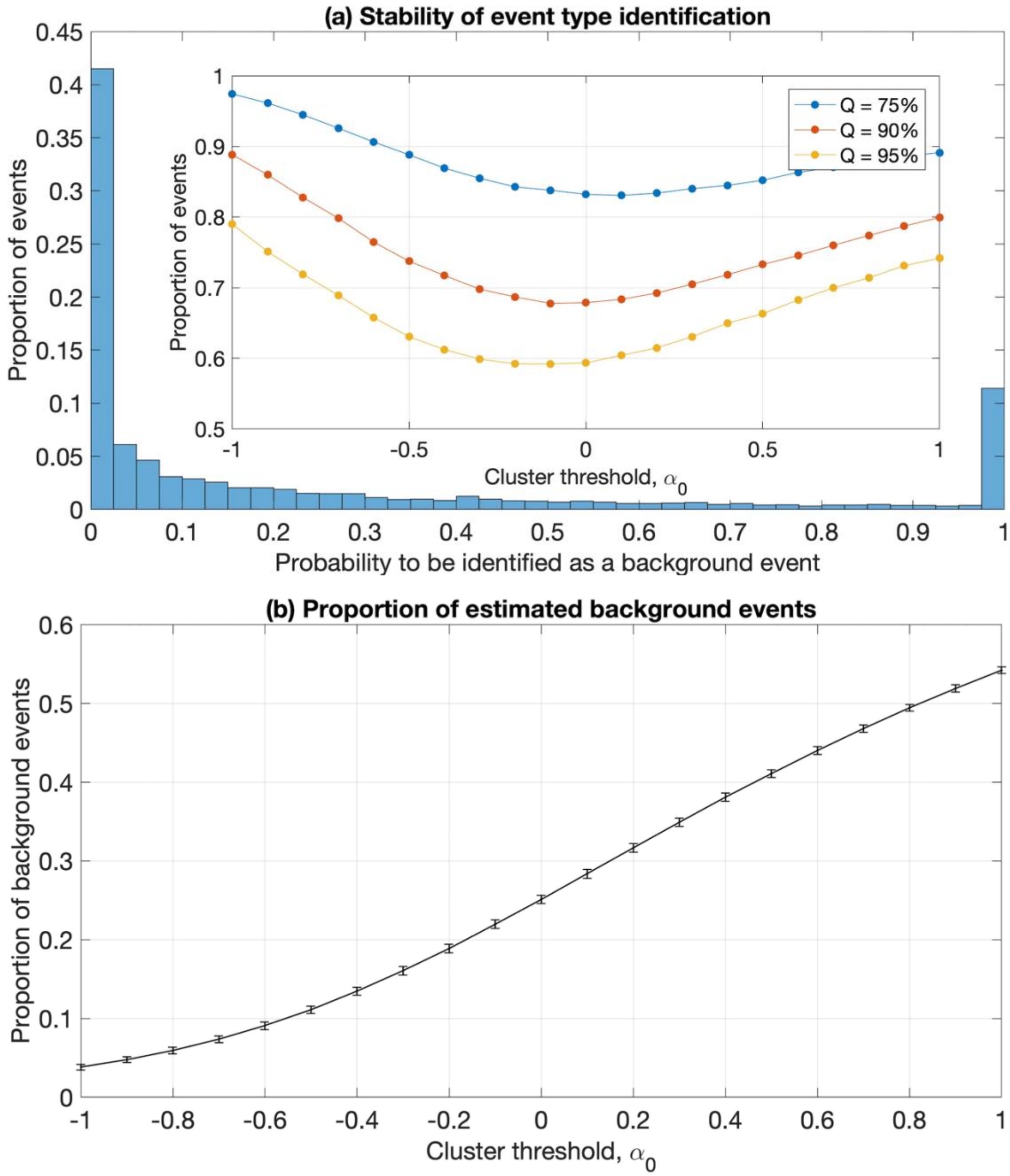

Figure S7: Declustering results for Parkfield (2004, M6) sub-catalog of *Waldhouser and Schaff* [2008]. Stability of declustering. The analysis is done for 10,000 independent realizations of a declustered catalog for each value of cluster threshold α_0. (a) The main panel refers to $\alpha_0 = 0.0$. The rest of notations as in Fig. 5.

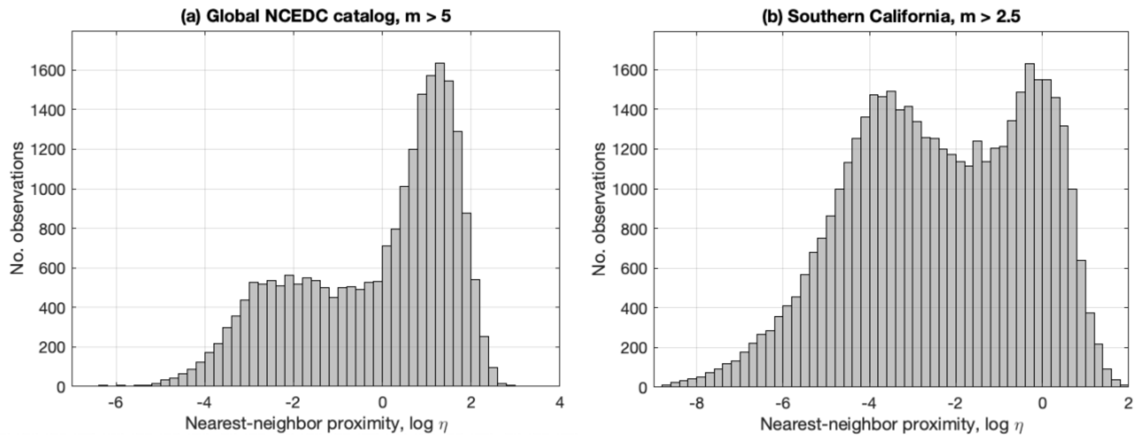

Figure S8: Bimodal distribution of the nearest-neighbor proximity. (a) Global NCEDC catalog, with $m \geq 5$; (b) Southern California catalog by *Hauksson et al.*, [2012]. (See **Sects. 2.1, 2.2** of the main text for complete data description).

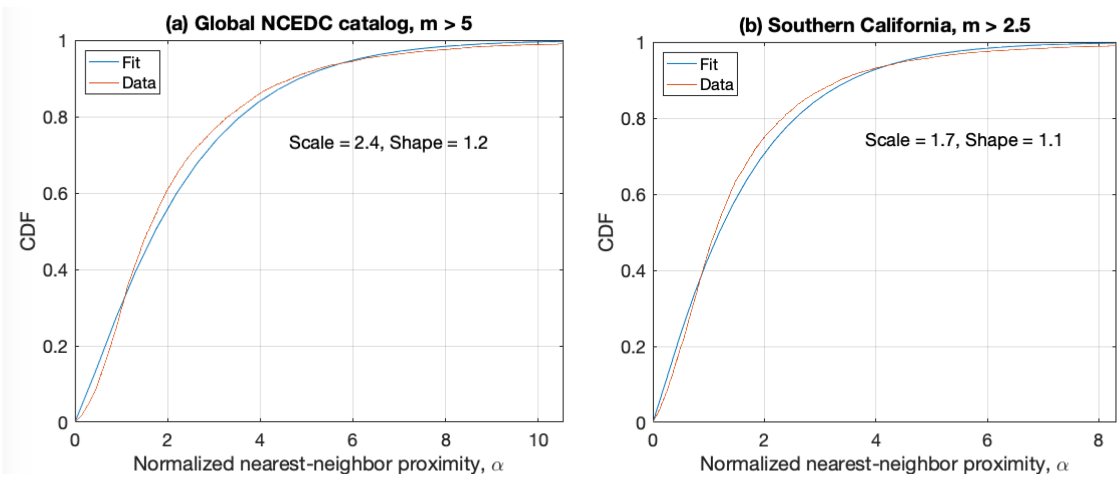

Figure S9: Weibull approximation to the normalized nearest-neighbor proximities after thinning. (a) Global NCEDC catalog, with $m \geq 5$; (b) Southern California catalog by *Hauksson et al.*, [2012]. (See **Sects. 2.1, 2.2** of the main text for complete data description).