Unsupervised Learning Used in Automatic Detection and Classification of Ambient-Noise Recordings from a Large-*N* Array

Michał Chamarczuk*[1], Yohei Nishitsuji[2,3], Michał Malinowski[1], and Deyan Draganov[2]

Abstract

We present a method for automatic detection and classification of seismic events from continuous ambient-noise (AN) recordings using an unsupervised machine-learning (ML) approach. We combine classic and recently developed array-processing techniques with ML enabling the use of unsupervised techniques in the routine processing of continuous data. We test our method on a dataset from a large-number (large-*N*) array, which was deployed over the Kylylahti underground mine (Finland), and show the potential to automatically process and cluster the volumes of AN data. Automatic sorting of detected events into different classes allows faster data analysis and facilitates the selection of desired parts of the wavefield for imaging (e.g., using seismic interferometry) and monitoring. First, using array-processing techniques, we obtain directivity, location, velocity, and frequency representations of AN data. Next, we transform these representations into vector-shaped matrices. The transformed data are input into a clustering algorithm (called *k*-means) to define groups of similar events, and optimization methods are used to obtain the optimal number of clusters (called elbow and silhouette tests). We use these techniques to obtain the optimal number of classes that characterize the AN recordings and consequently assign the proper class membership (cluster) to each data sample. For the Kylylahti AN, the unsupervised clustering produced 40 clusters. After visual inspection of events belonging to different clusters that were quality controlled by the silhouette method, we confirm the reliability of 10 clusters with a prediction accuracy higher than 90%. The obtained division into separate seismic-event classes proves the feasibility of the unsupervised ML approach to advance the automation of processing and the utilization of array AN data. Our workflow is very flexible and can be easily adapted for other input features and classification algorithms.

Cite this article as Chamarczuk, M., Y. Nishitsuji, M. Malinowski, and D. Draganov (2019). Unsupervised Learning Used in Automatic Detection and Classification of Ambient-Noise Recordings from a Large-*N* Array, *Seismol. Res. Lett.* **91**, 370–389, doi: 10.1785/0220190063.

Introduction

Arrays with ever-increasing station counts are fundamental in seismology (Rost and Thomas, 2002). Developments of the nodal technology to acquire seismic data by the oil and gas industry brought the concept of large-number (large-*N*) arrays to academia (Hand, 2014). Their applications include structural imaging, studies of seismicity, and monitoring (e.g., Lin *et al.*, 2013; Ben-Zion *et al.*, 2015; Quiros *et al.*, 2015; Karplus and Schmandt, 2018). Large-*N* arrays are often combined with long recording times, which facilitates seismic ambient-noise (AN) recordings (Karplus and Schmandt, 2018). AN is generally defined as a complex wavefield composed of the superposition of signals from natural and anthropogenic sources that are not generated specifically for the purpose of a study. Here, we address the issue of characterizing the AN content by developing automatic detection and classification of the various seismic events in the recorded wavefield. Because AN is recorded and stored during every regular continuous acquisition campaign (in particular, using nodal systems; Dean *et al.*, 2015), our methodology is a step forward to maximize the information from passive recordings.

The key technique using AN is called seismic interferometry (SI; Schuster *et al.*, 2004; Wapenaar and Fokkema, 2006; Draganov *et al.*, 2007; Wapenaar *et al.*, 2008; Schuster, 2009). SI allows the retrieval of virtual-source records by correlating noise recordings between pairs of receivers. SI is considered a cost-effective alternative for controlled-source operations, especially when terrain access is an issue. Successful applications

1. Institute of Geophysics PAS, Warsaw, Poland; 2. Faculty of Civil Engineering and Geosciences, Delft University of Technology, CN Delft, Netherlands; 3. Petro Summit E&P Corporation, Tokyo, Japan

*Corresponding author: mchamarczuk@igf.edu.pl

of AN SI can provide the velocity and structural information at the exploration scale (Draganov *et al.*, 2009), and at the crustal scale (Ruigrok *et al.*, 2010).

Imaging and monitoring of the shallow crust require high-frequency data, that is, high-frequency sources recorded with high sampling rate (Niu and Yamaoka, 2018). Moreover, dense geophone arrays deployed in areas of abundant noise activity such as operating mine sites or volcanoes enable unaliased spatial sampling of the noise-source distribution characteristic for such high-seismicity areas (Rost and Thomas, 2002). For such continuous event-rich AN recordings, the preferred processing approach should be automatic and require minimum human interaction (Hansen and Schmandt, 2015). At the same time, most conventional array-processing techniques require high signal coherency across the array, implying important constraints on the array geometry, spatial extent, and data quality (Almendros *et al.*, 1999). Therefore, for an existing dataset, the main interest is to optimize the processing time (including tuning array-processing parameters), especially in cases where months of human work could be necessary.

Detection and classification of seismic signals using machine-learning (ML) already has a well-established history (Dowla *et al.*, 1990; Dysart and Pulli, 1990; Wang and Teng, 1995; Del Pezzo *et al.*, 2003; Wiszniowski *et al.*, 2014). These studies were a step toward developing effective ML techniques for distinguishing tremors and earthquakes (Nakano *et al.*, 2019), geyser-eruption signals detection (Yuan *et al.*, 2019), earthquake early warning (Li, Meier, *et al.*, 2018; Kong *et al.*, 2019), and many automatic approaches for accurate earthquake-parameter estimation (Böse *et al.*, 2008; Meier *et al.*, 2015; Cuéllar *et al.*, 2018; Ochoa *et al.*, 2018), including the almost separate branch of accurate phase-picking methods (Chen, 2018; Zhu and Beroza, 2018).

From the previously mentioned applications of seismic arrays, imaging studies (Araya-Polo *et al.*, 2018), monitoring volcanic tremors (Malfante *et al.*, 2018), and earthquake early warning (Kong *et al.*, 2016; Li, Meier, *et al.*, 2018) are a few of the most interesting and challenging areas for employing ML techniques. An inherent processing part related to some of the ML applications in seismology is the detection and classification of specific event types (Rouet-Leduc *et al.*, 2017; Zhou *et al.*, 2019), for example, signals (tremors) in volcano monitoring (Bhatti *et al.*, 2016), specific precursors observed in the seismograms in earthquake early warning (Minson *et al.*, 2018), and body- or surface-wave events for AN SI imaging studies (Nakata *et al.*, 2016). For some of these signals such as the volcanic tremors or body-wave events, the sole detection might not suffice because the waveforms of the different event types are similar to each other in the time domain. They might be differentiated only by minor features when transformed using signal representations (i.e., transformed to other domains) such as Fourier transform, envelope, autocorrelation, and kurtosis to name just a few from the long list of signal transformations used in the ML detection context (see the review in Malfante *et al.*, 2018). These applications demand more detailed classification and assessment of AN event types performed in real time. Such processing appears to be a good target for combined ML and array-processing techniques. A hybrid approach that combines ML and array processing is an emphasized and anticipated development in seismology (Kong *et al.*, 2018; Bergen *et al.*, 2019).

Motivation

Here, we focus on the performance of ML for detecting and assessing different categories of seismic events present in continuous AN recordings. We rely on the fact that the appearance of AN in seismic records differs in amplitude and frequency and that the various kinds of signal transformations derived therefrom provide varying characteristics (Bormann, 1998). We aim for evaluation of the feasibility of unsupervised clustering methods using these differences to track down various AN events without the need to know their exact representation in different domains. These characteristics can be retrieved using array-processing techniques (Rost and Thomas, 2002). This idea is applied to data recorded using a large-N array above the Kylylahti active underground mine in eastern Finland for testing AN SI for mineral exploration. The same dataset has already been used to demonstrate extraction of body-wave events using supervised ML (support-vector machine) in combination with a two-step wavefield evaluation and detection method (Chamarczuk *et al.*, 2019). With this hybrid approach, the authors made a binary classification to discriminate between body- and surface-wave events. They applied two-step wavefield evaluation and detection method on a small portion of the AN recordings to obtain labels (in this study, label is the type of recorded seismic event) and then used the labels as input to a support-vector machine to classify the remaining part of the data. In our study, we do not provide input labels and aim to discriminate between several (>2) classes of seismic events and thus provide a more detailed description of the recorded AN data. In addition to the lack of labels, typical for unsupervised clustering, we treat the number of clusters as an unknown parameter to be established (we refer to this approach as "blind clustering"). In blind clustering, we estimate the range of optimal number of clusters by comparing the results of k-means for a broad range of clusters (1–81) and input-parameter subset sizes (1–81). The upper limit of evaluated range of clusters and parameter subsets is related to the discretization of input features used in this study. Then, in the "constrained clustering," we use a computationally heavier optimization method, which is expected to give us better prediction accuracy, in the already narrowed range of input parameters (provided by the blind clustering) to find an optimal value of the number of clusters.

The article is organized as follows. In the Method section, we introduce the hybrid workflow combining array processing and unsupervised clustering of continuous recordings. We also explain the motivation behind each processing step and briefly

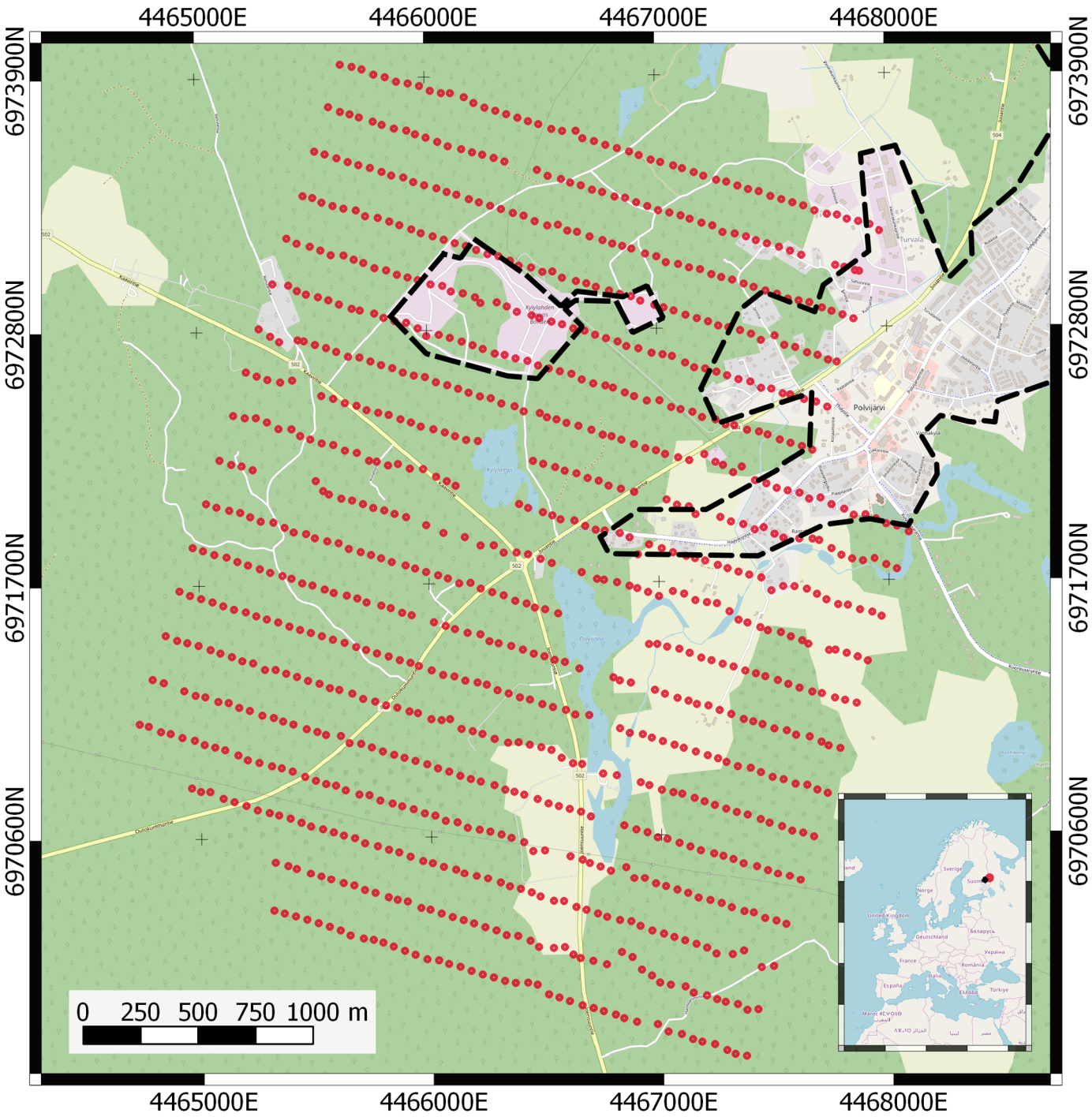

Figure 1. Layout of the Kylylahti array. Receiver stations are denoted with red dots; dashed black lines show the mine and city area. (Inset) The study area in a map of Europe. The color version of this figure is available only in the electronic edition.

describe the array-processing and ML techniques we propose in our workflow (for more detailed technique explanation, see Appendices A and B). In the Dataset section, we describe the Kylylahti array (shown in Fig. 1) we use to benchmark our methodology. The Results section describes the outcome of applying our methodology to the Kylylahti dataset. The subsections in this section are named in the same way as the processing steps in our workflow. We focus on the key processing steps related to the ML module of our workflow. The results from the array-processing module and temporal cluster analysis (supporting processing step in the ML analysis module) are described in the Appendices C and D, respectively. In the Discussion section, we focus on the flexibility of our workflow and explain the possibility to test various other array-processing techniques, and unsupervised clustering techniques. We mention also other datasets and applications

suitable to test our methodology. In the Conclusions section, we summarize our experiences from applying unsupervised clustering in automatic detection and classification of seismic events recorded by Kylylahti array, and conclude with general statement about feasibility of this approach.

Method

Hybrid workflow

Our hybrid methodology consists of two parts: (1) array processing and (2) ML analysis (see workflow summary in Fig. 2). The array processing provides input to the ML part.

Outputs of individual array-processing techniques are directly related to the measurable characteristics of AN data. Thus, following ML community we refer to these as input features (Bishop, 2006). To highlight individual values forming the entire output of a given processing technique, we use the term input parameter.

After having acquired the data, the first fundamental step is Input-Feature Selection, in which the decision is made about input features representing the AN data and, consequently, the array-processing techniques to be used for extracting those features. In the Data Preprocessing step, the continuous data are separated into shorter segments (noise panels) and processed using conventional AN preprocessing techniques (see e.g., Bensen *et al.*, 2007) to apply array-processing techniques selected in the previous step. Subsequently, input features are extracted from the data (Input-Feature Extraction step). Next, in the Data Augmentation step, we adjust the output of array-processing techniques to be used in unsupervised clustering (see Fig. 3 for input-feature extraction and data augmentation scheme for location input feature).

The key step in the ML analysis module is the unsupervised clustering. It consists of two parts: (1) the blind clustering and (2) constrained clustering. In the blind clustering step, we use *k*-means to estimate the range of optimal number of clusters and size of input-parameter subset. Then, in the multicluster feature selection (MCFS; Cai *et al.*, 2010) processing step, we use the MCFS method to determine which input parameters should be selected for creating the input-parameter subset. The values estimated in the blind clustering are used to perform constrained clustering. In the constrained clustering step, we apply the *k*-means method again, but this time by providing the single optimal number of clusters using the subset of input features indicated by MCFS. Finally, cluster quality control (QC) is performed using visual inspection and silhouette measure. In Figure A1, we show the expanded version of the workflow with details on the selected parameters.

Array-processing and ML techniques used in this study

Here, we briefly describe the array-processing techniques and ML tools employed in our hybrid workflow (see Appendix A for a more detailed description of selected array-processing techniques).

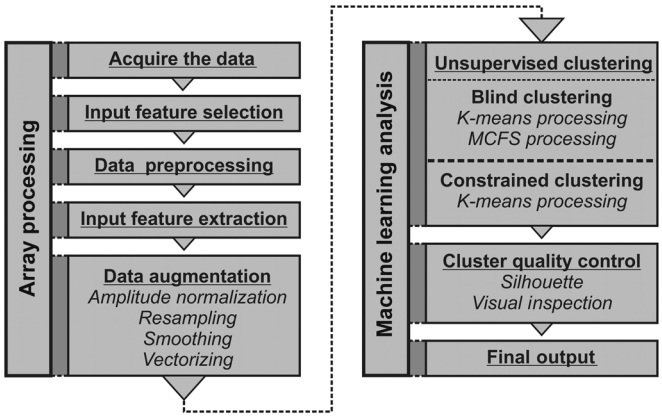

Figure 2. Sketch of the array-processing and unsupervised-clustering hybrid workflow. The color version of this figure is available only in the electronic edition.

Seismic AN can be described by its dominant features: frequency, velocity, directionality, and energy (Bormann, 1998). To quantify these characteristics, we select the following array-processing techniques: beamforming for determining azimuth and velocity, InterLoc (Dales *et al.*, 2017) for location, and power spectral density (PSD) for frequency and energy. We use the conventional delay-and-sum beamforming (Johnson and Dudgeon, 1993) which is based on summing the signal amplitudes along assumed travel paths for determining azimuths related to the strongest sources. PSD is defined as Fourier transform of the signal autocorrelation function and provides the energy of the signal at each frequency component. InterLoc is a spatiotemporal tool similar to beamforming, but instead of scanning azimuth and velocities, it scans the different location points and the input comprises cross-correlated waveforms instead of signals in the time domain (see Fig. A3 for the output of all three techniques calculated for one day of recordings).

Our basic tool for dividing the AN data into clusters is the *k*-means algorithm (Lloyd, 1982). The *k*-means is a tool for partitioning data points into predetermined number of clusters by assigning each sample to its closest cluster center (defined as mean value of all the points within a cluster). In the blind clustering step, we address the issue of estimating the number of clusters which needs to be predetermined. We find this optimal value of clusters by running the *k*-means algorithm multiple times and finding the best solution (this approach is called elbow test). In the constrained clustering step, we additionally provide (1) number of clusters, (2) size of input-parameter subspace, and (3) selection of those parameters. For assessment of the clustering results, in the cluster QC step, we apply the silhouette method, which is a visual tool for QC of individual clusters (Kaufman and Rousseeuw, 1990). More detailed description of the ML techniques used in our workflow is provided in Appendix B.

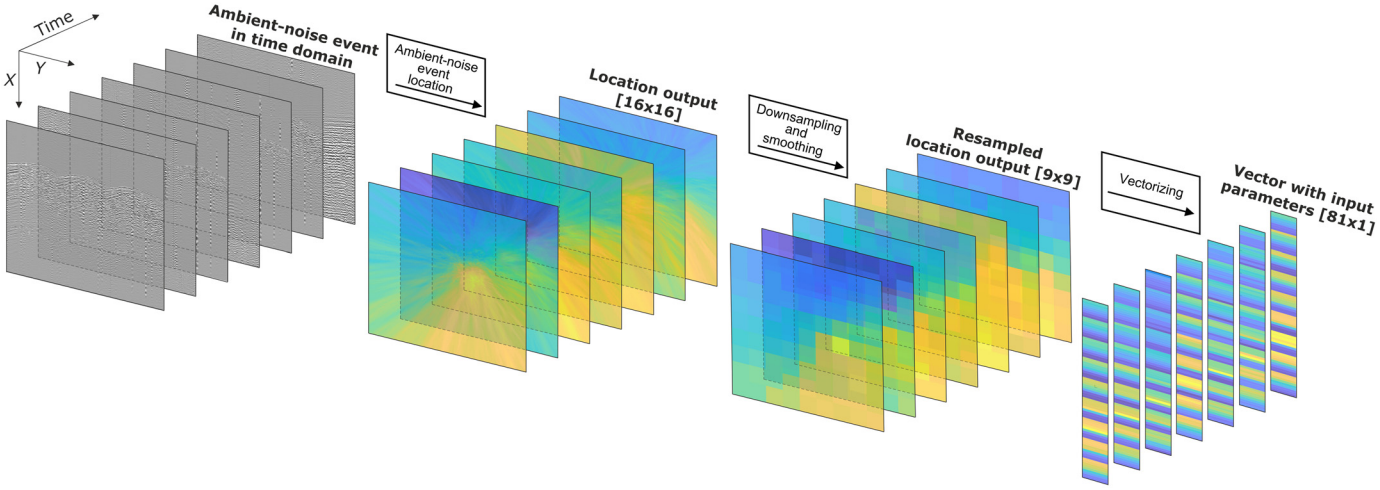

Figure 3. Input-feature extraction and data augmentation scheme for the location feature domain. The color version of this figure is available only in the electronic edition.

Dataset

The Kylylahti large-*N* array (see Fig. 1 for the layout) was deployed in the vicinity of the polymetallic underground Kylylahti mine in Polvijärvi (eastern Finland) as a part of the Cost-effective Geophysical Imaging Techniques for supporting Ongoing MINeral exploration in Europe (COGITO-MIN) project. Its primary purpose was to advance the development of AN SI imaging techniques for mineral exploration and provide a baseline for testing novel array-processing techniques. The Kylylahti array was formed by 994 receiver stations distributed regularly over the 3.5 × 3 km area with 200 m line spacing and 50 m receiver interval. Surface conditions varied from exposed bedrock to swamps. Each receiver station consisted of a Geospace seismic recorder and 6 × 10 Hz geophones bunched together and buried whenever possible, was recording at a 2 ms sample rate for about 20 hr per day for about 30 days. As a result, more than 600 hr of AN data per each receiver were recorded.

Results

Blind clustering

In this section, we estimate the range of optimal cluster number and determine the content of input-parameter subset.

***K*-means processing.** In the blind clustering step, we apply the *k*-means to the Kylylahti-array data. The clustering performance is evaluated by automatically running the *k*-means procedure with an initial number of clusters ranging from 1 to 81, using randomly generated input-parameter subsets (see Appendix D for the details on subset generation) with sizes ranging from 1 to 81 (maximum number of input parameters for each input-feature domain). We separately analyze the best number of clusters for each input-feature domain (beamforming, location, and PSD) depending on the number of input parameters. The results for this exercise are shown in Figure 4.

From the results shown in Figure 4, we obtain a limited range of optimal numbers of clusters (k) and numbers of input parameters to investigate. The *k*-means processing in blind unsupervised step provides us the reasonable range of *k*-values to inspect and, potentially, the optimal sizes of the input-parameter subsets. In general, choosing different numbers of input parameters affects the retrieved number of clusters (e.g., for four input parameters the optimal cluster number is 20, and for 11 input parameters it is 39; see Fig. 4b). Thanks to the blind clustering analysis summarized in Figure 4, we did not further have to test all of them but only the limited range that provides stable results (between 25 and 81 input parameters). Apart from this observation, analysis of Figure 4 indicates that, depending on the selected subset of input features, different numbers of clusters are indicated as optimal, but most of the subsets provide a number of clusters close to the average value of all evaluations. This average value (from hereafter referred to as optimal) of number of clusters quickly reaches the stable level (the values between the dashed lines in Fig. 4) of ∼40 for every input feature. The convergence to around 40 clusters is reached at 25 input parameters, and interestingly this trend appears for every input feature (compare plots in Fig. 4a–c). The fact that we can observe convergence to the optimal *k*-values for relatively small numbers of input parameters suggests that the computational effort of unsupervised clustering can be reduced using only 25 instead of 81 input parameters. These numbers might differ when other steps (e.g., different input features are used) are adapted. The analysis of input parameters derived from input features we use in this study (see Appendices B and C; for the detailed comparison of input features used in this study) indicated that PSD does not provide sufficient differentiation of seismic events in the Kylylahti area. Basing on the temporal changes of the input-parameter values (see Fig. A3), and cluster membership (see Fig. C1), for further analysis we use only the location and beamforming feature domains.

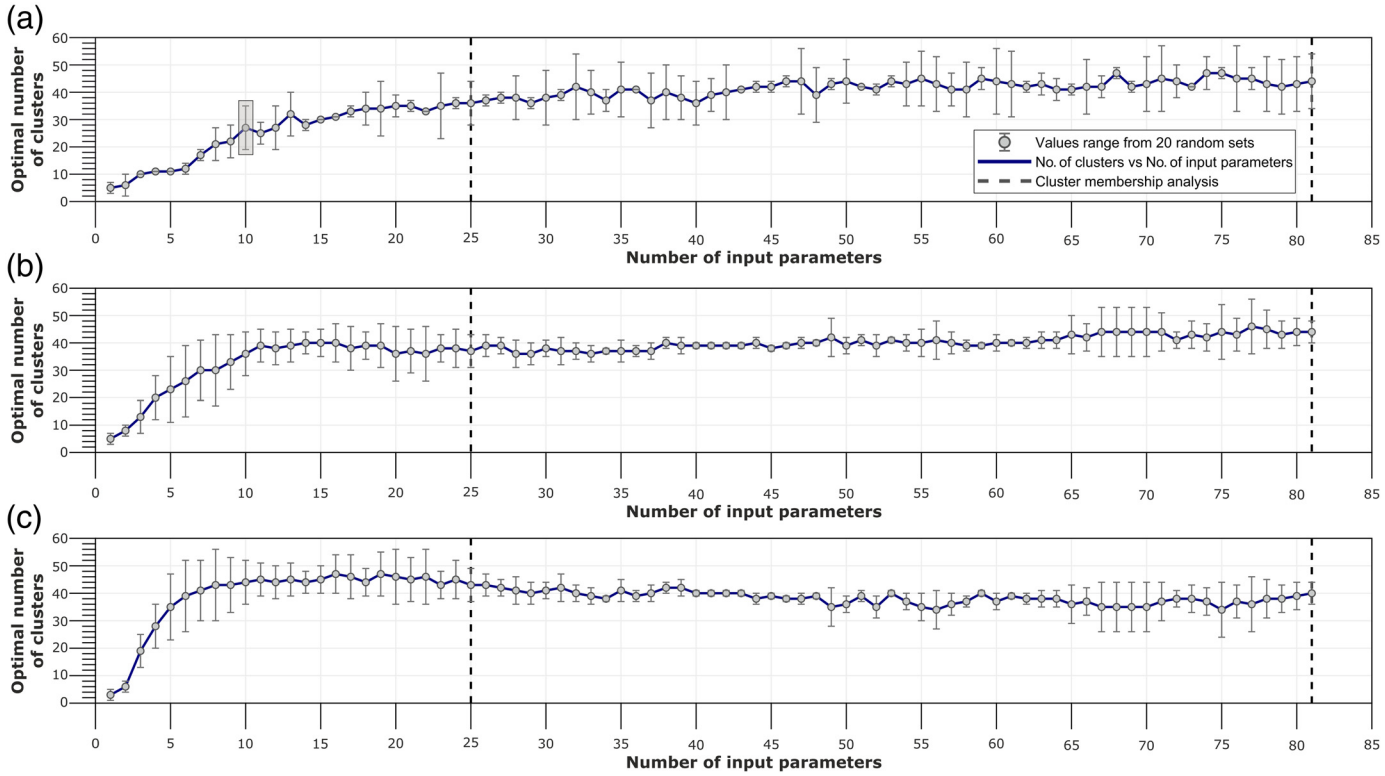

Figure 4. Optimal number of clusters for different number of input parameters obtained from (a) location, (b) beamforming, and (c) power spectral density feature domains. Gray dots represent the mean results from the elbow test run on 20 random combinations of input parameters. Vertical gray intervals indicate the range of results obtained from 20 tests. Dashed lines denote the range of input-parameter subsets selected for further analysis. The transparent gray rectangle in (a) indicates the result for which we show the procedure of random input parameter selection in Figure D1. The color version of this figure is available only in the electronic edition.

MCFS processing. From Figure 4, we can see that for the majority of the tested input-parameter numbers the difference between the resulting k-numbers is small, that is, we end up having ~40 different clusters regardless of using 25 or 80 input parameters. This observation might be biased because we have still not evaluated all possible combinations of input parameters. To select the best subset of input features from the two input-feature domains (location and beamforming) and 162 total input parameters, we used the MCFS technique. The procedure of MCFS processing is described in Appendix C. In Figure 5, we show the most significant input parameters (results of the MCFS processing step) projected from the downsampled grid (see Fig. C2b for location of input parameters on a resampled grid) on feature maps. Figure 5a shows location input parameters with the highest significance mapped on a nonresampled grid. The maxima (warm colors) occur at the mine and city areas (gray lines); however, we can also see the presence of high-significance grid points in areas not related to the currently known AN sources. The beamforming plot (Fig. 5b) shows significant parameters at angles consistent with the location of AN sources visible in the location map. Apparent velocities related to seismic energy from these directions cover a wide range (~2–10 km/s). Again, in addition to the known, expected sources of noise, Figure 5b indicates the presence of energy with velocities of ~3 km/s coming from the southwest direction and very low-velocity events indicated by the high-amplitude edges of the beamforming map. The latter is intuitive because beamforming with a plane-wave assumption means averaging the wavefield along straight azimuthal paths (Rost and Thomas, 2002), thus it might enhance recorded plane-wave events (usually having low frequency) traveling with velocities determined by the near-surface layers. The results confirm the fidelity of MCFS processing and indicate that unsupervised clustering can detect AN seismic events related to dominant noise sources.

Constrained clustering

In this step, we obtain the final cluster membership (i.e., optimal number of clusters) by performing the k-means analysis using parameters indicated in blind clustering step. As input to the constrained clustering step, we use a subset of 40 parameters of the highest significance derived from both the location and beamforming domains (38 parameters with the highest significance and two parameters with hits oscillating around 30; see Fig. C2a). To establish the final, optimal k-number for our study, we use silhouette analysis and highlight the best

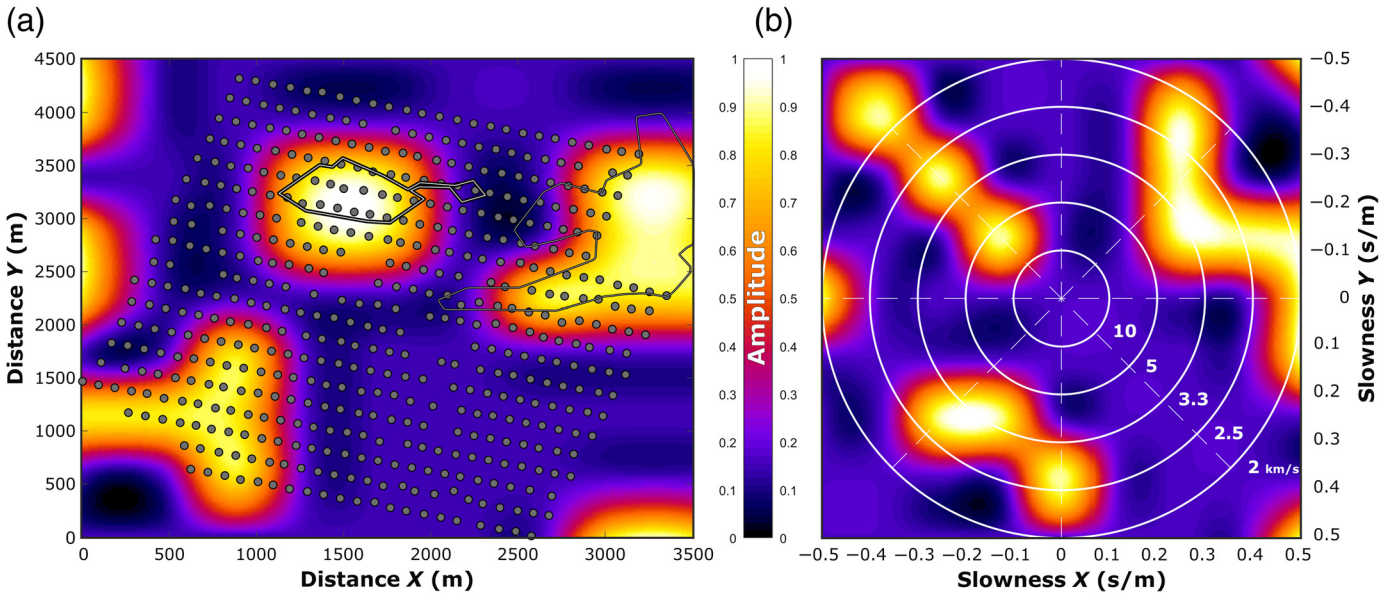

Figure 5. Multicluster feature selection results for location and beamforming feature domains. Result of 2D interpolation of values shown in Figure C2b on (a) location and (b) beamforming plots. The color version of this figure is available only in the electronic edition.

k-number using average silhouette values calculated for 11 k-numbers ranging from 35 to 45. For further analysis, we select $k = 40$ because it exhibits the highest average silhouette value. Figure 6a shows the silhouette plot for $k = 40$.

Cluster QC

Silhouette analysis. The general overview of cluster quality and their size can be analyzed on the silhouette plot and bar plot showing cluster sizes (Fig. 6a and 6b, respectively). The cluster sizes on the bar plot (Fig. 6b) are reflected in the width of the silhouette results (Fig. 6a) for each cluster. To obtain more insight into the clustering behavior with time, in Figure 6c we show which clusters are appearing at which time instances (hours of recorded AN).

The plots in Figure 6 allow us to draw first conclusions about the cluster structure exhibited by the Kylylahti data. First, the cluster size indicates whether we deal with a common event type expected in the recording area (surface-activity events) or unusual events related to specific activities (e.g., underground blasts or active seismic shooting). Second, the silhouette plot provides the consistency of each cluster. Higher silhouettes indicate a similarity of events from the same cluster, which are thus more likely related to repetitive sources. The plot in Figure 6c visualizes how many different clusters appear at each hour, which may allow relating specific event types with *a priori* knowledge about AN sources in the recording area. For the Kylylahti-array data, the silhouette values are relatively low (Rousseeuw, 1987; maxima ~0.4), and we accordingly use the presence of negative silhouette scores as the threshold criterion. From the 40 clusters, we select 10 clusters with positive silhouette values only, denoted with dark colors in Figure 6a. The selected clusters are shown on the bar plot (darker bars in Fig. 6b) and marked with bold roman numbers in Figure 6c. A representative event for each evaluated cluster is shown in Figure 7.

Visual inspection. The final step of the ML module is the visual inspection of the cluster content. We visually inspect the content of clusters whether the events present in one cluster represent the same type. These events can be further described in terms of human perception. For the clusters with only positive silhouette values the detection accuracy is ~90%, whereas for the clusters with negative silhouettes the score accuracy varies between 70% and 85%. By accuracy, we mean the ratio of the number of events that match the same type to the size of a cluster. After verifying the content of the clusters with only positive silhouettes, we can refer to them as event types.

Final output

AN events related to clusters I–V are mainly surface waves. Clusters IV and V contain the majority of the events and are the most common type of seismic events in the Kylylahti area. Cluster IV represents random noise without coherent energy. Cluster V mostly contains events related to surface waves generated by the persistent activity of mine ventilation, manifested as very strong air waves. A similar type of event was reported in other AN studies (e.g., Cheraghi *et al.*, 2015). Surprisingly, this event occurs more frequently than the incoherent noise (cluster IV), which indicates the dominant influence of mine-related noise in the Kylylahti area. The events in cluster VI are plane surface waves likely coming from a distant open-pit mine. Cluster VII contains events from active shots, and clusters VIII–X contain body-wave events due to underground

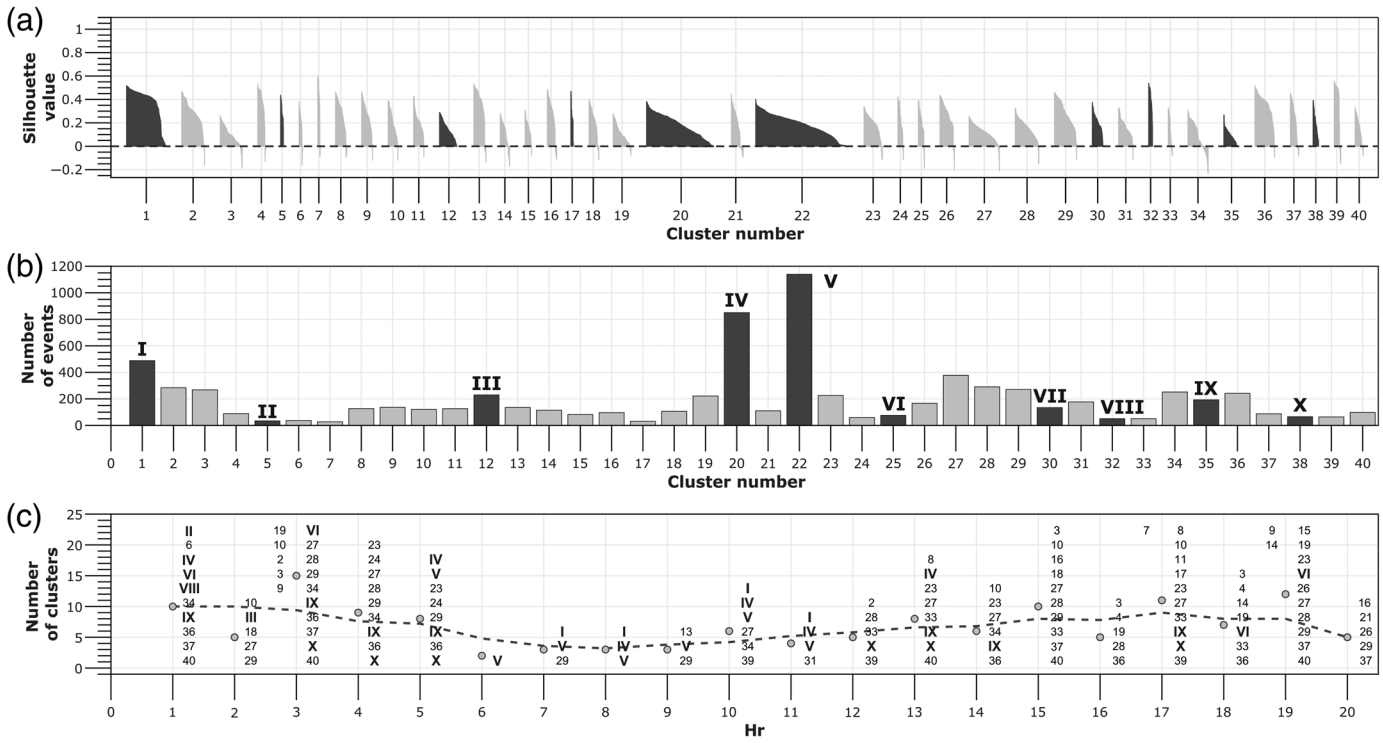

Figure 6. Quality-control plots for the constrained unsupervised clustering. (a) Silhouette result for the *k*-means clustering using $k = 40$; darker colors denote clusters with only positive silhouette values. (b) The number of ambient-noise (AN) events for the determined clusters; clusters selected for visual inspection are denoted with darker bars. (c) The number of clusters (shown with gray dots) and their incidence for every time point of recorded data for one day; clusters selected for evaluation are denoted with bold roman numbers. The gray dashed line connecting number of cluster values reflects the change of AN wavefield diversity in the recording area.

mine activity. Events from clusters VIII–X exhibit similar hyperbolic moveout but differ in the number of visible phases and apparent velocities. Based on these results, clusters VIII–X are preferred for SI reflection imaging (possibly also for direct reverse vertical seismic profiling, see e.g., Quiros *et al.*, 2017). Clusters related to surface sources (excluding noise generated by the mine) would be a useful input for AN surface-wave tomography. Events from cluster V could be used as an input for adaptive filters to suppress repetitive noise, which is otherwise difficult to remove in the frequency–wavenumber domain (Roots *et al.*, 2017).

Discussion

The hybrid workflow presented in this study is a generic method for detection and classification of seismic events in the continuous recordings. Here, we discuss the potential modifications in terms of applying different processing techniques and indicate other targets datasets feasible to apply our methodology.

Streamlining the hybrid workflow

Processing steps indicated in Figure 2 can be easily modified: in the array-processing module, alternative input features and array-processing techniques might be considered; and in the ML analysis module: the unsupervised clustering algorithms and their parameterization as well as cluster QC tools might be replaced.

The scope of this study is not related to the comparison of the performance of different techniques, as our goal was to investigate the general feasibility of unsupervised clustering in AN events classification. However, we indicate other solutions which might improve the detection and classification rate of our workflow; in particular, to adapt it for other datasets. For further discussion on this, please see Appendix E.

We expect that biggest improvement of our methodology might be related to the selection of clustering technique in blind and constrained clustering steps. As indicated in Appendix B, the characteristic features of *k*-means are (1) the necessity for providing the initial number of clusters, (2) using mean value to obtain the centroids, and (3) using specified distance metric. Exemplary unsupervised techniques which might be considered as replacement for *k*-means are (a) *k*-medians, (b) mean-shift clustering, and (c) hierarchical clustering. See Appendix E for more elaborate discussion about difference between these algorithms. Here, we mention some unsupervised clustering algorithms which propose other solutions and may provide different clustering results.

As indicated by comparison of time complexity of unsupervised algorithms (see Appendix E), all of these techniques are much more computationally heavy as compared to *k*-means

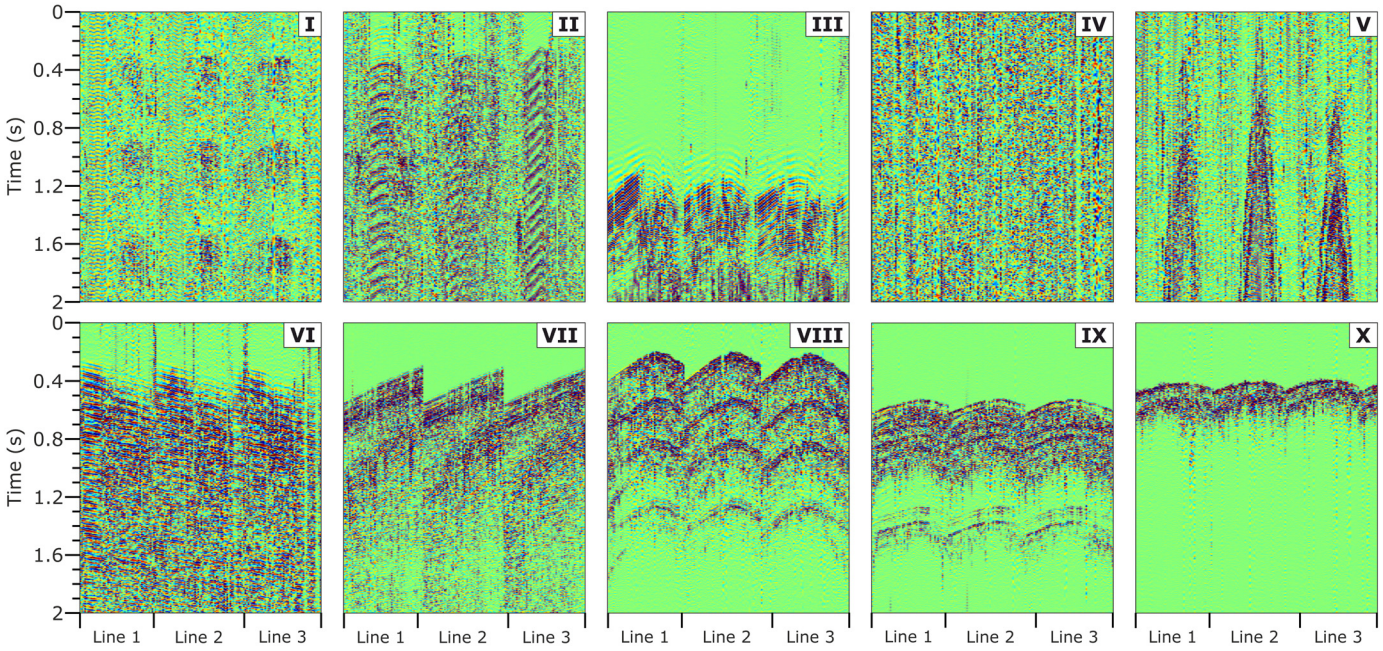

Figure 7. Representative events for the clusters indicated darker bars in Figure 6b. The color version of this figure is available only in the electronic edition.

(Steinbach *et al.*, 2004). *K*-median exhibits the same time complexity as *k*-means; however, because computing the median involves sorting the data points in each iteration it is still more time consuming than *k*-means. The computational time superiority of *k*-means algorithm makes it the natural candidate to evaluate the general performance of unsupervised approach for large seismic datasets (assuming we consider the unsupervised clustering as a robust tool for initial characterization of data, prior to more detailed evaluations). On the other hand, when the accurate events catalog serving as input labels for supervised studies is needed, then the aforementioned methods might be tried. Another solution is to combine two different algorithms into single workflow. Such approach allows to account for limitations of different algorithms. For instance, *k*-means can be applied for initial data clustering (as in blind clustering step in our workflow) and then mean-shift clustering can be performed on reduced dataset (using only the data points being cluster centers indicated by *k*-means). By such approach, the risks of (1) choosing wrong initial cluster number in *k*-means (by merging the surplus clusters with mean-shift), and (2) producing small uncertain clusters (*k*-means indicates cluster with high number of data points) are reduced. Toward adapting the unsupervised clustering of AN events as the routine seismological processing workflow, a study comparing the performance of different clustering methods should be undertaken.

Potential applications

The primary potential application of the approach we proposed here is for arrays deployed in areas where little or no prior knowledge is available about the AN content, for example, during site-assessment recordings (Wilmore, 1979), AN SI imaging studies in remote areas (Draganov *et al.*, 2013), or extraterrestrial terrains (Nishitsuji *et al.*, 2016). In such cases, assumptions in terms of data processing and detection thresholds need to be limited to a minimum, and the detection process must be based on data-driven differences between event representations in preselected transformed domains.

A second application is for volumes of AN recordings that require careful inspection of continuous recordings segmented into many short time windows (e.g., very-long continuous recordings coming from areas with an abundance of seismic activity; Hansen and Schmandt, 2015). Several successful studies have already shown the potential of ML techniques in such a context, with convolutional neural networks being the most promising technique (e.g., Perol *et al.*, 2018; Woollam *et al.*, 2019; Wu *et al.*, 2019). However, most of these studies required initial knowledge about the AN event types present in the recording area to use as labels in supervised ML. When such knowledge is unavailable, unsupervised ML should be considered primarily as a tool that provides labels (analyzing a small portion of the data) that support the more detailed supervised ML techniques. We aim for a methodology that is easily adaptable to different types of seismic arrays, with sparser spacing, less recording nodes, and deployment in less noisy or even remote areas. Regional-scale seismological arrays would be the next candidate to use to test our method.

Conclusions

In this study, we presented a hybrid-approach methodology for unsupervised clustering of AN events recorded by a large-*N*

array. The methodology allows the detection of multiple event classes and selectively using the events for example, reflected body-wave imaging or surface-wave tomography with seismic-interferometry techniques. Our hybrid approach combines (1) array-processing techniques that provide spatiotemporal characterization of continuous AN data and (2) ML techniques that rely on the array-processing outputs. We applied our methodology to a subset (20 hr) of data that was acquired using a large-*N* array deployed over the Kylylahti mine (eastern Finland). Using array-processing techniques, we obtained directivity, velocity, and frequency representations of the AN data. Then, we transformed these representations to vector-shaped matrices. The transformed data were input into unsupervised ML methods applied in a step-by-step workflow: elbow test, *k*-means, and silhouette. These methods estimated the best number of classes characterizing the AN recordings and consequently assigned the proper class membership (cluster) to each data sample. The unsupervised clustering indicated 40 clusters. This number was derived from the elbow-test analysis and was further reduced to 10 classes by visual inspection and silhouette QC. These 10 different classes represented different seismic-event types with a detection accuracy of ~90%. We achieved the automatic detection without *a priori* knowledge of AN wavefield and detection thresholds. We demonstrated that large volumes of continuous AN data can be easily classified, labeled, and turned into an event-oriented database, which can further be used for various purposes (imaging, monitoring, and so on).

Data and Resources

The Kylylahti-array data were acquired as a part of the ERA-MIN Cost-effective Geophysical Imaging Techniques for supporting Ongoing MINeral exploration in Europe (COGITO-MIN) project. The data are embargoed until 31 December 2020 after which time they will be available from the induced seismicity-European plate observing system Anthropogenic Hazards (EPOS TCS AH) online platform (https://tcs.ah-epos.eu). The multicluster feature selection (MCFS) method (Cai *et al.*, 2010) source codes are freely available at http://www.cad.zju.edu.cn/home/dengcai/Data/MCFS.html. All websites were last accessed August 2019.

Acknowledgments

The COGITO-MIN project was funded under the ERA-MIN initiative and received funding in Poland from the National Center for Research and Development (NCBR). The authors thank numerous people from the University of Helsinki, the Geological Survey of Finland, Institute of Geophysics Polish Academy of Sciences (IG PAS), Boliden, and NovaSeis for arranging, deploying, and maintaining the Kylylahti array. All unsupervised-clustering calculations and analyses were implemented in Python using the freely available Scikit-learn package (Pedregosa *et al.*, 2011) and the MATLAB Statistics and Machine Learning Toolbox. We thank Editors-in-Chief Zhigang Peng and Allison Bent, reviewers Michael Behm, Youzuo Lin, one anonymous reviewer, and SRL Editorial Staff for important suggestions and comments.

References

Almendros, J., J. J. M. Ibáñez, G. Alguacil, and E. Del Pezzo (1999). Array analysis using circular-wave-front geometry: An application to locate the nearby seismo-volcanic source, *Geophys. J. Int.* **136,** 159–170, doi: 10.1046/j.1365-246X.1999.00699.x.

Araya-Polo, M., J. Jennings, A. Adler, and T. Dahlke (2018). Deep-learning tomography, *The Leading Edge* **37,** no. 1, 58–66.

Bensen, G. D., M. H. Ritzwoller, M. P. Barmin, A. L. Levshin, F. Lin, M. P. Moschetti, N. M. Shapiro, and Y. Yang (2007). Processing seismic ambient noise data to obtain reliable broad-band surface wave dispersion measurements, *Geophys. J. Int.* **169,** 1239–1260, doi: 10.1111/j.1365-246X.2007.03374.x.

Ben-Zion, Y., F. L. Vernon, Y. Ozakin, D. Zigone, Z. E. Ross, H. Meng, M. White, J. Reyes, D. Hollis, and M. Barklage (2015). Basic data features and results from a spatially dense seismic array on the San Jacinto fault zone, *Geophys. J. Int.* **202,** 370–380, doi: 10.1093/gji/ggv142.

Bergen, K. J., T. Chen, and Z. Li (2019). Preface to the focus section on machine learning in seismology, *Seismol. Res. Lett.* **90,** doi: 10.1785/0220190018.

Bhatti, S. M., M. S. Khan, J. Wuth, F. Huenupan, M. Curilem, L. Franco, and N. B. Yoma (2016). Automatic detection of volcano-seismic events by modeling state and event duration in hidden Markov models, *J. Volcanol. Geoth. Res.* **324,** 134–143, doi: 10.1016/j.jvolgeores.2016.05.015.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*, Springer Verlag, New York, New York.

Bormann, P. (1998). Conversion and comparability of data presentations on seismic background noise, *J. Seismol.* **2,** 37–45, doi: 10.1023/A:1009780205669.

Böse, M., F. Wenzel, and M. Erdik (2008). PreSEIS: A neural network-based approach to earthquake early warning for finite faults, *Bull. Seismol. Soc. Am.* **98,** no. 1, 366–382.

Brenguier, F., P. Kowalski, N. Ackerley, N. Nakata, P. Boué, M. Campillo, E. Larose, S. Rambaud, C. Pequegnat, T. Lecocq, *et al.* (2016). Toward 4D noise-based seismic probing of volcanoes: Perspectives from a large-*N* experiment on Piton de la Fournaise Volcano, *Seismol. Res. Lett.* **87,** 15–25, doi: 10.1785/0220150173.

Cai, D., Ch. Zhang, and X. He (2010). Unsupervised feature selection for multi-cluster data, *KDD '10 Proceedings of the 16th ACM SIGKDD International Conf. on Knowledge Discovery and Data Mining*, 333–342, doi: 10.1145/1835804.1835848.

Chamarczuk, M., M. Malinowski, Y. Nishitsuji, J. Thorbecke, E. Koivisto, S. Heinonen, S. Juurela, M. Mężyk, and D. Draganov (2019). Automatic 3D illumination-diagnosis method for large-*N* arrays: Robust data scanner and machine learning feature provider, *Geophysics* **84,** doi: 10.1190/geo2018-0504.1.

Chen, Y. (2018). Automatic microseismic event picking via unsupervised machine learning, *Geophys. J. Int.* **212,** no. 1, 88–102.

Cheraghi, S., J. A. Craven, and G. Bellefleur (2015). Feasibility of virtual source reflection seismology using interferometry for mineral exploration: A test study in the Lalor Lake volcanogenic massive sulphide mining area, Manitoba, Canada, *Geophys. Prospect.* **63,** 833–848, doi: 10.1111/1365-2478.12244.

Cuéllar, A., G. Suárez, and J. M. Espinosa-Aranda (2018). A fast earthquake early warning algorithm based on the first 3 s of the *P*-wave coda, *Bull. Seismol. Soc. Am.* **108,** no. 4, 2068–2079.

Dales, P., P. Audet, G. Olivier, and J. P. Mercier (2017). Interferometric methods for spatio-temporal seismic monitoring in underground mines, *Geophys. J. Int.* **210,** no. 2, 731–742, doi: 10.1093/gji/ggx189.

Davies, D., E. Kelly, and J. Filson (1971). VESPA process for analysis of seismic signals, *Nat. Phys. Sci.* **232,** no. 27, 8–13, doi: 10.1038/physci232008a0.

Dean, T., J. C. Dupuis, and R. Hassan (2015). The coherency of ambient seismic noise recorded during land surveys and the resulting implications for the effectiveness of geophone arrays, *Geophysics* **80,** 1–10, doi: 10.1190/GEO2014-0280.1.

Del Pezzo, E., A. Esposito, F. Giudicepietro, M. Marinaro, M. Martini, and S. Scarpetta (2003). Discrimination of earthquakes and underwater explosions using neural networks, *Bull. Seismol. Soc. Am.* **93,** 215–223.

Diebold, J. B., and P. L. Stoffa (1981). The traveltime equation, tau-p mapping, and inversion of common midpoint data, *Geophysics* **46,** no. 3, 238–254, doi: 10.1190/1.1441196.

Ding, C., and X. He (2004). *K*-means clustering via principal component analysis, *ICML '04 Proceedings of the Twenty-First International Conf. on Machine Learning*, 29, doi: 10.1145/1015330.1015408.

Dowla, F. U., S. R. Taylor, and R. W. Anderson (1990). Seismic discrimination with artificial neural networks: Preliminary results with regional spectral data, *Bull. Seismol. Soc. Am.* **80,** 1346–1373.

Draganov, D., X. Campman, J. W. Thorbecke, A. Verdel, and K. Wapenaar (2009). Reflection images from ambient seismic noise, *Geophysics* **74,** no. 5, A63–A67, doi: 10.1190/1.3193529.

Draganov, D., X. Campman, J. W. Thorbecke, A. Verdel, and K. Wapenaar (2013). Seismic exploration-scale velocities and structure from ambient seismic noise (>1 Hz), *J. Geophys. Res.* **118,** 4345–4360, doi: 10.1002/jgrb.50339.

Draganov, D., K. Wapenaar, W. Mulder, J. Singer, and A. Verdel (2007). Retrieval of reflections from seismic background-noise measurements, *Geophys. Res. Lett.* **34,** L04305, doi: 10.1029/2006GL028735.

Dysart, P. S., and J. J. Pulli (1990). Regional seismic event classification at the NORESS array: Seismological measurements and the use of trained neural networks, *Bull. Seismol. Soc. Am.* **80,** 1910–1933.

Florek, K., J. Łukaszewicz, J. Perkal, and S. Zubrycki (1951). Sur la liason et la division des points d'un ensemble fini, *Colloquium Mathematicae* **2,** 282–285, doi: 10.4064/cm-2-3-4-282-285 (in Polish).

Gerstoft, P., and T. Tanimoto (2007). A year of microseisms in southern California, *Geophys. Res. Lett.* **34,** L20304, doi: 10.1029/2007GL031091.

Hand, E. (2014). A boom in boomless seismology, *Science* **345,** no. 6198, 720–721, doi: 10.1126/science.345.6198.720.

Hansen, S. M., and B. Schmandt (2015). Automated detection and location of microseismicity at Mount St. Helens with a large-*N* geophone array, *Geophys. Res. Lett.* **42,** 7390–7397, doi: 10.1002/2015GL064848.

Harmon, N., P. Gerstoft, C. A. Rychert, G. A. Abers, M. Salas de la Cruz, and K. M. Fischer (2008). Phase velocities from seismic noise using beamforming and cross correlation in Costa Rica and Nicaragua, *Geophys. Res. Lett.* **35,** L19303, doi: 10.1029/2008GL035387.

Hennenfent, G., and F. J. Herrmann (2006). Seismic denoising with nonuniformly sampled curvelets, *Comput. Sci. Eng.* **8,** no. 3, 16–25, doi: 10.1109/MCSE.2006.49.

Jain, A. K., and R. C. Dubes (1988). *Algorithms for Clustering Data*, Prentice-Hall, Inc., Upper Saddle River, New Jersey.

Johnson, D. H., and D. E. Dudgeon (1993). *Array Signal Processing: Concepts and Techniques*, Prentice Hall, Englewood Cliffs, New Jersey.

Karplus, M., and B. Schmandt (2018). Preface to the focus section on geophone array seismology, *Seismol. Res. Lett.* **89,** no. 5, 1597–1600, doi: 10.1785/0220180212.

Kaufman, L., and P. Rousseeuw (1990). Finding groups in data: An introduction to cluster analysis, *J. Roy. Stat. Soc. C* **40,** doi: 10.2307/2347530.

Kong, Q., R. M. Allen, L. Schreier, and Y.-W. Kwon (2016). MyShake: A smartphone seismic network for earthquake early warning and beyond, *Sci. Adv.* **2,** e1501055, doi: 10.1126/sciadv.1501055.

Kong, Q., A. Inbal, R. M. Allen, Q. Lv, and A. Puder (2019). Machine learning aspects of the MyShake global smartphone seismic network, *Seismol. Res. Lett.* **90,** doi: 10.1785/0220180309.

Kong, Q., D. T. Trugman, Z. E. Ross, M. J. Bianco, B. J. Meade, and P. Gerstoft (2018). Machine learning in seismology: Turning data into insights, *Seismol. Res. Lett.* **90,** no. 1, 3–14, doi: 10.1785/0220180259.

Lance, G. N., and W. T. Williams (1967). A general theory of classificatory sorting strategies: 1. Hierarchical systems, *Comput. J.* **9,** no. 4, 373–380, doi: 10.1093/comjnl/9.4.373.

Lehujeur, M., J. Vergne, J. Schmittbuhl, D. Zigone, A. Le Chenadec, and EstOF Team (2018). Reservoir imaging using ambient noise correlation from a dense seismic network, *J. Geophys. Res.* **123,** 6671–6686, doi: 10.1029/2018JB015440.

Li, Z., M.-A. Meier, E. Hauksson, Z. Zhan, and J. Andrews (2018). Machine learning seismic wave discrimination: Application to earthquake early warning, *Geophys. Res. Lett.* **45,** 4773–4779, doi: 10.1029/2018GL077870.

Li, Z., Z. Peng, D. Hollis, L. Zhu, and J. McClellan (2018). High resolution seismic event detection using local similarity for large-*N* arrays, *Sci. Rep.* **8,** Article Number 1646, doi: 10.1038/s41598-018-19728-w.

Lin, F.-C. L., R. Clayton, and D. Hollis (2013). High-resolution 3D shallow crustal structure in Long Beach, California: Application of ambient noise tomography on a dense seismic array, *Geophysics* **78,** no. 4, Q45–Q56, doi: 10.1190/geo2012-0453.1.

Lloyd, S. P. (1982). Least squares quantization in PCM, *IEEE Trans. Inform. Theor.* **28,** 129–136, doi: 10.1109/TIT.1982.1056489.

MacQueen (1967). Some methods for classification and analysis of multivariate observations, *Proc. of 5-th Berkeley Symposium on Mathematical Statistics and Probability*, Statistical Laboratory of the University of California, Berkeley, 27 December 1965–7 January 1966, Vol. 1, 281–297.

Malfante, M., M. Dalla Mura, J. Metaxian, J. I. Mars, O. Macedo, and A. Inza (2018). Machine learning for volcano-seismic signals: Challenges and perspectives, *IEEE Signal Process. Mag.* **35,** no. 2, 20–30, doi: 10.1109/MSP.2017.2779166.

Meier, M.-A., T. Heaton, and J. Clinton (2015). The Gutenberg algorithm: Evolutionary Bayesian magnitude estimates for earthquake early warning with a filter bank, *Bull. Seismol. Soc. Am.* **105,** no. 5, 2774–2786.

Minson, S. E., M. A. Meier, A. S. Baltay, T. C. Hanks, and E. S. Cochran (2018). The limits of earthquake early warning:

Timeliness of ground motion estimates, *Sci. Adv.* **4,** no. 3, doi: 10.1126/sciadv.aaq0504.

Nakano, M., D. Sugiyama, T. Hori, T. Kuwatani, and S. Tsuboi (2019). Discrimination of seismic signals from earthquakes and tectonic tremor by applying a convolutional neural network to running spectral images, *Seismol. Res. Lett.* **90,** doi: 10.1785/0220180279.

Nakata, N., P. Boué, F. Brenguier, P. Roux, V. Ferrazzini, and M. Campillo (2016). Body and surface wave reconstruction from seismic noise correlations between arrays at Piton de la Fournaise volcano, *Geophys. Res. Lett.* **43,** 1047–1054, doi: 10.1002/2015GL066997.

Neidell, N. S., and M. T. Taner (1971). Semblance and other coherency measures for multichannel data, *Geophysics* **36,** 482–497, doi: 10.1190/1.1440186.

Nishitsuji, Y., C. A. Rowe, K. Wapenaar, and D. Draganov (2016). Reflection imaging of the Moon's interior using deep-moonquake seismic interferometry, *J. Geophys. Res.* **121,** 695–713, doi: 10.1002/2015JE004975.

Niu, F., and K. Yamaoka (2018). Preface to the focus section on non-explosive source monitoring and imaging, *Seismol. Res. Lett.* **89,** no. 3, 972–973, doi: 10.1785/0220180092.

Ochoa, L. H., L. F. Niño, and C. A. Vargas (2018). Fast magnitude determination using a single seismological station record implementing machine learning techniques, *Geodes. Geodynam.* **9,** no. 1, 34–41.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, and V. Dubourg (2011). Scikit-learn: Machine learning in Python, *J. Mach. Learn. Res.* **12,** 2825–2830.

Perol, T., M. Gharbi, and M. Denolle (2018). Convolutional neural network for earthquake detection and location, *Sci. Adv.* **4,** no. 2, e1700578, doi: 10.1126/sciadv.1700578.

Peterson, J. (1993). Observations and modeling of seismic background noise, *U.S. Geol. Surv. Open-File Rept. 93-322*, 95 pp.

Quiros, D. A., L. D. Brown, K. K. Davenport, J. A. Hole, A. Cabolova, C. Chen, L. Han, M. C. Chapman, and W. D. Mooney (2017). Reflection imaging with earthquake sources and dense arrays, *J. Geophys. Res.* **122,** no. 4, 3076–3098, doi: 10.1002/2016JB013677.

Quiros, D. A., A. Cabolova, L. D. Brown, C. Chen, J. E. Ebel, and J. Starr (2015). Aftershock Imaging with Dense Arrays (AIDA) following the M_w 4.0 Waterboro earthquake of 16 October 2012 Maine, U.S.A., *Seismol. Res. Lett.* **86,** no. 3, 1032–1039, doi: 10.1785/0220140169.

Roots, E., A. Calvert, and J. Craven (2017). Interferometric seismic imaging around the active Lalor mine in the Flin Flon greenstone belt, Canada, *Tectonophysics* **718,** doi: 10.1016/j.tecto.2017.04.024.

Rost, S., and C. Thomas (2002). Array seismology: Methods and applications, *Rev. Geophys.* **40,** no. 3, 2-1–2-27, doi: 10.1029/2000RG000100.

Rost, S., and C. Thomas (2009). Improving seismic resolution through array processing techniques, *Surv. Geophys.* **30,** nos. 4/5, 271–299, doi: 10.1007/s10712-009-9070-6.

Rouet-Leduc, B., C. Hulbert, N. Lubbers, K. Barros, C. J. Humphreys, and P. A. Johnson (2017). Machine learning predicts laboratory earthquakes, *Geophys. Res. Lett.* **44,** doi: 10.1002/2017GL074677.

Rousseeuw, P. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster results, *J. Comput. Appl Math.* **20,** 53–65, doi: 10.1016/0377-0427(87)90125-7.

Ruigrok, E., X. Campman, D. Draganov, and K. Wapenaar (2010). High-resolution lithospheric imaging with seismic interferometry, *Geophys. J. Int.* **183,** 339–357, doi: 10.1111/j.1365-246X.2010.04724.x.

Schuster, G. T. (2009). *Seismic Interferometry*, Cambridge University Press, Cambridge, England.

Schuster, G. T., J. Yu, J. Sheng, and J. Rickett (2004). Interferometric/daylight seismic imaging, *Geophys. J. Int.* **157,** 838–852, doi: 10.1111/j.1365-246X.2004.02251.x.

Singh, A. K., Y. Avantika, and R. Ajay (2013). K-means with three different distance metrics, *Int. J. Comput. Appl.* **67,** 13–17, doi: 10.5120/11430-6785.

Singh, K. S, K. Verma, and A. S. Thoke (2015). Investigations on impact of feature normalization techniques on classifier's performance in breast tumor classification, *Int. J. Comput. Appl.* **116,** no. 19, doi: 10.5120/20443-2793.

Sokal, R. R., and C. D. Michener (1958). A statistical methods for evaluating relationships, *Univ. Kansas Sci. Bull.* **38,** 1409–1448.

Steinbach, M., L. Ertöz, and V. Kumar (2004). The challenges of clustering high dimensional data, in *New Directions in Statistical Physics*, L. T. Wille (Editor), Springer, Berlin, Heidelberg, 273–309, doi: 10.1007/978-3-662-08968-2_16.

Thorndike, R. L. (1953). Who belongs in the family? *Psychometrika* **18,** 267–276, doi: 10.1007/BF02289263.

Wang, J., and T.-L. Teng (1995). Artificial neural network-based seismic detector, *Bull. Seismol. Soc. Am.* **85,** 308–319.

Wapenaar, K., and J. Fokkema (2006). Green's function representations for seismic interferometry, *Geophysics* **71,** no. 4, SI33–SI46, doi: 10.1190/1.2213955.

Wapenaar, K., D. Draganov, and J. O. A. Robertsson (2008). *Seismic Interferometry: History and Present Status*, Geophysics Reprint Series, Vol. 26, SEG, Tulsa, Oklahoma, doi: 10.1190/1.9781560801924.

Ward, J. H. (1963). Hierarchical grouping to optimize an objective function, *J. Am. Stat. Assoc.* **58,** no. 301, 236–244, doi: 10.2307/2282967.

Wilmore, P. L. (1979). *Manual of Seismological Observatory Practice*, World Data Center for Solid Earth Geophysics, Washington, D.C., 5–7.

Wiszniowski, J., B. M. Plesiewicz, and J. Trojanowski (2014). Application of real time recurrent neural network for detection of small natural earthquakes in Poland, *Acta Geophys.* **62,** 469–485, doi: 10.2478/s11600-013-0140-2.

Woollam, J., A. Rietbrock, A. Bueno, and S. De Angelis (2019) Convolutional neural network for seismic phase classification, performance demonstration over a local seismic network, *Seismol. Res. Lett.* **90,** doi: 10.1785/0220180312.

Wu, Y., Y. Lin, Z. Zhou, D. C. Bolton, J. Liu, and P. Johnson (2019). DeepDetect: A cascaded region-based densely connected network for seismic event detection, *IEEE Trans. Geosci. Remote Sens.* **57,** no. 1, 62–75, doi: 10.1109/TGRS.2018.2852302.

Yuan, B., Y. J. Tan, M. K. Mudunuru, O. E. Marcillo, A. A. Delorey, P. M. Roberts, J. D. Webster, C. N. L. Gammans, S. Karra, G. D. Guthrie, *et al.* (2019). Using machine learning to discern eruption in noisy environments: A case study using CO_2-driven cold-water Geyser in Chimayó, New Mexico, *Seismol. Res. Lett.* **90,** doi: 10.1785/0220180306.

Zhou, Z., Y. Lin, Z. Zhang, Y. Wu, and P. Johnson (2019). Earthquake detection in 1D time-series data with feature selection and dictionary learning, *Seismol. Res. Lett.* **90,** no. 2A, 563–572, doi: 10.1785/0220180315.

Zhu, W., and G. C. Beroza (2018). PhaseNet: A deep-neural-network based seismic arrival time picking method, available at http://arxiv.org/abs/1803.03211v1 (last accessed January 2019).

Appendix A

Array-processing and machine-learning hybrid workflow

Input-feature selection. Below, we describe the array-processing techniques used in our study for the extraction of input features.

Beamforming is a very effective tool for evaluating velocity and direction using seismic arrays (e.g., Johnson and Dudgeon, 1993; Rost and Thomas, 2002; Gerstoft and Tanimoto, 2007; Harmon *et al.*, 2008; Draganov *et al.*, 2013; Brenguier *et al.*, 2016). In the simplest delay-sum form, it is based on summing the recordings from N nodes by applying appropriate time delays τ_i:

$$\boldsymbol{B}(t) = \frac{1}{N}\sum_{i=1}^{N} s_i(t+\tau_i), \tag{A1}$$

in which s_i is the sample of the seismogram from station i recorded at time t. The time delay τ_j is the arrival-time difference of the wavefront between the seismometer at site i and the seismometer at a reference site. Therefore, the beamforming output calculated for each noise panel (time sample) contains output being the function of the velocity and dominant direction of the ambient-noise (AN) wavefield recorded at a given time instance.

Power spectral density (PSD) captures both the energy and frequency of AN. It is used in global AN reference models (Peterson, 1993). PSD is a suitable spectral representation of seismic noise (Bormann, 1998), and it is commonly used as AN data quality indicator (e.g., Lehujeur *et al.*, 2018) and as a part of a selection filter in extracting signals useful for AN seismic interferometry (Draganov *et al.*, 2013; Roots *et al.*, 2017). Based on the PSD values, it is easy to discern intense and quiet periods of noise activity (Peterson, 1993); thus, PSD is a suitable tool for detection of AN events. PSD can be defined as the Fourier transform of the autocorrelation function $p(\tau) = \langle f(t)f(t+\tau)\rangle$, that is:

$$\boldsymbol{P}(\omega) = \int_{-\infty}^{\infty} p(\tau)\exp(i\omega\tau)d\tau. \tag{A2}$$

To account for the location of the AN sources, we select the InterLoc method (Dales *et al.*, 2017). This approach was already verified in the setting dominated by the mine noise (Dales *et al.*, 2017), and is calculated in the cross-correlation domain, therefore enhancing any potential AN event captured in a given noise panel. The location L is calculated using the following formula:

$$\boldsymbol{L}(\vec{q}) = \sum_{i=1}^{N}\sum_{j=i+1}^{N} C_{ij}(\tau_i - \tau_j), \tag{A3}$$

in which C_{ij} is the set of time-domain cross correlations between every unique pair of receivers i and j, in which N is the number of receivers and τ_n is the travel time from location vector $\vec{q}$ to receiver n.

Data preprocessing. In this section, the input raw continuous recordings are passed through conventional AN processing workflow using normalization techniques commonly applied in the seismological community (see e.g., Bensen *et al.*, 2007).

The input for the data preprocessing step in this study comprises 20 hr of continuous AN data, which was recorded during a single day. We select a day with a variety of different activities, including road traffic, mine operations (underground blasting and surface drillings), and controlled-source shooting (Vibroseis and explosives). We split the recordings into 10-second-long noise panels with a 0.1 s overlap, which results in 7980 noise panels for the entire day. Then, we apply band-pass filtering (5–10–110–120 Hz). Assuming we have no *a priori* knowledge about the frequency of seismic events in the Kylylahti area, we set up the corner frequencies only to avoid spatial and frequency aliasing. After filtering, we apply trace energy normalization (by dividing each trace in each noise panel by its energy) and finally taper the 5% on each end of every trace using a Gaussian-shape window.

Input-feature extraction. The input-feature extraction step for location features for the Kylylahti data consisted of (1) computing the location for every noise panel using the InterLoc method, (2) computing the PSD for every noise panel, and (3) computing the beamforming for every noise panel using the delay and sum beamforming algorithm.

Data augmentation. In the data augmentation step, we process the input features represented by array-processing outputs obtained in the Input-Feature Extraction step. As opposed to the processing applied in the Data Preprocessing step, here we apply normalization techniques commonly used in the machine-learning community (this normalization is referred to as feature scaling). In Figure A2, we show the example output of all processing techniques used in this study, calculated for seven consecutive noise panels.

In this study, the input to the data augmentation step consists of input features obtained from the beamforming, InterLoc, and PSD. To lower the computational effort, without losing information about coherent events, the output of these array-processing techniques is subjected to the following procedure: (1) smoothing, (2) downsampling, and (3) vectorizing to obtain a vector with 81 input parameters. The result of applying this procedure on exemplary seven noise panels is shown in Figure A2. Then, with feature scaling (Singh *et al.*, 2015), we scale the data per feature so that every input parameter of one feature over all

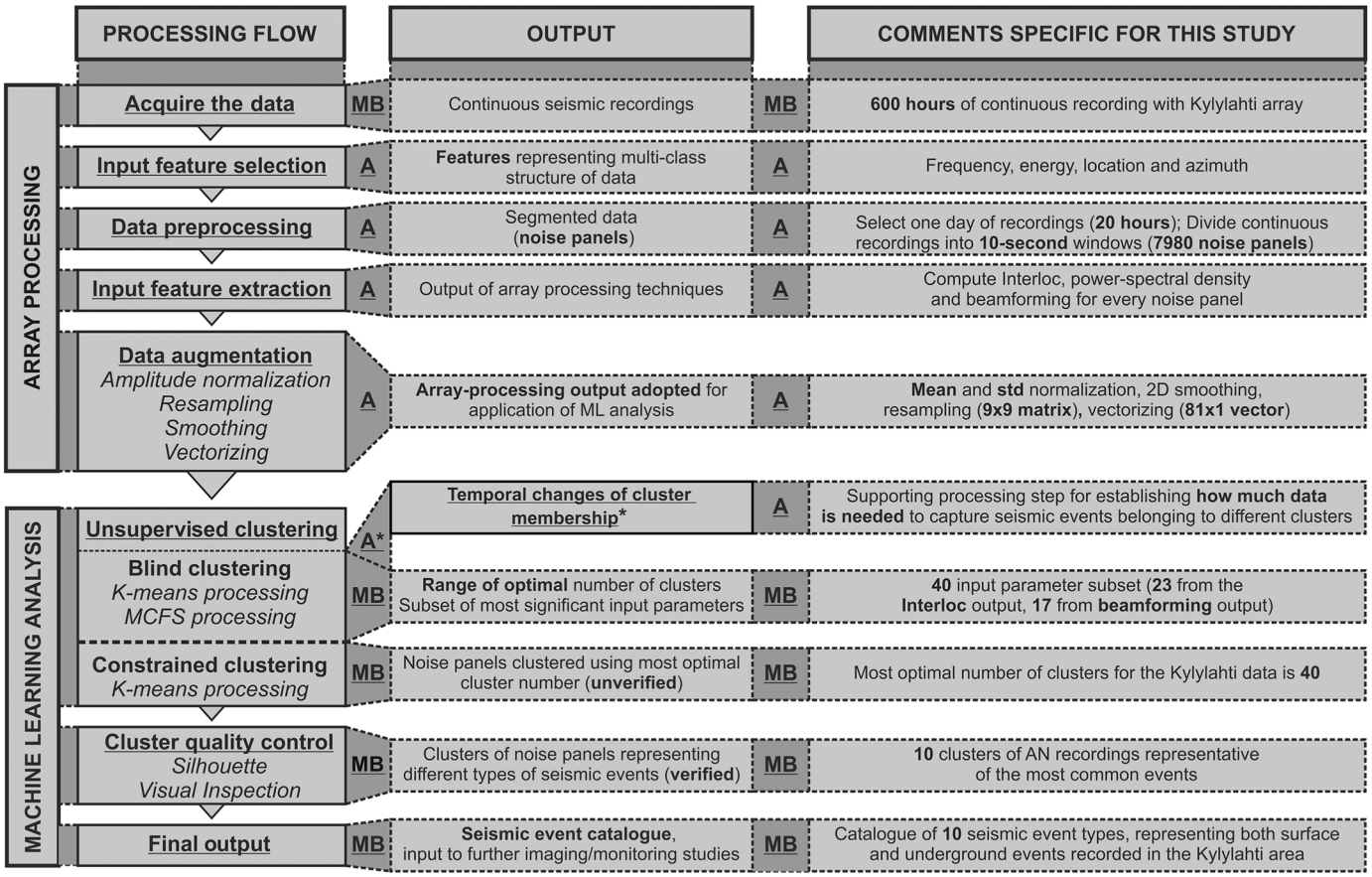

Figure A1. Extended version of the hybrid workflow sketch showed in Figure 2. The parts of processing flow described in the main body are denoted with "MB" and those in the Appendix A. The asterisk denotes additional processing step for estimating the influence of recording time on clustering results. The bold values and terms are provided to indicate the processing details and values selected especially for this study. AN, ambient noise; ML, machine learning. The color version of this figure is available only in the electronic edition.

panels (noise samples) is normalized having a mean value of 0. Figure 3 illustrates the procedure of the data augmentation in the location feature domain. For the more detailed parameters, see comments in the Figure A1. For the location-parameter case, each input parameter represents the value from a single grid node. The previous procedure is repeated for every noise panel and we thus obtained a vector of input parameters for each recorded noise panel. A similar procedure is applied to obtain input-parameter vectors for beamforming and PSD (see Fig. A2). The data were processed such that the vectors with input parameters for each feature domain had an equal number of elements as indicated in Figure A2.

Figure A3 shows the input parameters obtained for the three input-feature domains for the entire day of recording, represented by 7980 columns (noise panels; this totals to 20 hr). Each of the 81 rows in each of the three panels in Figure A3 represents a single input parameter for all noise panels. Temporal changes of the values presented in Figure A3 provide the first hints of the clustering preponderances of each feature domain. Evaluating features from the three different domains enables detection of different types of AN variations (varying length of horizontal stripes of similar color). The beamforming input-feature domain (Fig. A3b) seems to change in the slowest manner. The plot for the location input feature (Fig. A3a) indicates relatively faster changes, although the pattern of continuity remains visible. The location and the beamforming domain seem to exhibit hourly changes, which indicate periodicity. On the other hand, the PSD-feature domain (Fig. A3c) exhibits a spiky character. In general, the beamforming and location domains, due to their smooth variability and the presence of repeatable patterns, appear to be more sensitive to possible different source types. The high randomness of the PSD feature indicates that it is very prone to random local noise that affects only up to several neighboring receivers and is not related to coherent events on the array scale. Location and beamforming, due to their inherent spatial summation, suppress the random local-noise fluctuations to some extent.

Appendix B

Unsupervised clustering techniques

***K*-means.** Clustering (e.g., Singh *et al.*, 2013) is a division of data samples into groups of similar characteristics. This

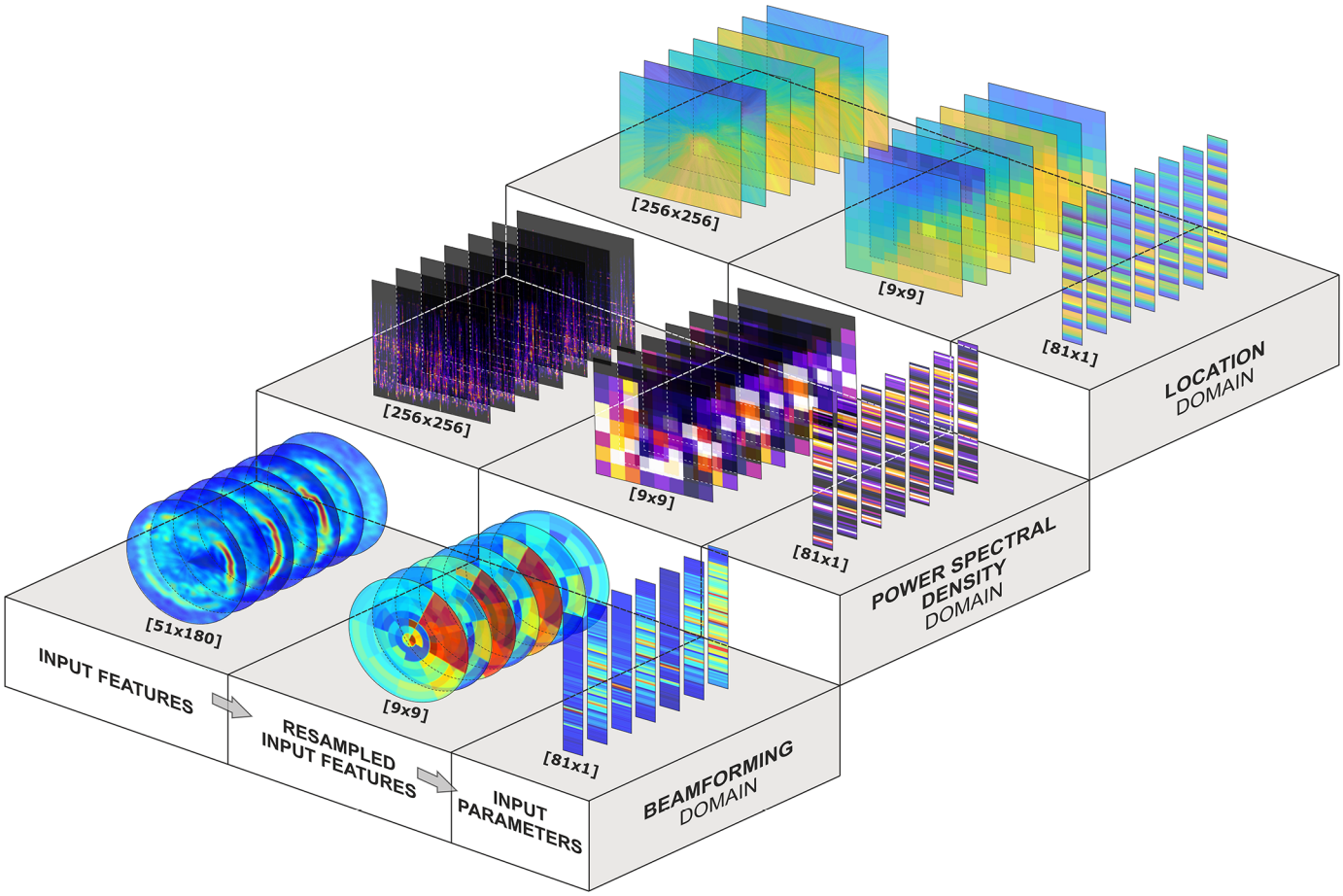

Figure A2. Data augmentation scheme for all array-processing techniques analyzed in this study: location (top row), power spectral density (PSD) (middle row), and beamforming (bottom row). The color version of this figure is available only in the electronic edition.

selection is often formulated as minimization of an objective function (Ding and He, 2004). One of the most intuitive and popular clustering algorithms is k-means clustering (Lloyd, 1982). The basic idea behind k-means clustering is to define k clusters (groups) $S = \{S1, S2, S3, \ldots, S_k\}$ such that a given dataset is classified through a predefined number of clusters k. Mathematical expression of the core of k-means can be given as minimization of an objective function defined as

$$\operatorname{argmin} \sum_{j=1}^{k} \sum_{i=1}^{n} \left|\left| x_i^{(j)} - m_j \right|\right|^2, \tag{B1}$$

in which $||x_i^{(j)} - m_j||^2$ is a distance measure between a given data point $x_i^{(j)}$ and the cluster center m_j, equal to the mean of points in the given cluster S_i. Thus, the cluster center (centroid) is the indicator of the distance of the n data points from their respective cluster centers. The practical measure of cluster compactness is the within-cluster sum of squares (WSS), which is expressed as total variance of clustering:

$$\sum_{i=1}^{k} |S_i| \mathrm{Var} S_i. \tag{B2}$$

Generally, smaller values of WSS mean better clustering. The minimization of the total variance in k-means is preformed iteratively as follows:

1. choose k-points $m_1^{(1)}, \ldots, m_k^{(1)}$ as initial cluster centers (centroids) in the space of the clustered objects;
2. assign each observation x_p to the group S^t with the closest cluster center:

$$S_i^{(t)} = \{x_p : ||x_p - m_i^{(t)}||^2 \le ||x_p - m_j^{(t)}||^2 \forall j, 1 \le j \le k\}, \tag{B3}$$

   in which t denotes iteration number, $||x_p - m_i^{(t)}||^2$ is the Euclidean distance from the observation x_p to the cluster center m_i, and $\forall j$ stands for every j;
3. after assigning all data points, recompute the centroids for each cluster

$$m_i^{(t+1)} = \frac{1}{|S_i^{(t)}|} \sum_{x_j \in S_i^{(t)}} x_j; \tag{B4}$$

4. repeat steps (2) and (3) until convergence is achieved, that is, until the cluster centers stop changing.

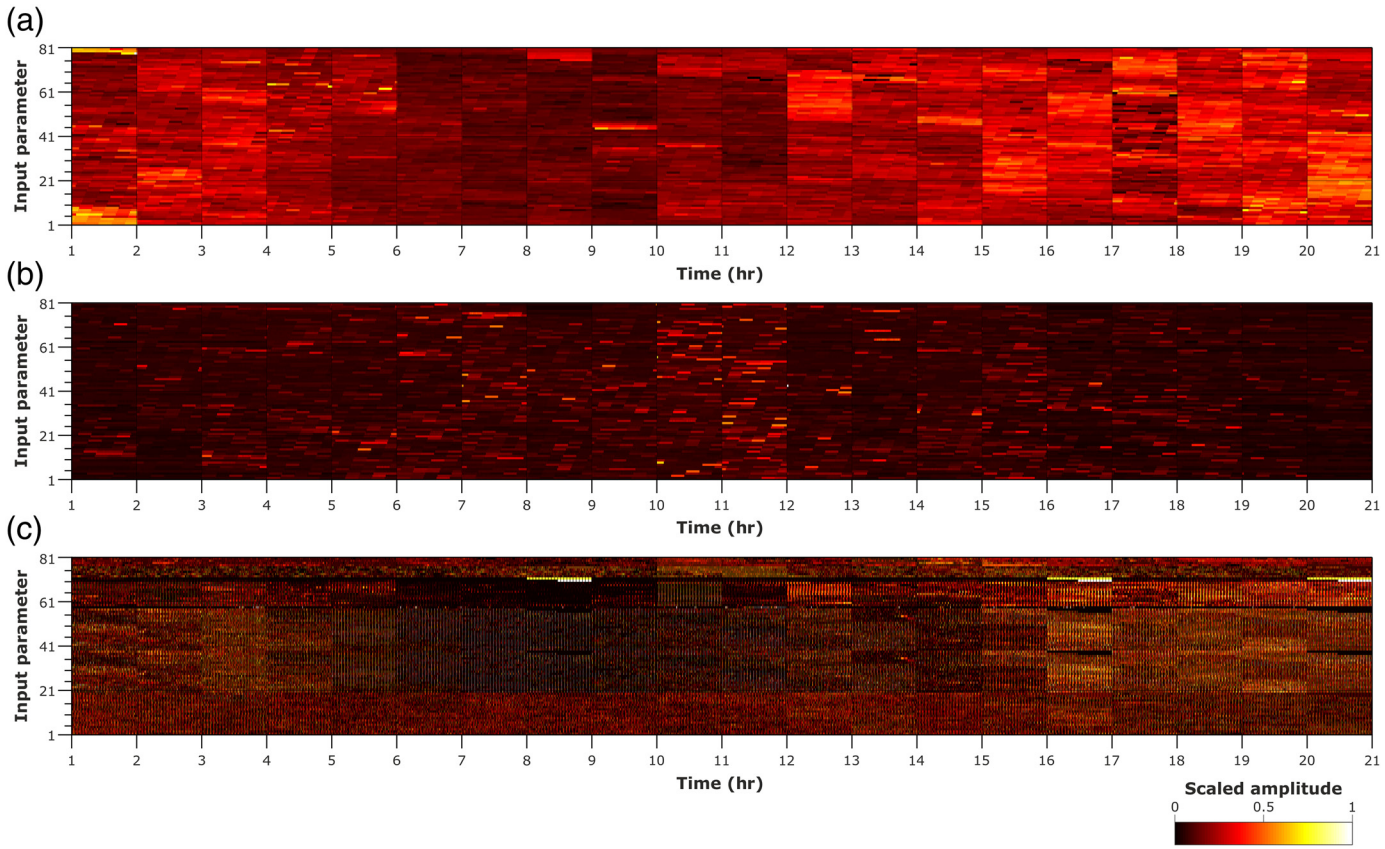

Figure A3. Input parameters (after data augmentation step) calculated for 20 hr of ambient-noise recordings. (a) Location, (b) beamforming, and (c) PSD. Each of the 81 rows in every panel represents a single input parameter for the entire day of recording, and each column shows the entire input-parameter vector for one noise panel. The color version of this figure is available only in the electronic edition.

Similar to other clustering approaches, the convergence of the k-means result does not assure finding the global-objective function minimum (MacQueen, 1967). This is because k-means is a data exploratory method. In other words, the algorithm result is affected by the *a priori* chosen number of clusters. To reduce such bias, several methods have been developed. In this study, we adapt the elbow test (Thorndike, 1953) together with the silhouette analysis (Rousseeuw, 1987).

Elbow test and Silhouette analysis. The elbow test (Thorndike, 1953) measures the total WSS as a function of the number of clusters, which is chosen such that adding another cluster does not improve the total WSS. We run the elbow test in an automatic way and measure the percentage of variance explained by the currently tested number of clusters. Out of all elbow-test evaluations, we select the one with the highest percentage of data-variance explained. In addition to testing k-numbers, we evaluated the elbow test on different sizes of input-parameter subsets.

We use the elbow test as the main tool in our blind unsupervised approach, that is, it allows scanning the range of potential numbers of clusters instead of assuming one (possibly wrong) k-value. The results obtained for adjacent cluster numbers could be similar. Therefore, the elbow test indicates the range of optimal numbers of clusters. To further select the best k-value, we use the silhouette value (Rousseeuw, 1987).

Silhouette analysis, on the other hand, is performed on the already clustered data. This method graphically provides how individual clusters are separated each other and how dense samples per cluster are plotted. To explain the logic behind, let us first assume that each data point x_i belongs to the certain cluster S_i indicated by k-means procedure. Then, we introduce the average distance between x_i and the remaining data points in the same cluster as

$$a(x_i) = \frac{1}{|S_i| - 1} \sum_{j \in C_i, i \neq j} d(x_i, x_j), \tag{B5}$$

in which $d(x_i, x_j)$ is the distance between data points x_i and x_j within the cluster S_i. Thus, $a(x_i)$ measures the dissimilarity of x_i to all other points in cluster S_i. Now we define the minimum average distance of data x_i to all data points in clusters other than S_i:

$$b(x_i) = \min_{k \neq i} \frac{1}{|C_k|} \sum_{j \in C_k} d(x_i, x_j). \quad \text{(B6)}$$

The value $b(x_i)$ gives the measure of dissimilarity of all clusters to which x_i does not belong. The cluster with the smallest $b(x_i)$ value is called the neighboring cluster of data point x_i. The neighboring cluster would be the second-best choice for accommodating x_i, that is, the data point x_i would be placed there whenever cluster S_i would be discarded.

Finally, the silhouette value is obtained by combing the measures $a(x_i)$ and $b(x_i)$ as follows:

$$s(i) = \begin{cases} 1 - \frac{a(x_i)}{b(x_i)} & \text{if } a(x_i) < b(x_i), \\ 0 & \text{if } a(x_i) = b(x_i), \\ \frac{b(x_i)}{a(x_i)} - 1 & \text{if } a(x_i) > b(x_i). \end{cases} \quad \text{(B7)}$$

The value $s(i)$ can be written in single formula as:

$$s(x_i) = \frac{b(x_i) - a(x_i)}{\max\{(a(x_i), b(x_i)\}}. \quad \text{(B8)}$$

The value of $s(x_i)$ measures how well object x_i has been classified. When $s(x_i)$ is close to 1, the dissimilarity of $a(x_i)$ is much smaller than the dissimilarity $b(x_i)$, that is, the data point x_i has been assigned to the proper cluster. Conversely, when $s(x_i)$ approaches −1, $a(x_i)$ is much larger than $b(x_i)$, so x_i is located closer to the neighboring cluster, that is, the object was misclassified. After computing $s(x_i)$ for each data sample, the silhouette graphical display can be constructed (see e.g., Fig. 6a). In the cluster QC step of our workflow, we use the presence of negative silhouette values as a key indicator of clusters with probable low detection rate of seismic events.

To assess the clustering quality, we calculate not only visual inspection of silhouette plots but also the average silhouette value of individual clusters. In this approach, the average silhouette of observations for different k-values can be compared. The optimal number of clusters k we eventually use is the one that maximizes the average silhouette over a range of possible k-values (Rousseeuw, 1987).

Appendix C

Temporal changes of cluster membership

To investigate the change of clustering behavior with time and facilitate the selection of input-feature domains for further study, we computed the average cluster membership for each of the three evaluated input-feature domains (see Fig. C1). To avoid averaging over different membership ranges, the averaging was done only for the results obtained for the minimum number of input parameters of 25 and the maximum of 81. This plot does not provide the correct cluster memberships (due to averaging over different memberships); rather, it indicates the general behavior of event clustering with time and facilitates the comparison of input-feature selection on the obtained clustering. The cluster memberships for location and beamforming input features (Fig. C1a and C1b, respectively) tend to change in a somewhat smooth way, which indicates that specific types of events appear in specific periods. The power spectral density (PSD) input parameters provide sparse and irregular membership (already expected from Fig. A3c) and the smallest range of cluster memberships (25–40). Such clustering indicates that even if the PSD input-feature domain might allow distinguishing between different clusters, the relatively small distance between them creates the risk of producing artificial clusters that do not represent the multiclass structure of AN from Kylylahti (i.e., do not explain the variance of our data).

Recordings acquired in an area dominated by human-induced AN such as a mine or road traffic would likely be rich in seismic events of similar type that appear at time instances covering a time span of a few noise panels. Therefore, collating observations from Figures 4, A3, and C1, we decide to further use only the location and beamforming domains in this study.

MCFS processing

The multicluster feature selection (MCFS) method is a technique for the unsupervised selection of input-parameter subset. In this method, the combination of input parameters which would maintain the high number of clusters provided minimum input data are selected (for details, see Cai *et al.*, 2010). In the MCFS processing step, we run MCFS on the range of k-values indicated by the initial blind clustering results shown in Figure 4. We test different sizes n of those subsets and iterate through them with n ranging from 20 to 59 (i.e., we run MCFS 40 times). For each test, we retrieve a subset of input parameters with the highest MCFS score.

Then, we rank each input parameter by the number of times it was included in the best subset. Figure C2a shows the number of MCFS hits for each input parameter. The plot is divided into two sections by the dashed line separating the input parameters obtained from the location and beamforming domains. We can see that in general there are three main groups in terms of input-parameter significance: (1) features of very low significance (values of ∼0–1), (2) moderate-significance features (1–30), and (3) high-significance features (∼40). The split of the most significant parameters between beamforming and location is relatively balanced. In terms of the most influential input parameters, 22 come from location and 16 from beamforming (significance parameters with values of 40 were selected as the best subset in every iteration).

Each input parameter represents a single grid node either in the location or beamforming output. This means that the input parameters as the specific characteristics describe the noise panels (location, slowness, and angle of incidence). Figure C2b shows the most significant parameters located on the resampled grid for the location (left panel) and beamforming (right panel). In addition, we use a 2D linear interpolation to project the most significant input parameters on feature maps shown in Figure 6.

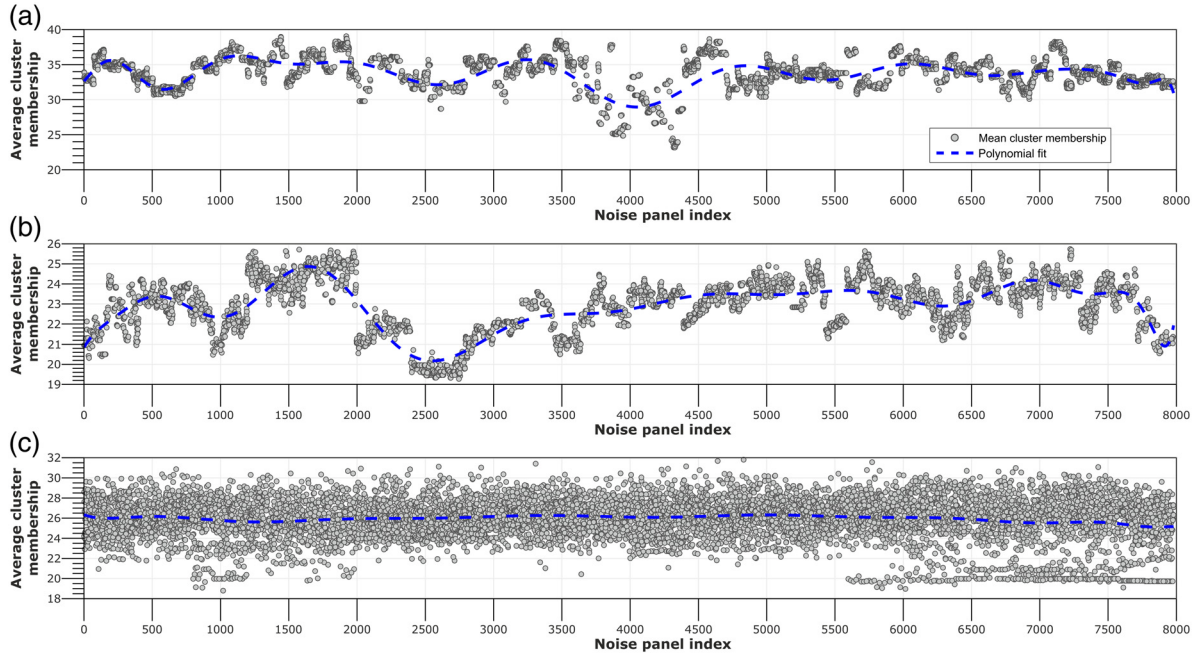

Figure C1. Average cluster membership for every noise panel recorded for one day obtained from (a) location, (b) beamforming, and (c) power spectral density feature domains. Averaging was performed over *k*-results obtained for input-parameter numbers between 25 and 81. Gray dots represent the cluster membership for a given data sample and dashed blue line is the polynomial fit visualizing the time behavior of cluster variability. The color version of this figure is available only in the electronic edition.

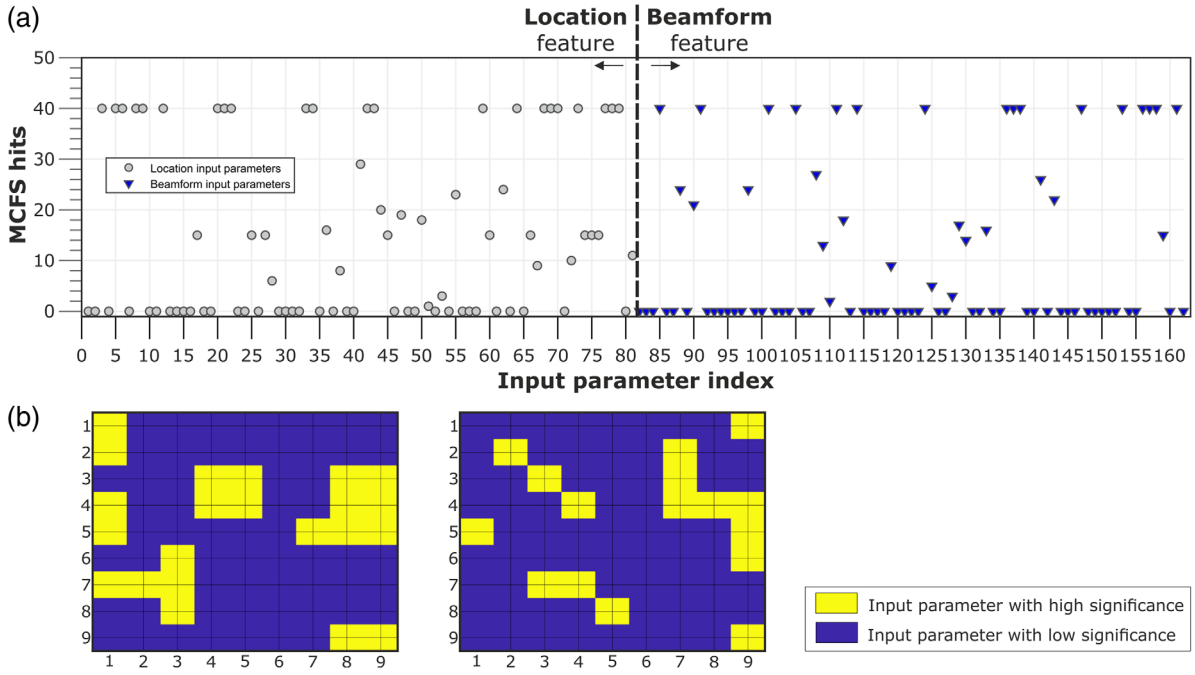

Figure C2. Multicluster feature selection (MCFS) results for the location and beamforming feature domains. (a) The number of selections for the best input-parameter subset for each parameter, (b) the most significant input parameters projected on a resampled output of location (left panel) and beamforming (right panel). The color version of this figure is available only in the electronic edition.

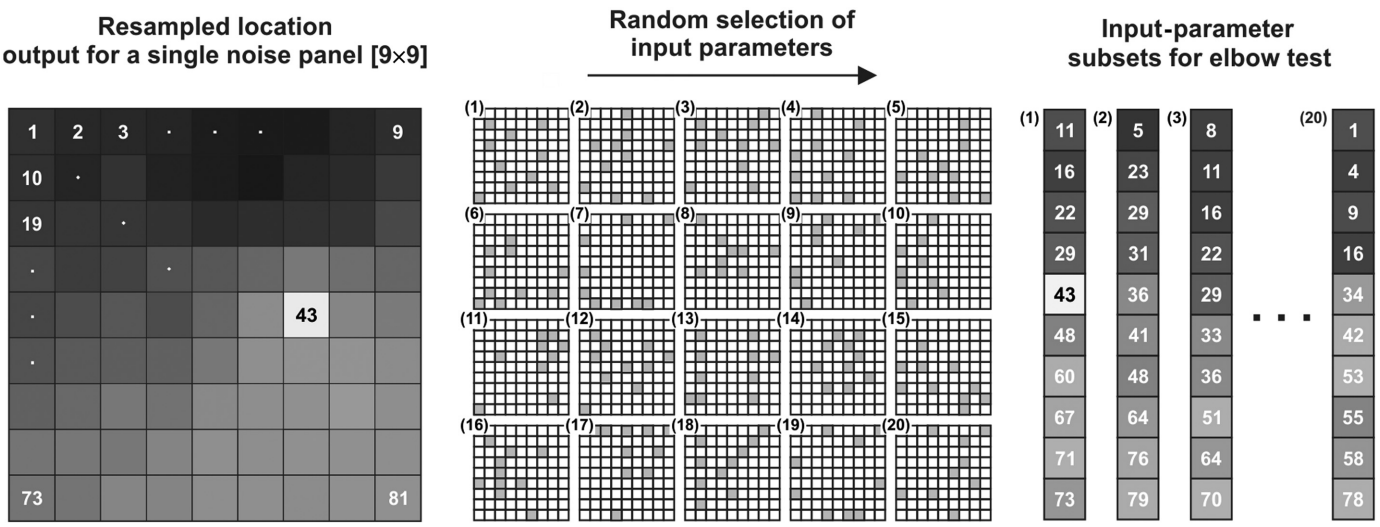

Figure D1. Procedure of generating random input-parameter subsets for the case of 10 input parameters. The average cluster membership evaluated for this case is denoted with the transparent gray rectangle in Figure 4.

Appendix D

Random input-parameter subsets for elbow-test analysis

Here, we describe the processing step which assures avoiding of sampling the same input parameters in every iteration of *k*-means in the blind clustering step. To this end, we randomize the experiment: for each tested number of input parameters, we run elbow tests on 20 different randomly selected subsets of input parameters (see the whiskers plot in Fig. 4); the average values obtained from these 20 tests are denoted with gray circles in Figure 4. The procedure of selecting subsets with random input parameters is shown in Figure D1.

Appendix E

Workflow, other clustering methods, and computational comparison

Input-feature selection, data preprocessing, and input-feature extraction. As our methodology is aimed for events recorded by arrays (of any size), the minimum criterion requires that selected input-feature domain (in the input-feature selection step) accounts for travel time and coherency of the events. Therefore, it is logical to use any signal transformations after which we preserve or enhance this characteristics of data. Such information can be also retrieved using single trace operations by simultaneous analysis of data recorded by several sensors (such as power spectral density in this study). Apart from domains selected in this work, one could consider other domains, for example, tau-*p* (Diebold and Stoffa, 1981), velocity spectral analysis (Davies *et al.*, 1971), frequency–wavenumber, curvlets (Hennenfent and Herrmann, 2006), and skewness (or kurtosis). These domains, not necessarily but might provide better representation of events. However, comparisons of different domains or finding the best combination is out of our scope. In this study, we used beamforming as one of input features into the workflow. However, we find that beamforming has useful aspect when even its performance is not superior because of plane-wave propagation assumption and the Nyquist wavenumber imposed by the used array geometry.

For extraction of single input feature (for the input-feature extraction step), usually more than one array-processing technique can be used (e.g., Rost and Thomas, 2009). Thus, the selection of array-processing technique is more ambivalent than selection of input feature. This means that the choice of specific technique basically depends on two factors as follows. In the first place, it depends on the selected input-feature domain. Meaning is that if we choose to replace the frequency input-feature domain with the coherency of events, then for example the array processing measuring semblance can be used (Neidell and Taner, 1971). In the second place, the choice of specific technique depends on the array parameters and type of AN in the recording area. It means that all processing techniques we chose to extract the input features for this study might be replaced. For instance, instead of InterLoc, the recently developed local similarity check (Li, Peng, *et al.*, 2018) developed for large-number (large-*N*) arrays could be used. This method allows to maximize the number of detected events and thus more detailed event catalog.

Data preprocessing step involves applying a sequence of processing techniques belonging to conventional routines applied in seismological studies (see e.g., Bensen *et al.*, 2007). Their application aims to accentuate any seismic events present in the recordings. Thanks to this, we increase probability of detecting more events, and consequently obtaining a more comprehensive catalog of seismic events (high number of

clusters populated with seismic events). Depending on the type of background noise in the recording area, the most suitable processing for enhancing the coherent events may vary (e.g., high-pass filtering for rejecting surface-wave content, or low-pass filtering to suppress AN caused by urban activities), and should be adjusted accordingly into our processing flow.

Other clustering methods. Clustering methods are mainly categorized either of nonhierarchical (or called partitional optimization) method (e.g., *k*-means) or hierarchical method. In general, nonhierarchical method is known to perform better than hierarchical method especially when we deal with large amount of data (Kaufman and Rousseeuw, 1990). In addition to *k*-means, there are other nonhierarchical methods such as *k*-medians method (Jain and Dubes, 1988). This method uses median (defined as value in the center of all sorted data points) to compute the cluster centers. Median value as opposed to mean is less influenced by outliers, and thus *k*-medians algorithm results in clustering less affected by values not representing any class.

As for the hierarchical clustering, a number of methods exist such as the Ward (1963) method, group average method (Sokal and Michener, 1958), single linkage method (Florek *et al.*, 1951), and complete linkage method (Lance and Williams, 1967). Because the hierarchical clustering is generally considered to be not suitable for large dataset (Kaufman and Rousseeuw, 1990), we do not introduce their details here.

The choice of an appropriate method is extremely difficult in practice to determine *a priori* because one consequently need to evaluate all approaches with different test (score) based on different metric (e.g., euclidean, cityblock, minkowski, canbella, cosine, and so on). Our primary goal is a hybrid automation of seismic processing rather than which clustering method gives best outcomes.

Computational efficiency comparison. To express it quantitatively, we us the big *O* notation used in computer science as an algorithmic efficiency measure and allows to express the upper bound of maximum number of operations specific for the given type of algorithm. In this study, n denotes number of data points. For the simplest case of linear complexity of algorithms it can be written as $O(n)$, which means that runtime of algorithm grows proportionally to n. For the unsupervised clustering algorithms mentioned earlier, the runtime algorithm complexities are *k*-means $O(n)$,; mean-shift clustering $O(n^2)$,; and hierarchical clustering $O(n^3)$. It means that the runtime of *k*-means algorithm exhibit linear proportionality to number of data points, for the mean-shift—is proportional to the squared number of points, and cubically for hierarchical clustering (assuming standard implementations for all of them).

Manuscript received 15 March 2019
Published online 13 November 2019