PICOSS: Python Interface for the Classification of Seismic Signals

A. Bueno[a], L. Zuccarello[b,c], A. Díaz-Moreno[d], J. Woolam[d], M. Titos[b], L. García[a], I. Álvarez[a], J. Prudencio[b], S. De Angelis[d]

[a]*Department of Signal Theory, Telematic and Communications, University of Granada, Spain.*
[b]*Department of Theoretical Physics and Cosmos, University of Granada, Spain.*
[c]*Istituto Nazionale di Geofisica e Vulcanologia, Sezione di Pisa, Italy*
[d]*Department of Earth, Ocean and Ecological Sciences, University of Liverpool, UK*

Abstract

Fueled by machine learning, past decade has seen significant research efforts dedicated to advance volcano-seismic data processing. However, to meet operational requirements, these algorithms are validated on direct comparisons of their predictions with annotated references. Thus, the development of a data-driven software is required to refine annotations in a manageable way. This paper presents PICOSS (Python Interface for the Classification of Seismic Signals), a modular data-curator platform for volcano-seismic data analysis: detection, segmentation and classification. With exportability and standardization at its core, users can select automatic or manual workflows to annotate seismic data from the suite of expanded tools, including deep neural networks and spectral analysis. Implemented modules follow an intuitive design to comprise, on a small-scale software, the essential data-labelling tasks required for large-scale volcano-seismic studies.

Keywords: Volcanoes, Software, Classification, Segmentation, Detection

---

1. Introduction

Seismic networks are the backbone of volcano monitoring programs worldwide. Rapid technological advances over the past two decades have made the installation of geophysical networks with multiple sensors increasingly more affordable; thus, large amounts of data are routinely generated and archived at volcano and earthquake observatories. Earthquakes at volcanoes may occur at rates as high as hundreds of events per hour during periods of unrest [1]; their waveforms must be extracted from the continuous seismic records and classified. Volcano-seismic signals are traditionally classified based on either waveform appearance (e.g., the number and type of seismic phases visible) and frequency domain features (e.g., the frequency band over which most energy is predominantly delivered). Two of the most widely adopted classification schemes for volcanic earthquakes are those of [2] and [3], with different terminology but similar classification criteria. Today's most commonly adopted volcano-seismic classifications include high-frequency (HF), hybrids or mixed frequency (MF), low-frequency (LF) earthquakes and tremor (T), in addition to a number of other signals generated by surface processes (landslides, lahars, pyroclastic flows), and explosions (E). Over the years, different solutions for analysing continuous seismic data have been implemented ([4], [5], [6], [7], [8], [9], [10]). These softwares range from open source solutions, frequently offered at the cost of complex setup procedures, to simpler data analysis frameworks often based on commercial high-level programming languages. Most recently, the Python language has gained popularity in the field of seismology and new toolboxes have been developed; the most popular example is Obspy [11], a flexible and modular environment to access seismic data in different formats, and to perform both basic and high-level data analysis tasks on multi-channel seismic data [12]. Our first-generation interface [10] is rooted on Python and Obspy routines specifically designed for manual picking of P and S wave arrival times, of essential importance on source location and tomography tasks. However, modern seismic processing pipelines requires data-centric software design guidelines to capture the needs of large-scale deployments. Thus, based on our experience with [10] and a more complete understanding of the requirements for building robust and desirable procedures to refine seismic datasets, we have built PICOSS (Python Interface for the Classification of Seismic Signals), a modular open source interface designed to support visualization, detection and characterization of volcano-seismic signals. Having a single graphical interface that can span a wide range of tasks significantly simplify the seismologists workflow. PICOSS expands our first-generation interface by implementing additional signal processing and machine learning techniques routinely used in volcano observatories and in research. Additionally, the simplicity of its GUI is an enhancement, offering opportunities for its use in higher education classroom settings.

2. PICOSS Description

PICOSS is a program conceived to detect and label volcano-seismic datasets, which offers a high-level of modularity as required for modern seismology toolboxes. All outputs are stored in easily accessible, platform-independent, standard format. This format includes segmentation times, along with the identified type of event, quality of the event, and extra seismological information used to build robust datasets. PICOSS can access seismic data from community waveform data servers (e.g., the Incorporated Research Institution for Seismology Data Management Center, IRIS DMC), and from off-line data structures in multiple formats. Therefore, PICOSS is build to enable the following work-flow loop:

1. **Manual Inspection**: PICOSS provides a fully-functional GUI interface for manual inspection of seismic events, allowing the analyst to associate custom loaded labels and organise relevant seismic information to annotate the data. Supervised earthquake analysis capabilities include time-frequency analysis, and assignment of classification labels and picking times.
2. **Detection**: PICOSS supplies algorithms, such us STA/LTA (Short-Term Average/Long-Term Average) [17] or adaptative multi-band processing algorithm (AMPA; [9]) to perform automatic detection and picking. Additional infrastructure and binding modules are also provided to ease the integration with new algorithms.
3. **Classification**: The program includes a classification module that categorizes seismic signals according to Frequency Index analysis ([21]) or a multi-volcano pre-trained neural network ([20]) specifically designed to refine datasets and decrease manual inspection time.

Our software can read labels and/or phase arrival times from existing catalogs, and associate them to waveform data. By default, PICOSS include the most common labels in volcanic-seismology. However, specific categorization schemes can be loaded via the *"Extra Info"* menu, which permits a broader range of applications (i.e, infra-sound or hydro-acoustic data). Finally, the classification module permits the imported segmented signals to be grouped based on unsupervised learning techniques, thus providing an intuitive visualization tool of data hierarchies. Specifically, active learning, i.e., the continuous and interactive querying of data following an acquisition function criteria, is incorporated as a procedure to decrease the amount of time required for data labelling and model adaptation [13]. A Command Line Interface (CLI) is provided with a set of auxiliary task for data conversion between formats and add scalability to the whole workflow.

2.1. PICOSS GUI

PICOSS can display continuous streams, currently up to 24h, along with their spectrogram and Fourier power spectrum. Figure 1 shows an example of continuous seismic data recorded on 19th October, Montserrat Volcano observatory (MVO), station $MBGA$, vertical component Z [14]. The analyst can drag-select any part of the loaded stream, visualize the spectrogram and the spectrum, and annotate the data. Above the main trace panel (*main*), PICOSS includes a toolbar from which the user can access most of the program facilities, i.e: loading traces, signal processing routines, detection and picking routines or add extra (custom) information. The default seismic labels offered by PICOSS follows common classification practice and terminology in volcano-seismology, although custom labels can be loaded via the *"Extra Info"* menu. Additionally, a quality factor, Q, can be assigned to the signal as a qualitative measure of the analyst's confidence in the classification; Q ranges from 1 (very poor) to 5 (excellent). Segmented events can be compared with data recorded on other stations/components; the options

Visualize other components and *Visualize other stations* open an auxiliary small GUI in which the user can explore other components and/or stations.

PICOSS is written in Python, and built on Obspy [11], a popular Python-based seismic data analysis package. Libraries for segmentation, classification and detection are programmed in NumPy and Scipy. The CLI provides access to these specific routines, with a set of auxiliary tasks, including the conversion of segmented results from NumPy to MATLAB, earthquake detection, picking routines and automated classification of volcanic signatures.

3. Earthquake detection and segmentation

The earthquake detection functionality allow the configuration of earthquake detection and picking based on STA/LTA, AMPA and/or more specific detectors for low frequency events, including those based on simple envelop processing [15] and wavelet decomposition [16].

Figure 2 shows results from the application of the STA/LTA triggering algorithm to a seismic swarm recorded at Mount St. Helens in 2004. Detection results can be further extended with automated segmentation with REMOS (Recursive Entropy Method of Segmentation) algorithm [18]. REMOS is an end-to-end approach which uses the detection time as the anchor point of an exploration window to determine the exact boundaries (beginning and end of an event) based on entropy measures. By taking measurements of the seismic energy, a minimum entropy criterion is used by REMOS to investigate large amounts of earthquake triggers and to discriminate and parse events into individual waveforms. REMOS configuration parameters shall be selected by the analyst to meet operational requirements specifically for each volcano.

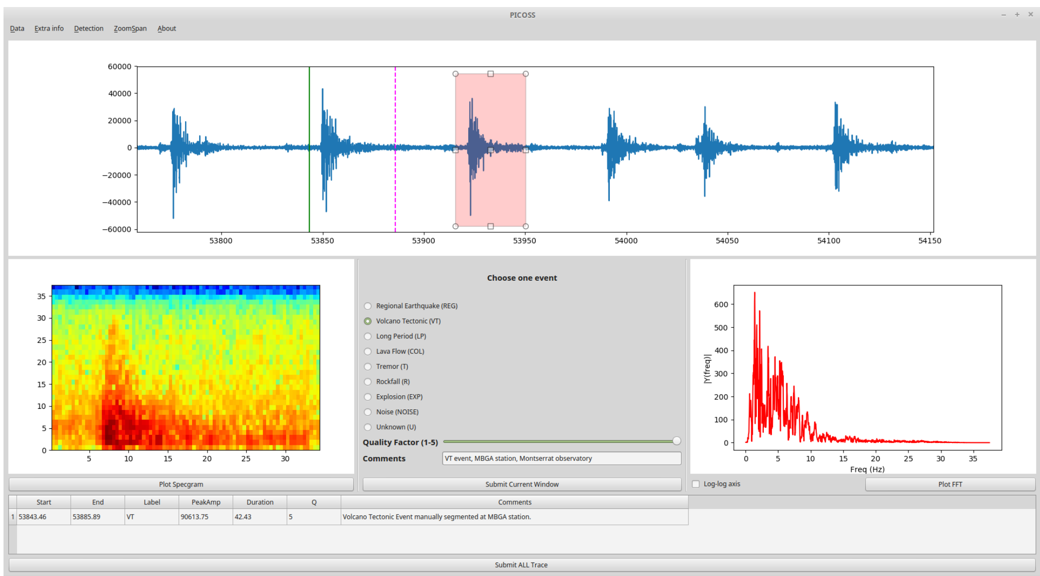

Figure 1: (a) Screenshot of PICOSS Main Interface, showing a sequence of manually segmented high-frequency events from Soufriere Hills Volcano, Montserrat, recorded the 19th October 1997.

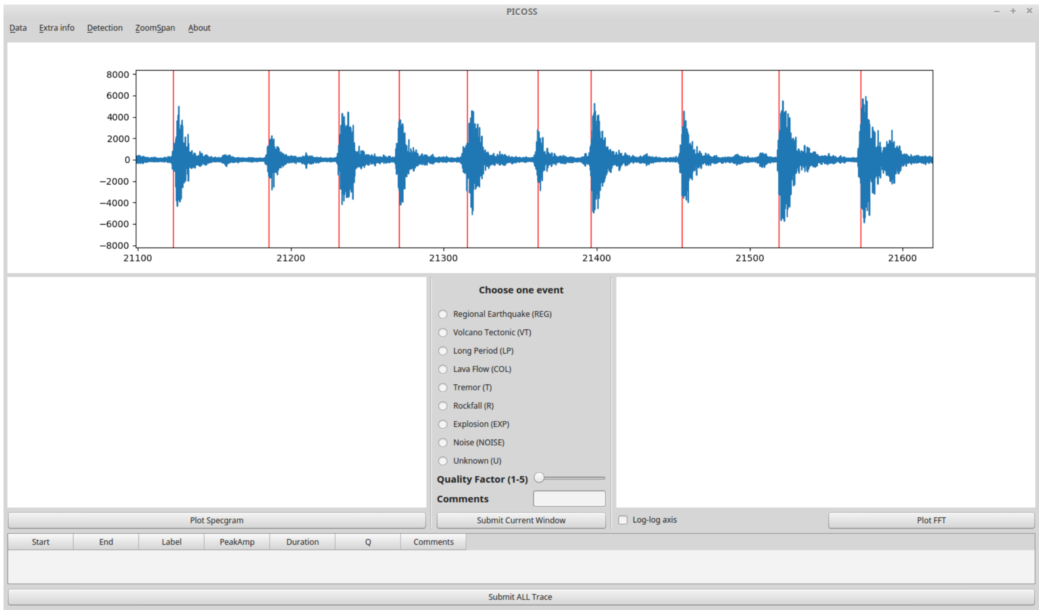

Figure 2: Example of application of STA/LTA, using the PICOSS interface, to continuous data recorded at Mt. St. Helens volcano on the 23rd November, 2004. PICOSS outputs a standard format which includes onset detection times. Alternative earthquake triggering methods (e.g, AMPA), along with automated data segmentation (i.e., extraction of earthquake waveforms with REMOS) are also included.

4. Spectral Analysis

Whilst an analyst can manually segment and classify data from the main interface, PICOSS incorporates a semi-automated classification module. This classification is based on the well-known categorization scheme proposed by [3]. PICOSS automatic classification is based on a pre-trained Bayesian Neural Network (BNN; [20]) and a frequency-index (FI) analysis [21], in which the logarithm of the ratio of spectral energies over user-specified frequency bands is computed:

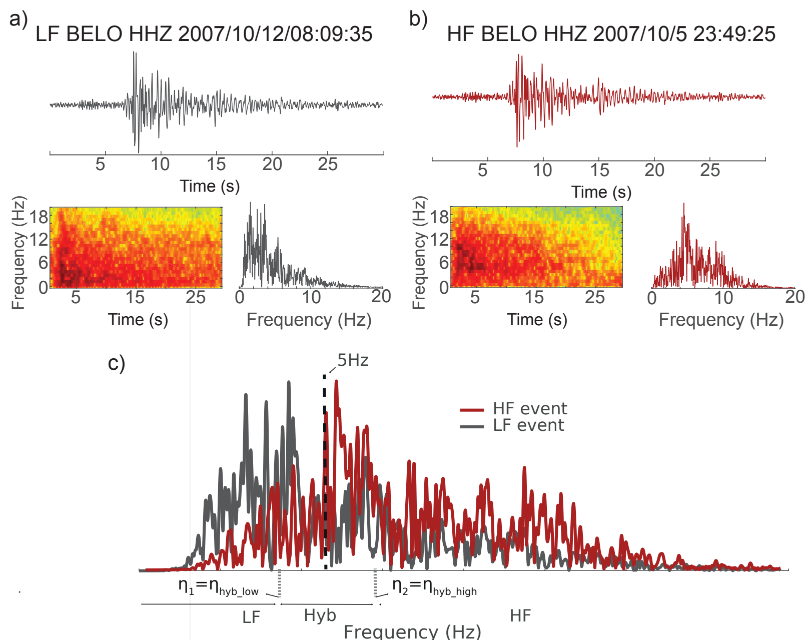

Figure 3: Figure 3.a and 3.b show seismograms, spectrograms and power spectrum for two segmented signals at BELO station. These earthquakes are classified as low-frequency (3.a) and high-frequency (3.b), respectively. Figure 3.c shows high-frequency and low-frequency spectra along with the threshold selection for frequency index calculation (see main text). Mixed frequency events are confined within the interval $[\eta_1, \eta_2]$.

$$FI = log_{10}\left(\frac{A_{high}}{A_{low}}\right) \tag{1}$$

where A_{high} and A_{low} are the spectral energies above and over user-specified frequency bands. A set of threshold values $\eta = \{\eta_1, \eta_{hybrid\,low}, \eta_{hybrid\,high}, \eta_2, \eta_3\}$ are defined as the FI thresholds that control how events are categorized according to their FI value. Events are classified as Low-Frequency (LF) if the FI is below the given threshold η_1. Similarly, if the FI its greater than threshold η_2, it is classified as High-Frequency (HF). The events with FI between η_1

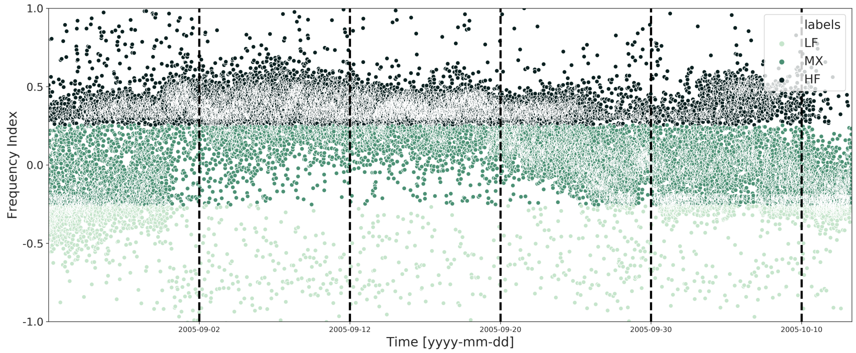

Figure 4: Temporal evolution of the frequency of detected in the 2005 St.Helens eruption, station S02.

and η_2 correspond to events with significant energy at both low and high frequencies and are categorized as hybrids. Parameters $\eta_{hybrid\,low}$ and $\eta_{hybrid\,high}$ can be configured to quantify hybrid events and overcome the limitation of a single threshold for HF and LF events (see Fig.3.c). The sufficient time t over which to consider low frequency or tremor depends of the analyst and the volcanic-environment [3]. As tremor and rockfall require different frequency characterization as those of high/low frequency events, η_3 is a threshold to control whether an event is classified as a rockfall or tremor. For each event identified in the initial continuous trace, PICOSS extracts their waveform, computes the FI as in equation 1, assigns a label, and stores the results in *.npy* format.

The implemented module includes a deep neural network trained in a multi-volcano datasets with shared knowledge representations across similar seismic events. The trained network follows a Bayesian approach, in which the deterministic neural connections are substituted by probability distributions, thus bridging the gap between Bayesian modelling and deep learning architectures [20] [13]. This probabilistic approach allows principled uncertainty quantification for transient seismic sources. The computed uncertainty is exploited under an active learning setting: using a maximum entropy acquisition strategy, those segmented events with the highest classification uncertainty are selected (queried) to the user, which ultimately decides the samples that shall be used for re-training purposes. The included neural network is trained on very characteristic events from Bezymianny and St.Helens volcanoes, under the generalization scheme proposed by [3] (see 3). This has the advantage of reducing time while allowing the model to generalize for new data samples rapidly, even under intense seismic activity. Figure 4 depicts the temporal evolution of the FI for the volcano-seismic segmented events recorded at Mount St. Helens volcano, $S02$ station, using the pre-trained neural network. Only Q>2 are selected for this figure, as those below exhibit low SNR. Note the evolution of the frequency index over time; this type of analysis provides valuable, rapid, initial assessment of if and how seismic unrest evolves over time.

5. Output formats

The ultimate objective of PICOSS is to assist users to produce high-quality seismo-volcanic datasets. PICOSS stores earthquake segmentation and classification results by serializing the *segmentation table* (manual or automatic) in a high and efficient Python format, the *pickle* format. PICOSS saves the *start*, *end* times (in seconds), along with the assigned *label*, the $PeakAmplitude$, the *duration*, quality Q values and additional comments for further post-processing. Additionally, PICOSS allows the user to further edit the *segmentation table*. Finally, *pickle* formats can be converted to NumPy arrays, MATLAB data formats (*.mat*), text CSV or numerical Pandas *dataframe* framework. The trigger times, along with the visualization, if required, are stored as NumPy arrays and can also be converted using the CLI utility functions included within *"convert data"* script. Each stream is saved according to its metadata within the seismic network, including the station, component, year, day and last segmentation time.

6. Conclusions

At present, a wealth of available seismic data acquired at volcanoes worldwide remain largely underutilized. In this paper, we present PICOSS, a Python Interface for the Classification of Seismic Signals. PICOSS is a modular open source software, with a graphical user interface designed for detection, segmentation and classification, focus on exportability and standardization of data formats. PICOSS includes functionalities that are programmed as independent modules that can be easily adapted for operational requirements at volcano observatories. The implemented modules are independent from each other, and provide a collection of tools to analyse volcano-seismic data. The user can switch between automatic or manual modes by incorporating this suite of tools to efficiently compile complete catalogues of labelled volcano-seismic events. PICOSS could also be used for educational purposes, or refine other types of seismic data. All the modules of this interface are designed to reduce data labelling fatigue while increasing data classification efficiency.

7. Acknowledgments

PICOSS has benefited greatly from helpful feedback and advice from all the members at the Seismic Laboratory at the University of Liverpool and *Instituto Andaluz de Geofísica* (Granada). The authors are indebted to Prof. Jesús M. Ibáñez and Prof. Carmen Benítez, without a doubt, the functionalities of PICOSS has been greatly expanded with their feedback. This interface is currently being used to produce the datasets used in the following projects: TEC2015-68752 and VOLCANOWAVES (Marie Sklodowska-Curie Grant Agreement no 798480). This research is funded by TEC2015-68752 (MINECO/FEDER), by NERC Grant NE/P00105X/1 and by European Unions Horizon 2020 Research and Innovation Programme Under the Marie Sklodowska-Curie Grant Agreement no 798480.

[1] Bell, A. F., and C. R. J. Kilburn (2012), Precursors to dyke-fed eruptions at basaltic volcanoes: insights from patterns of volcano-tectonic seismicity at Kilauea volcano, Hawaii, *Bulletin of Volcanology*, *74*(2), 325–339, doi:10.1007/s00445-011-0519-3.

[2] Minakami, T. (1974), Prediction of volcanic eruptions, *Developments in Solid Earth Geophysics*, *6*, pp. 313–333.

[3] McNutt, S. R., G. Thompson, J. Johnson, S. D. Angelis, and D. Fee (2015), Chapter 63 - seismic and infrasonic monitoring, in *The Encyclopedia of Volcanoes (Second Edition)*, edited by H. Sigurdsson, second edition ed., pp. 1071 – 1099, Academic Press, Amsterdam, doi: https://doi.org/10.1016/B978-0-12-385938-9.00063-8.

[4] Goldstein, P., Dodge, D., Firpo, M., and Ruppert, S., (1998), What's new in SAC2000? Enhanced processing and database access, *Seismological Research Letters*, *69* (3), pp. 202–205.

[5] Havskov, J. and Ottemoller, L., (1999), SEISAN earthquake analysis software, *Seismological Research Letters*, *70* (5), pp. 532–534.

[6] Lesage, P. (2009), Interactive Matlab software for the analysis of seismic volcanic signals, *Computers and Geosciences*, *35* (10), pp. 2137–2144.

[7] Abdelwahed, M.F. (2012), SGRAPH (SeismoGRAPHer): Seismic waveform analysis and integrated tools in seismology, *Computers and Geosciences*, *40*, pp. 153–165.

[8] Olivieri, M., and Clinton, J., (2012), An almost fair comparison between

Earthworm and SeisComp3, *Seismological Research Letters*, *83* (4), pp. 720–727.

[9] Álvarez, I., García, L., Mota, S., Cortes, G., Benítez, C and De la Torre, A., (2013), An automatic P-phase picking algorithm based on adaptive multiband processing, *IEEE Geoscience and Remote Sensing Letters*, *10* (6), pp. 1488–1492.

[10] Romero, J.E., Titos, M., Bueno, A., Álvarez, I., García, L., De la Torre, A., Benítez, C., (2016), APASVO: A free software tool for automatic P-phase picking and event detection in seismic traces, *Computer and Geosciences*, *90*, pp. 213–220.

[11] Beyreuther, M., Barsch, R., Krischer, L., Megies, T., Behr, Y., and Wassermann, J. (2010), Obspy: A python toolbox for seismology, *Seismological Research Letters*, *81* (3), pp. 530–533.

[12] Zuccarello, L., Paratore, M., La Rocca, M., Ferrari,F., Messina, A., Branca, S., Contrafatto, D., Galluzzo, D., and Rapisarda, S., (2016), Shallow velocity model in the area of Pozzo Pitarrone, Mt. Etna, from single station, array methods and borehole data, *Annals of Geophysics*, DOI: 10.4401/ag-7086

[13] Gal, Y., Islam, R., and Ghahramani, Z., (2017), Deep bayesian active learning with image data, *Proceedings of the 34th International Conference on Machine Learning*, *70*, pp. 1183–1192.

[14] Luckett, R., Baptie, B., Ottemoller, L., and Thompson, G., (2007), Seis-

mic Monitoring of the Soufriere Hills Volcano, Montserrat, *Seismological Research Letters*, *78* (2), pp. 192–200.

[15] García, L., Álvarez, I., Titos, M., Díaz-Moreno, A., Benítez, C., and de la Torre, A., (2017), Automatic Detection of Long Period Events Based on Subband-Envelope Processing, *10* (11), pp. 5134–5142

[16] Kanwaldip Singh, A., and Dowla, F., (1997). Wavelet transform methods for phase identification in three-component seismograms. Bulletin of the Seismological Society of America. *87* (6), pp. 1598–1612.

[17] Allen, R., (1982), Automatic phase pickers: their present use and future prospects, *Bulletin of the Seismological Society of America*, *72* (6B), pp. 225–242.

[18] Bueno, A., Diaz-Moreno, A., De Angelis, S., Benitez, C., and Ibañez, J.M. (2019), REMOS: Recursive Entropy Method of Segmentation, *Seismological Research Letters*, (accepted), https://doi.org/10.1785/0220180317.

[19] Roman, D. C. (2017), Automated detection and characterisation of harmonic tremor in continuous seismic data: Harmonic tremor detection, *44*.

[20] Bueno, A., Titos M., García, L., Álvarez, I., Ibañez, J.M., Benítez, C. (2018), Classification of Volcano-Seismic Signals with Bayesian Neural Networks, *2018 26th European Signal Processing Conference (EUSIPCO)*, (IEEE), pp. 2295–2299.

[21] Buurman, H., and West, M. (2010), Seismic precursors to volcanic explosions during the 2006 eruption of Augustine volcano, *1769*.