Automatic Reclassification of Volcano-Seismic Signals from Soufrière Hills Volcano, Montserrat, 1996-2008

Glenn Thompson[1], Alexis Falcin[2], Jean-Philippe Metaxian[3], and Francois Beauducel[4]

[1]University of South Florida
[2]Institut de Physique du Globe de Paris
[3]IRD, IPGP, Université de Paris
[4]Université de Paris

November 24, 2022

Abstract

Seismic activity during the eruption of Soufriere Hills volcano comprised various transient signals, which were classified visually by the Montserrat Volcano Observatory (MVO), considering waveforms recorded at several stations. For 217,290 transients detected on the MVO digital seismic network between 1996/10/21 and 2008/10/16, five main classes have been identified: rockfall (ROC: 58%), hybrid (HYB: 19%), long-period (LPE: 11%), lp-rockfall (LP-ROC: 5.8%), and volcano-tectonic (VT: 3.1%). Temporal trends in the rate and energy release of these different transients (in addition to swarms and tremor) were key to short-term forecasting of eruptive activity. However, visual classification is highly subjective and non-repeatable, and the inconsistency of the catalog is a barrier to research. In a pilot study, we automatically removed waveforms with dropouts, and manually verified transient classifications until we had approximately 100 transients of each class (total 522). We found ~21% of these transients were incorrectly classified at MVO. Our re-labelled dataset was then used as a starting point for supervised learning, using code from http://github.com/malfante/AAA. This code was used by Malfante et al. (2008) to classify 109,609 transients at Ubinas volcano with a 93.5% accuracy. They transformed each waveform into a set of 102 features: 34 features for each of three domains (time, spectral, cepstral). We added 6 frequency features of our own, including band ratios, peak frequency, median frequency, bandwidth, and frequency change. The resulting 108-point vectors of features were then used for modeling. The dataset is randomly divided 50 times into training and testing datasets, to produce a robust model. One model is produced per channel. We use the Random Forest Classifier algorithm from the scikit-learn library. For each waveform, a probability is computed for each class. Initial results are promising. Separate models for 3 channels yield accuracies of 76-80%. If the LP-ROC class is omitted (following Langer et al, 2006), accuracy rises to 82-85%. If only VT and LP classes are considered, accuracy is 96-99%. We intend to expand our labelled dataset to 1000 events, add new features, build models for each channel, and reclassify the catalog of 217,290 transients by a weighted average of probabilities.

Auto-Reclassification of Volcano-Seismic Signals from Soufrière Hills Volcano, Montserrat, 1996-2008

Glenn Thompson[1], Alexis Falcin[2], Jean-Philippe Metaxian[2], François Beauducel[2]
[1]University of South Florida, [2]Université de Paris, IPGP, CNRS

V25D-0146
Tuesday, 2021/12/14 5-7pm, Hall D-F

1. Motivation

By January 2000, the Soufrière Hills Volcano had been erupting for 3½ years. Seismicity was dominated by rockfall/pyroclastic flow (ROC) signals from the rapidly growing lava dome, which could collapse at any time. Other classes of transient event signals (Fig. 1) were volcano-tectonic (VTE), long-period (LPE) and hybrid (HYB) earthquakes, and long-period rockfall (LPE+ROC) signals. Swarms, tremor and banded tremor heralded escalations in activity. Tropical thunderstorms remobilized loose ash, causing lahars. But the Montserrat Volcano Observatory (MVO) had a seismic monitoring program that was designed around cataloging tectonic earthquakes, and examining each earthquake in isolation. There was no routine analysis of seismicity patterns, precursory signals, or hazard-related signals, and no alarm system to alert MVO scientists outside of office hours. As incoming MVO Seismologist, author GT sought to expand real-time seismic monitoring to automatically detect, locate, quantify, classify and catalog the full range of volcano-seismic signals, and create an early warning system for pyroclastic flows. A model of the expanded MVO monitoring system realized that year is shown in Fig. 2., which made possible an early-warning system for pyroclastic flows. However, the real-time event classifier was poor. External help was needed.

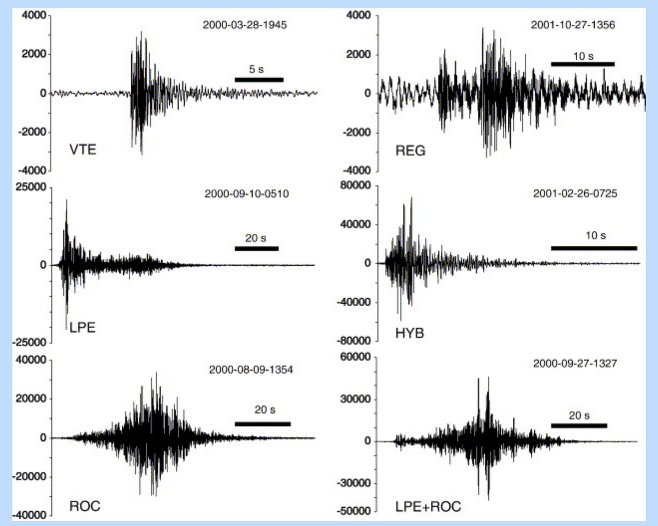

Fig. 1. Waveforms of the six classes of transients considered. Year, month, day, hour, and minute at the onset of each transient are indicated at the upper right of each trace. From Langer et al. (2006).

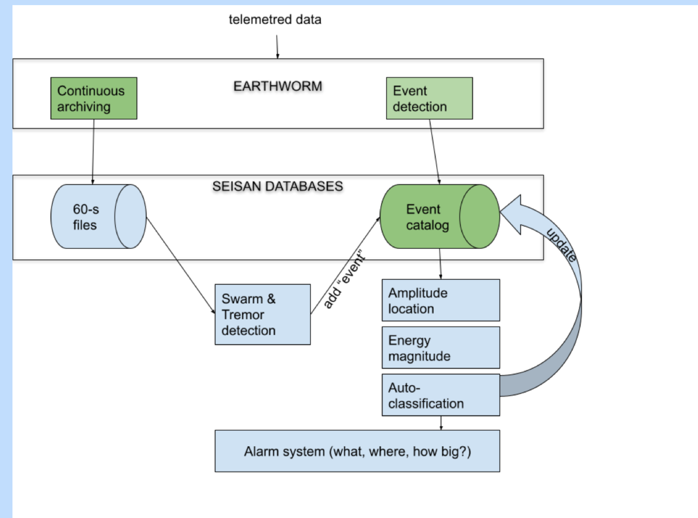

Fig. 2: The expanded MVO seismic monitoring system. The concept was to locate, quantify and classify every event in real-time, and detect swarms and tremor events too. Previously, only tectonic earthquakes were located or quantified, and only during analyst-review. Key: parts colored green pre-existed, those in blue were added in 2000.

In January 2001, GT initiated collaboration through the MULTIMO Consortium that led to two papers. Langer et al. (2003) used an Artificial Neural Network applied to 336 Z-component (event) seismograms between 1996 and 2002, achieving 70% accuracy in reproducing classifications assigned by MVO. Langer et al. (2006), expanded this method to 6000 seismograms, again achieving a 70% accuracy compared to classifications assigned by MVO. To improve on this, seismograms with a peak amplitude of less than 5000 counts were removed, reducing the dataset to 2400 seismograms. After manually reclassifying these events, and merging the LP-ROC class with ROC, the accuracy rose to 80%. However, GT left MVO in 2004 and MVO made no further efforts to implement a real-time classifier.

The Soufrière Hills Volcano last erupted in 2010, and seismicity rates have declined 100-fold since, so a real-time classifier is no longer a priority. Nevertheless, Langer et al. (2006) estimated that around 30% of events in the MVO catalog are misclassified, which limits the usefulness of the catalog for understanding the eruption, and for constructing physical models based on seismicity before, during, and after major volcanic events. For the first 15 months of the eruption - arguable the most important - no digital recordings of event classifications exist & manual classification would take months. Moreover, the richness of the MVO catalog may provide an opportunity to develop a general auto-classifier that could help for volcanic crises elsewhere, e.g. other Caribbean volcanoes.

But what really got this project off the back-burner was the publication of Malfante et al. (2018a). This is an excellent review of machine learning methods applied of volcano-seismic signals, an almost a textbook guide to the subject. Their dataset had a similar size and complexity to the MVO catalog, and their model reached 93.5% accuracy - a spectacular result! GT reached out to the authors in April 2018, one of whom (JPM) is co-author on this poster.

2. Data

The dataset used in this study come from the MVO digital seismic network. Events were detected in real-time, with waveform files saved and registered into a Seisan event database – this is the MVO catalog. Registration creates an event metadata file in "Nordic" format, corresponding to each event waveform file. Events were manually reviewed and classified by MVO's seismic team (assigning one of the classes shown in Fig. 1), within 24 hours. The catalog available to us begins on 1996/10/23 and ends on 2008/08/31.

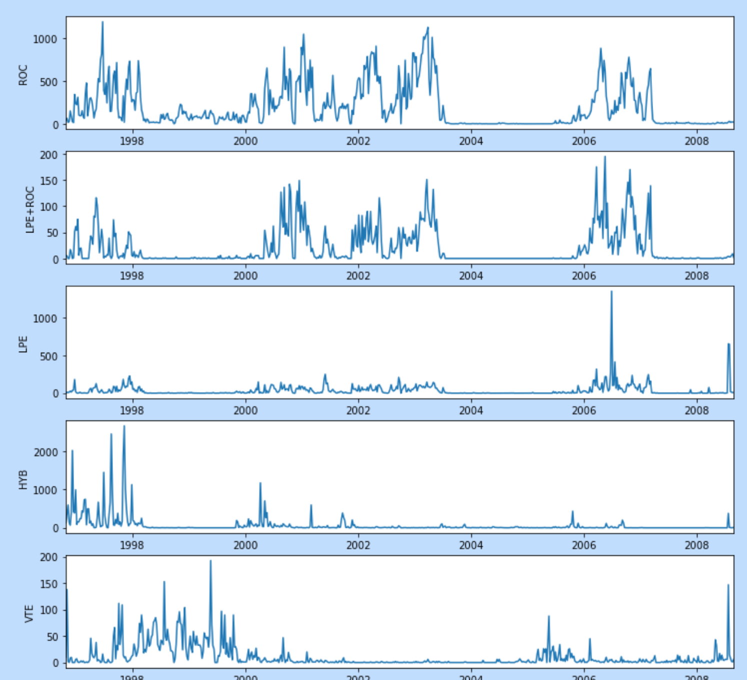

Fig. 3. The catalog after applying "Seisan2Pandas". Y-axis is number of events per week for the 5 volcano-seismic classes. Pauses in extrusion (e.g. most of 1998-9, 2004-5, 2007) are characterized by low event rates, except for VTE.

Seisan2Pandas

As a Seisan database isn't a convenient format to work with, we developed an ObsPy-based workflow we call "Seisan2Pandas". This loops through all the Nordic files and: (1) loads the correspond waveform file if it exists; (2) computes several data quality metrics and removes bad traces; (3) removes the instrument correction (we constructed a StationXML file for the entire network history); (4) computes numerous amplitude, frequency, energy, magnitude and statistical metrics for each Trace; (5) saves the QC'd/corrected traces in MiniSEED file along with a corresponding event CSV file containing all the metrics for that event. (A Trace is an ObsPy object that holds a single waveform/seismogram).

The catalog we ultimately construct is a single CSV file around 150 MB in size. Of the 231,951 events in the Seisan catalog, 213,582 were detected on the MVO digital seismic network (the rest come from a pre-existing analog network). 3,821 waveform files are missing from the Seisan database, and a further 626 failed the Seisan2Pandas quality checks, so our catalog for machine learning comprises 209,135 events (1,859,161 good Traces). Fig. 3 shows a timeseries plot of this catalog, based on original classifications assigned by MVO, with totals in Table 1.

Catalog of reclassified (or labelled) events used in this study

Since we could not rely on the a-priori classifications in the MVO catalog, we visually re-analyzed and reclassified events. This is a time consuming process, so we stopped when we had approximately 100 of each of the 5 main classes (total 522 events). Based on our re-analysis, ~21% of these events were incorrectly classified at MVO. This reclassified subset of the MVO catalog is used for supervised machine learning.

	Count
ROC	118954
LPE-ROC	12344
LPE	22451
HYB	40375
VTE	6793
REG	116
TELESEISM	10
other	8092

Table 1: The catalog after Seisan2Pandas contains 209,135 events. Number of events are listed, according to class assigned by MVO.

3. The Malfante Method

We follow the method described by Malfante et al. (2018b) to classify 109,609 events at Ubinas volcano, Peru, with a 93.5% accuracy. They considered 6 main event classes. VTE, HYB and LPE, tornillos, explosions, and tremor. This is a remarkably high accuracy with a dataset of similar size and complexity to the MVO catalog (and with 3 classes in common). Their method requires only a simple catalog containing event time, class, and path to the seismic waveform file. There are 5 steps to the method:

1) Preprocessing:

Read each signal from file. Normalize signal, so that the model will be applicable to signals of all sizes. Filter above 1 Hz, because different sensors were used at different times. Trim any events longer than 5 minutes.

2) Feature extraction:

A key point is how to select an appropriate set of features to be measured on, or computed from, each signal. They choose 9 statistical features, 9 entropy features, and 16 shape descriptors (34 total), and compute these on three different representations of each signal: (1) time domain, (2) frequency domain, and (3) cepstral domain. Thus, the feature vector for each signal has dimension 102 (34 x 3). This is a significant compression compared to using the thousands of samples present in the original signal, plus it reduces all feature vectors to the same length, regardless of signal length.

3) Learning (or Training):

A prediction model is built from a learning algorithm and a randomly selected 50% of the dataset of labelled feature vectors. The labels are the event classes assigned. Malfante et al. (2018b) used a Random Forest Classifier algorithm (100 decision trees) from the scikit-learn library, and compared it with a Support Vector Machine algorithm (results were similar). They use a maximum of 800 signals per event class for computational reasons.

4) Testing:

The 50% of the dataset of labelled feature vectors not used in learning/training, are then classified with the prediction model. This result is a confusion matrix, and a measurement of accuracy and precision. Accuracy and precision are best understood by example. If the reclassified (i.e. "labelled") dataset contains 100 rockfalls, and the model classifies 75 of those as rockfalls, the accuracy is 75/100 = 75%. However, if 50 non-rockfalls were also classified by the model as rockfalls, the precision is 75/125 = 60%. The average accuracy and precision over all classes considered are identical, but differ for each event class.

5) Cross-validation:

To test the stability of the prediction model, the learning & testing phases are repeated 50 times. This generates a generalized confusion matrix and a mean and standard deviation of the accuracy and precision.

Their code is available at https://github.com/malfante/AAA. That was our starting point.

Our modifications

We introduced some small modifications to the Malfante method:

- In the and feature extractionpreprocessing step, we added an ObsPy+Pandas based function to read the MiniSEED and CSV files for each event.
- Since our data are instrumented corrected, we chose a lower high pass corner of 0.5 Hz, to preserve LPE events.
- Preprocessing are slow, but learning, testing and cross-validation are fast. The Malfante code reloads the data files every time, and recomputes features. We modified the code to read each MiniSEED file just once, compute features for each Trace, and save the features vectors to Pickle files. This way, the slow parts only needed to be done once. This sped subsequent model runs by a factor of ~100.
- We added 6 new frequency features, including 2 band ratios, peak frequency, median frequency, and bandwidth. This increased the feature vector dimension from 102 to 108.
- We add a wrapper to run the method for different Trace IDs, and different sets of event classes.

4. Results

The network has changed over time, and stations have periodically failed. We ran the Malfante method for the 3 most common Trace IDs (net-sta-loc-chan combinations) appearing in our reclassified catalog of 522 events. For each Trace ID, we subsetted the catalog by different sets of event classes, to examine the method's ability to resolve them. These were:

- All 5 classes: LPE+ROC, HYB, LP, VTE, & ROC
- HYB, LPE, VTE, & ROC (no LPE+ROC).
- HYB, LPE, & VTE, to examine just earthquake events.
- LPE, & VTE, since HYB appear like a VTE with LPE coda

Full results are shown in Table 2, and grouped by Trace ID and classes in Tables 3 and 4, respectively.

traceID	classes	nEvents	nEventsPerClass	acc_mean	acc_std
MV.MBWH..SHZ	LP-ROC, HYB, LP, VTE, ROC	517	[110, 93, 106, 106, 102]	76.0	1.8
MV.MBWH..SHZ	HYB, LP, VTE, ROC	424	[110, 106, 106, 102]	82.1	1.8
MV.MBWH..SHZ	HYB, LP, VTE	314	[106, 106, 102]	84.0	2.2
MV.MBWH..SHZ	LP, VTE	208	[106, 102]	98.4	1.2
MV.MBWH..SHZ	LP-ROC, ROC	203	[110, 93]	78.0	3.5
MV.MBRY..SHZ	LP-ROC, HYB, LP, VTE, ROC	452	[94, 83, 89, 100, 86]	79.0	2.1
MV.MBRY..SHZ	HYB, LP, VTE, ROC	369	[94, 89, 100, 86]	85.0	2.5
MV.MBRY..SHZ	HYB, LP, VTE	275	[89, 100, 86]	87.3	2.4
MV.MBRY..SHZ	LP-ROC, ROC	177	[94, 83]	76.9	3.5
MV.MBRY..SHZ	LP, VTE	175	[89, 86]	96.4	1.6
MV.MBLG..SHZ	LP-ROC, HYB, LP, VTE, ROC	474	[102, 84, 105, 100, 83]	78.0	2.2
MV.MBLG..SHZ	HYB, LP, VTE, ROC	390	[102, 105, 100, 83]	85.3	2.4
MV.MBLG..SHZ	HYB, LP, VTE	288	[105, 100, 83]	87.5	2.5
MV.MBLG..SHZ	LP, VTE	188	[105, 83]	97.9	1.3
MV.MBLG..SHZ	LP-ROC, ROC	186	[102, 84]	79.4	3.2

Table 2: Results for all model runs. The column acc_mean is the mean accuracy (in %) from cross-validation (step 5). Results range from 76.0% to 98.4%.

traceID	acc_mean	acc_std
MV.MBLG..SHZ	85.6	2.3
MV.MBRY..SHZ	84.9	2.4
MV.MBWH..SHZ	83.7	2.1

Table 3: Average results across all sets of event classes, grouped by Trace ID (net-sta-loc-chan). The results are within 1 standard deviation (acc_std), suggesting we could treat each Trace as a separate event.

classes	acc_mean	acc_std
HYB, LP, VTE	86.3	2.4
HYB, LP, VTE, ROC	84.1	2.2
LP, VTE	97.6	1.4
LP-ROC, HYB, LP, VTE, ROC	77.7	2.0
LP-ROC, ROC	78.1	3.4

Table 4: Results grouped by classes considered, ignoring Trace ID.

	VTE	HYB	LPE	LPE+ROC	ROC	Accuracy
VTE	120	13	0	0	1	90%
HYB	17	114	14	2	6	75%
LPE	0	5	133	5	7	89%
LPE+ROC	0	0	7	98	24	76%
ROC	3	11	8	34	96	63%
Precision	86%	80%	82%	71%	72%	

Table 5: Confusion matrix for all 5 event classes (using all 1443 traces). Rows are given labels, columns are predicted labels. 90% of VTE are correctly predicted, but only 63% of ROC.

Table 4 shows an accuracy of 77.7% when all 5 classes are considered. When the LPE+ROC class is ignored, accuracy rises to 84.1% which is an improvement on the ~80% accuracy achieved by Langer et al. (2006) for the same classes, and we used only ~400 traces for each model compared to their 2400. Our model accuracy should rise as we reclassify (label) more events. Surprisingly, LPE+ROC and ROC were resolved with 78.1% accuracy. Table 5 shows these are the most easily confused signal types. When considering only the 3 earthquake types, LPE, HYB and VTE, accuracy is 86.3%. And when only LPE and VTE are considered, performance rises to 97.6%. These two results are consistent with LPE, HYB and VTE forming a continuum.

5. Discussion & Further Work

While initial results are encouraging, a 5-class model with 77.7% accuracy implies an error rate of 23.3%, which is not much better than the 30% Langer et al. (2006) estimated from a subset of the MVO catalog. Furthermore, we have ignored regional and teleseismic earthquake signals, and exotic volcanic events that are also present in the MVO catalog. We anticipate we may need to expand our reclassified (labelled) catalog (from 522) to between 1000 and 5000 events, and include examples of these other signals.

We see several opportunities to improve classification accuracy:

1. We can add new features. For example, a feature that measures the frequency change within a signal. This could help distinguish HYB (frequency increase) and LPE+ROC (frequency increase) from others (no change).
2. The MiniSEED event files have anywhere from 0-60s noise before the event signal. Testing suggests this degrades performance of the model by 5%. So if we can eliminate this variable, accuracy could improve from 77% to 82%. We could either run a detector, or compute features on the autocorrelation (as Langer et al., 2006, did), since this always peaks at t=0. This would be a 4th domain (in addition to time, spectral and cepstral).
3. We can try classifier algorithms other than RandomForest. For example, Support Vector Machine.
4. We can exploit the full network to classify events (as human analysts do). Table 3 reveals that across all experiments, the models created for 3 Trace IDs are similarly accurate. This suggests we can treat each Trace as a separate event. An alternative would be to train separate models for each Trace ID (as we did here), and return the full prediction probability vector for each class (rather than just the predicted class), and average these across all traces for that event. We will test both approaches and compare. Crucially, both approaches are impervious to station outages, which plague models based on a single station/channel.
5. We can exploit 3-component data, where available. For example, P and S waves have different relative amplitudes on vertical versus horizontal seismograms. More sophisticated analyses consider the polarization (particle motion) time series, to distinguish different wave types.

We intend to experiment with 1-4 during AGU week, and update the corresponding iPoster. .

References

Langer, H., Falsaperla, S., Thompson, G. (2003). Application of Artificial Neural Networks for the classification of the seismic of the seismic transients at Soufrière Hills volcano, Montserrat. Geophysical Research Letters, 30, 21, 2090. doi:10.1029/2003GL018082
Langer, H., Falsaperla, S., Powell, T., and Thompson, G. (2006). Automatic classification and a-posteriori analysis of seismic event identification at Soufriere hills volcano, Montserrat. Journal of Volcanology and Geothermal Research, 153, 1-10. doi:10.1016/j.jvolgeores.2005.08.012
Malfante, M., Dalla-Mura, M., Metaxian, J. P. , Mars, J. I., Macedo, O., Inza, I. A. (2018a). Machine Learning for Volcano-Seismic Signals: Challenges and Perspectives, IEEE Signal Processing Magazine, vol. 35, pp. 20-30.
Malfante, M., Dalla-Mura, M., Mars, J. I., Métaxian, J. P., Macedo, O., Inza, I. A. (2018b). Automatic Classification of Volcano Seismic Signatures, Journal of Geophysical Research - Solid Earth, 123, 12, pp. 10,645-10,658. https://doi.org/10.1029/2018JB015470.