Journal of Volcanology and Geothermal Research 316 (2016) 34–49

Contents lists available at ScienceDirect

Journal of Volcanology and Geothermal Research

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

Feature selection of seismic waveforms for long period event detection at Cotopaxi Volcano

R.A. Lara-Cueva [a,c,*], D.S. Benítez [d], E.V. Carrera [a], M. Ruiz [e], J.L. Rojo-Álvarez [b,c]

[a] Departamento de Eléctrica y Electrónica, Universidad de las Fuerzas Armadas ESPE, Av. Gral. Rumiñahui s/n, Sangolquí, Ecuador
[b] Prometeo Program, Departamento de Eléctrica y Electrónica, Universidad de las Fuerzas Armadas ESPE, Av. Gral. Rumiñahui s/n, Sangolquí, Ecuador
[c] Information and Communications Technology Department, Rey Juan Carlos University, Camino del Molino s/n, 28943 Fuenlabrada, Madrid, Spain
[d] Universidad San Francisco de Quito USFQ, Colegio de Ciencias e Ingenierías El Politécnico, Campus Cumbayá, Casilla Postal 17-1200-841 Quito, Ecuador
[e] Instituto Geofísico, Escuela Politécnica Nacional, Ladrón de Guevara E11-253, Quito, Ecuador

ARTICLE INFO

Article history:
Received 1 February 2015
22 February 2016
Accepted 23 February 2016
Available online 3 March 2016

Keywords:
Machine learning
k-NN and decision trees
Feature extraction and selection
Volcano seismic classification
Seismic event detection

ABSTRACT

Volcano Early Warning Systems (VEWS) have become a research topic in order to preserve human lives and material losses. In this setting, event detection criteria based on classification using machine learning techniques have proven useful, and a number of systems have been proposed in the literature. However, to the best of our knowledge, no comprehensive and principled study has been conducted to compare the influence of the many different sets of possible features that have been used as input spaces in previous works. We present an automatic recognition system of volcano seismicity, by considering feature extraction, event classification, and subsequent event detection, in order to reduce the processing time as a first step towards a high reliability automatic detection system in real-time. We compiled and extracted a comprehensive set of temporal, moving average, spectral, and scale-domain features, for separating long period seismic events from background noise. We benchmarked two usual kinds of feature selection techniques, namely, filter (mutual information and statistical dependence) and embedded (cross-validation and pruning), each of them by using suitable and appropriate classification algorithms such as *k* Nearest Neighbors (*k*-NN) and Decision Trees (DT). We applied this approach to the seismicity presented at Cotopaxi Volcano in Ecuador during 2009 and 2010. The best results were obtained by using a 15 s segmentation window, feature matrix in the frequency domain, and DT classifier, yielding 99% of detection accuracy and sensitivity. Selected features and their interpretation were consistent among different input spaces, in simple terms of amplitude and spectral content. Our study provides the framework for an event detection system with high accuracy and reduced computational requirements.

1. Introduction

Volcano monitoring systems have been deployed as an attempt to mitigate risks, to forecast eruptions, and to assess hazards, due to the necessity of safeguarding human lives and resources. These monitoring systems use principled information related to ground deformation (Dzurisin, 1980; Dvorak and Dzurisin, 1997; Voight et al., 1998; Bonaccorso et al., 2006), gas flux (Baubron et al., 1991; Galle et al., 2003; Lewicki et al., 2003), seismicity (McNutt, 1996; Chouet and Matoza, 2013; Sparks, 2003), and other factors, as main monitoring measurements to determine the activity of volcanoes. In this context, seismology is an important and effective tool for monitoring volcanoes, since seismicity is the fastest and most commonly used method in order to detect changes on volcanoes, by assessing earthquakes and other ground vibrations sensed by the seismometers or geophones networks (McNutt, 2000; Sicali et al., 2015; Papadimitriou et al., 2015). The events recorded in these systems present differences in their seismic wave patterns so their seismological signature can be interpreted by analysts to identify different types of events. For instance, most volcanoes present Volcano Tectonic (VT) earthquakes, Long Period (LP) events, Tremors (TRE), and Hybrid (HYB) events. Other non-volcanic originated events, such as Lightnings (LGH), can be occasionally recorded by seismometers (Behnke et al., 2013).

Several techniques have been developed for automatic identification of events, such as stochastic processes analysis, mathematical modeling, and signal processing in time, frequency, and scale domains (Scarpetta et al., 2005). The latter refers to the use of the wavelet transform for a time-scale domain analysis of signals with fast changing spectral contents, using wavelets it is possible to represent a signal in a time and frequency response scale where any event contained in the signal will mark an entire region in the time-scale plane, solving therefore the

* Corresponding author at: Departamento de Eléctrica y Electrónica, Universidad de las Fuerzas Armadas ESPE, Av. Gral. Rumiñahui s/n, Sangolquí, Ecuador.
E-mail addresses: ralara@espe.edu.ec (R.A. Lara-Cueva), dbenitez@usfq.edu.ec (D.S. Benítez), evcarrera@espe.edu.ec (E.V. Carrera), mruiz@igepn.epn.edu.ec (M. Ruiz), joseluis.rojo@urjc.es (J.L. Rojo-Álvarez).
URL: http://www.wicom.espe.edu.ec (R.A. Lara-Cueva).

http://dx.doi.org/10.1016/j.jvolgeores.2016.02.022
0377-0273/

resolution problem presented by the Fourier transform in the frequency domain. The stated advantage of using wavelets compared to Fourier transform is the trade-off between frequency and time resolution at different frequencies. Special attention has been put in classification techniques from Machine Learning Theory, since these methods are capable of describing in detail patterns from different types of events, as it will be summarized latter in Section 2. However, to the best of our knowledge, no comprehensive and principled study has been conducted in previous works to compare the influence of the many different sets of features that have been used to approximate the input space, defined in terms of the possible values that the input parameter can have.

The aim of this work is to present an automatic recognition system for volcano seismicity, by considering all the stages of the process including feature extraction, feature selection, and event classification, and in order to provide us with a high-fidelity event detection, as a first step to an automatic detection system in real-time. Our primary hypothesis is that a carefully designed feature extraction with a suitable and appropriate machine learning technique will reduce the processing time and will avoid overfitting. We address here the optimal design of a detection system, based on classification techniques, for separating LP seismic events from seismic background noise with high accuracy, since LP events often precede volcanic eruptions and they are accordingly used in forecasting (Chouet and Matoza, 2013; Chouet, 1996; Trombley and Toutain, 2005; Lyons et al., 2014; Bean et al., 2014; Cusano et al., 2015; Syahbana et al., 2014). The classification among the other types of events is beyond our scope at this stage, but it could be specifically addressed with the proposed approach. We benchmarked two commonly used feature selection techniques, namely, filter and embedded, each of them in a suitable and appropriate classification algorithm, namely, *k* Nearest Neighbors (*k*-NN) and Decision Trees (DT).

Our study refers to Cotopaxi, an active volcano, located in the so-called Ring of Fire at Ecuador, in which a permanent monitoring system (24 h/7 days a week) has been previously deployed (Ortiz Erazo, 2013). Its activity produced over 100 MB of data per day during 2009 and 2010, which means an average of 21 and 17 events per day, respectively. There are 16 seismological stations deployed at Cotopaxi Volcano as highlighted in Section 3, therefore, expert scientists must daily analyze the vertical component of 16 seismograms of volcanic signals by visual inspection in order to label and classify the events. Currently, its activity is increasing and presents 130 events on average per day, too many records are generated during periods of high volcanic activity; therefore data assessment can become an extremely slow process, which can also cause damming of information (Newman and Jain, 1995; Mery and Medina, 2004).

The rest of the paper is organized as follows. Section 2 summarizes previous works and results about the automatic classification of seismic events by using machine learning techniques. Section 3 describes the dataset used in this work, which were collected at Cotopaxi Volcano. Section 4 describes the proposed approach and the experimental study including feature extraction and event detection. Section 5 presents the results obtained in automatic classification and detection processes for different segmentation windows and with different classifiers. Finally, Section 6 presents the discussion and some concluding conclusions.

2. Detecting seismic events

One of the main goals of volcano monitoring institutions around the world is to understand the behavior of volcanoes, in order to forecast a possible or imminent eruption for safeguarding lives, which is achievable by sensing and distinguishing the increase of volcanic events. In relation to this, the analysts visually identify seismic events received from remote seismometers by actually using digital signal processing techniques both in the time and the frequency domain to make this process more efficient, so event data, such as timestamps of start and end, time duration, and arrival times, can be stored for future reference.

In this context, several techniques have been developed to support vulcanologist in the automatic classification process, and some authors have used supervised (Falsaperla et al., 1996; Langer et al., 2003, 2006; Curilem et al., 2009) or unsupervised (Messina and Langer, 2011; Esposito et al., 2008; Ohrnberger, 2001) learning techniques, in order to distinguish among two or more classification groups of events.

In Ibáñez et al. (2009), for example, Etna and Stromboli volcanoes were studied in terms of VT and TRE events for the first volcano, and background noise and Very Long Period (VLP) events for the second one. Authors found a total of 39 data parameters of main temporal and spectral characteristics, including coefficients of the time evolution of the signal and the energy in a frequency band, by using Hidden Markov Models (HMM) as classifier, yielding 86% and 84% of successful classification rates, respectively.

Meanwhile in Álvarez et al. (2012), a 1 to 25 Hz band-pass filter, first using a time windowing of 4 s and then extended it to 8 s was used to extract temporal and spectral characteristics from data obtained at Colima Volcano in Mexico, yielding two proposed feature vectors with 39 and 84 features, extended the feature vector defined in Ibáñez et al. (2009) by considering the presence or absence of harmonics and the spectral envelope. Discriminative Feature Selection (DFS) based on the Minimum Classification Error (MCE) criterion, and a Gaussian Mixture Model (GMM) were used, which reduced the original sets to 14 and 10 features, respectively. The misclassification percentage was improved for the first feature vector set from 24% to 16%, and for the extended feature vector from 28% to 14%. However, the main features, which improved the results, were not mentioned in such work.

A feature extraction and a subsequent feature selection steps were developed for Nevado del Ruiz Volcano in Colombia (Cárdenas-Peña et al., 2013), considering VT, LP, TRE, and HYB events. In this work, a feature selection strategy was developed based on the relevance of time variant features, i.e., the most significant set of features or those with the greatest contribution to the event, and the results were compared to the use of HMM and *k*-NN. With this approach, the classification error rate was improved from 22% to 12% when using *k*-NN instead of HMM. Another system was defined in Cortés et al. (2014), which used the GMM classifier, which obtained a baseline recognition rate of 92% by using the feature vector with the main features via DFS at Deception Volcano Island in Antarctica.

Although previous works have demonstrated the possibilities of using machine learning techniques in this setting, the literature lacks of supportive evidence about which are the main design parameters to be considered in each stages for signal preprocessing, feature extraction, feature selection, classification, and detection, especially for real-time or near-real-time detection systems.

3. Dataset description

The *Instituto Geofísico de la Escuela Politécnica Nacional* (IGEPN) is the institution responsible for monitoring the seismic activity in Ecuador. IGEPN has installed a high quality seismometer network, covering near 70% of the country. The monitoring stations collect data every day and continuously transmit them to a volcano observatory, which is located 40 km from the Cotopaxi Volcano. These data are transmitted by using radio links in the UHF band.

Cotopaxi is a snow-capped volcano located at latitude 00° 41′ 05″ S and longitude 78° 25′ 54.8″ W in the Andean mountain region of Ecuador. We have chosen this particular volcano for our study given its high hazard and risk status of future eruptions. Cotopaxi has experienced 5 eruptive cycles with 13 significant eruptions since 1534, and past eruptions have produced pyroclastic flows, ash and lapilli falls, lava flows, and far reaching lahars (Hall and Mothes, 2008). The Cotopaxi Volcano is located 40 km from Quito and near to the city of Latacunga, and a potential eruption will directly affect about 800,000 people living in the surrounding area of the volcano. However, being also so close to Quito, a city with a population of over two million inhabitants, the number of people that

may be affected by an eruption is extremely high. Furthermore, the Cotopaxi Volcano is located in the center of Ecuador, and an eruption will be a national catastrophe of major proportions, since it will affect all regions of the country. Major highways cross nearby and an eruption will directly affect traffic connections between all regions. The cost for the government will be extremely high.

As depicted in Fig. 1, the IGEPN currently has installed at Cotopaxi Volcano: (a) five short period (SP) seismological stations (PITA, NAS2, VC1, CAMI, and TAMB), four of them with vertical component sensors and two of them with three components sensors, and all of them with response frequency range of 1–50 Hz; (b) eleven broadband (BB) stations (TOMA, SUCR, BVC2, BREF, BNAS, BTAM, BMOR, SLOR, SRAM, BRRN and VCES), with response frequency range of 0.1–50 Hz (Córdova Regalado, 2013). Data used in our study were recorded during 2009 and 2010 at the VC2 station, located 3 km from the Cotopaxi summit, by using a triaxial broadband seismometer CMG-40T Güralp with a sensitivity of 1600 V/ms^{-1}. A Short-Term/Long-Term Average ratio triggering algorithm is used by this seismometer to detect the seismic activity. This algorithm evaluates the ratio of short-to-long-term energy density, for further details on theoretical and algorithmic details, see Withers et al. (1998). Then, the seismograms are recorded in files of 12,000 s duration, which are digitized at 100 Hz sampling rate by using a 12 bit analog to digital converter. Data were taken from the VC2 broadband station in the vertical axis, since this site was less noisy than others, and since data obtained from this station displayed the highest signal to noise ratio, with high acquisition quality and reliability. Dataset consisted of 759 LP, 116 VT, 30 HYB, and 9 TRE events.

Fig. 2(a) shows some examples of typical events at Cotopaxi Volcano, where we can observe a peak around 3000 s corresponding to an LP event, with a typical duration between some few seconds to more than 1 min. These events exhibit different spectral peaks in the frequency range between 2 and 7 Hz, as depicted in Fig. 2(b) and (c), respectively. Additionally, we can observe another peak around 6000 s corresponding to a LGH event, which presents a typical spectrum in the frequency range between 17 and 21 Hz, as shown in Fig. 2(e), and another prominent peak at 9000 s corresponding to a VT event with a typical duration below 30 s and a spectral content around 9 Hz, as depicted in Fig. 2(f) and (g), respectively. Note that since seismometers are extremely sensitive, very small variations produced by wind and other surface effects can also be sensed, hence producing background noise whose main contribution is in the form of white noise, a random signal with a constant power spectral density.

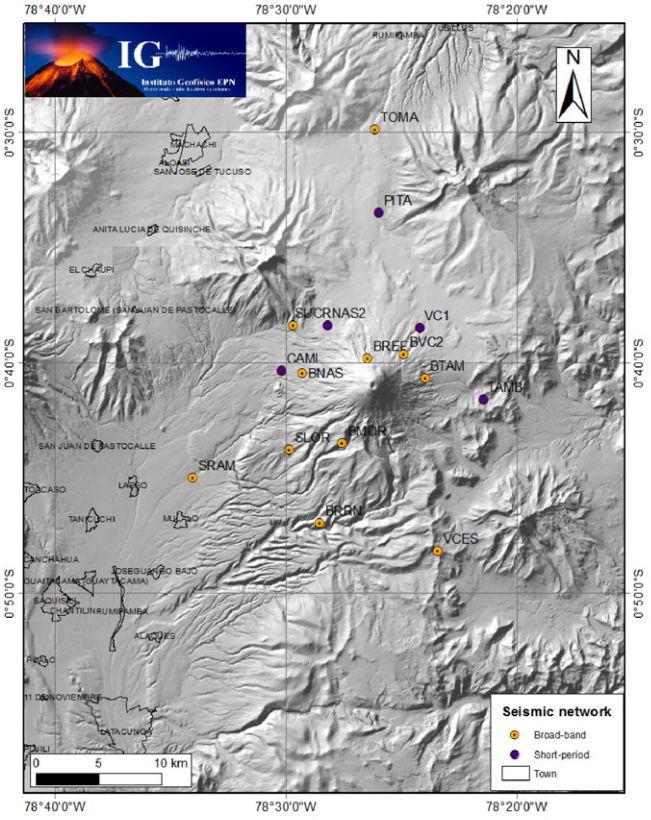
4. Methodology

4.1. Preprocessing stage

Fig. 3 shows the scheme of the proposed system, which can be divided into 4 components. The first one consists of a preprocessing stage, the second one accounts for both feature extraction and feature selection, which are necessary in order to reduce both the amount of data and the processing time required, and to avoid overfitting. The third component includes some degree of intelligence by using machine learning techniques in order to define a model to classify new data. As mentioned earlier, our first approach to develop the event detector will be to focus only on LP events, therefore the last component will detect LP events, and allowing the determination of the performance of the entire system for both, classification and detection capabilities.

Initial spectral analysis of the seismic signals shows a permanent peak located at 0.2 Hz, most probably related to sea microseisms events (Kenneth, 2001). The main portion of spectral content for LP events is known to be between 2 to 7 Hz, therefore a 128th order band-pass finite impulse response (FIR) with bandwidth (1, 15) Hz was applied to each

(a) Map of Ecuador, showing location of Cotopaxi Volcano



(b) Seismic Monitoring System

Fig. 1. Location of Cotopaxi Volcano and the deployed seismological stations. Dataset has been taken from the VC2 BB station.

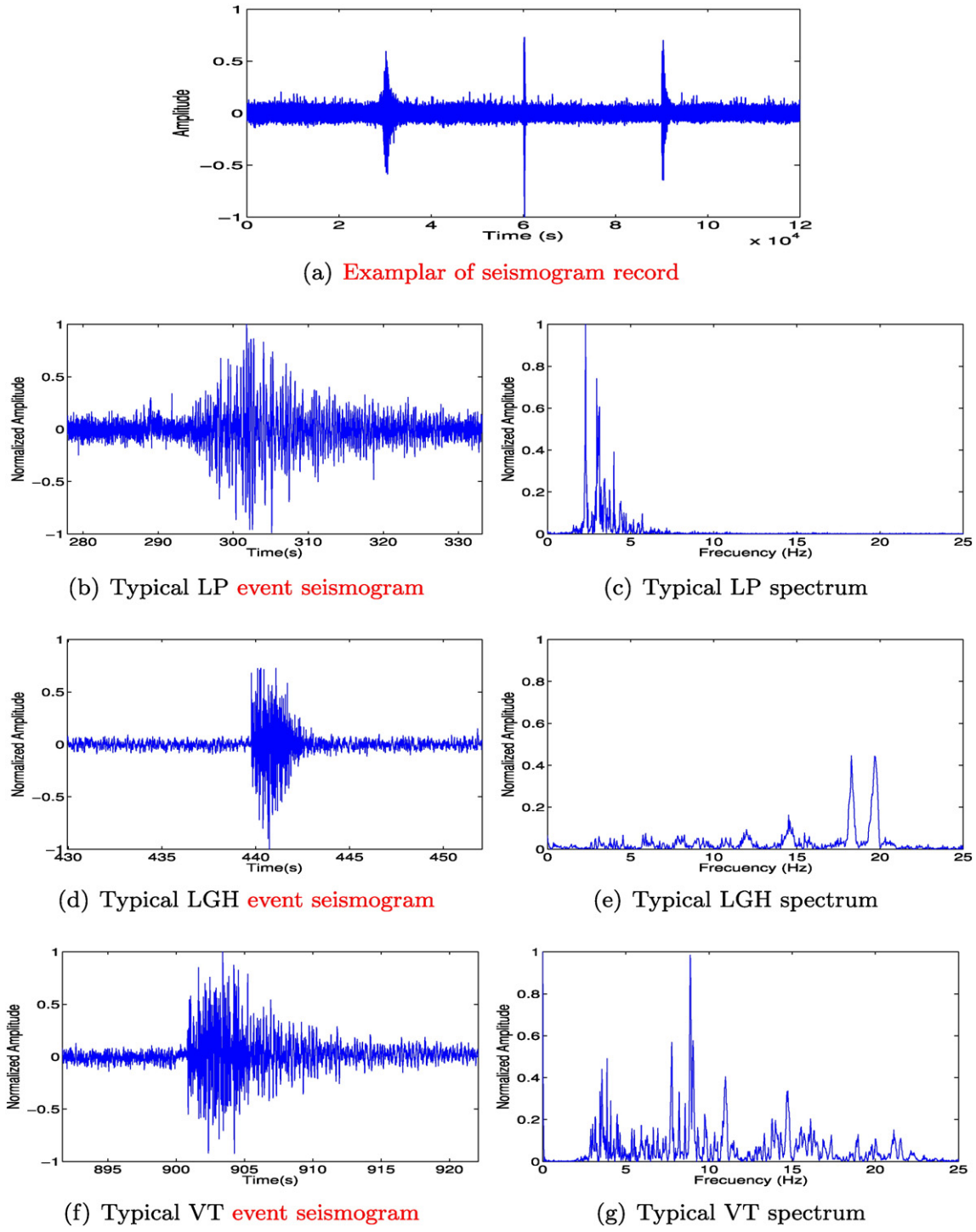


Fig. 2. Examples of seismicity records at Cotopaxi Volcano: (a) the three most common event types; (b) a LP event is presented around 3000 s; (c) typical LP spectrum; (d) a LGH event is presented near to 6000 s; (e) typical LGH spectrum; (f) a VT event is presented around 9000 s; (e) typical VT spectrum.

volcano seismic record $\mathbf{r}_i$, $\mathbf{r}_i \in \Re^l$, where l is the number of samples in the records, and with $i \in 1,\ldots,N$, where N is the number of available recordings. This step also eliminated LGH events, whose spectral content is above 17 Hz. Then, classical centering and scaling of the data was used for each $\mathbf{r}_i$, giving the z-score normalized recording, as follows,

$$\mathbf{r}_i = \frac{\mathbf{r}_i - \mu_i}{\sigma_i}, \tag{1}$$

where $\mathbf{r}_i$ are the normalized recordings, and μ_i and σ_i are their mean and standard deviation, respectively. Finally, different sliding window lengths of w samples were applied to each $\mathbf{r}_i$ without overlapping, yielding $j = \frac{l}{w}$ segments for each recording. This provided us with a windowed signal matrix, $\mathbf{S}_i$, defined as follows,

$$\mathbf{S}_i = \left[\mathbf{s}_{i,1}^T, \mathbf{s}_{i,2}^T, \ldots, \mathbf{s}_{i,j}^T\right]^T, \tag{2}$$

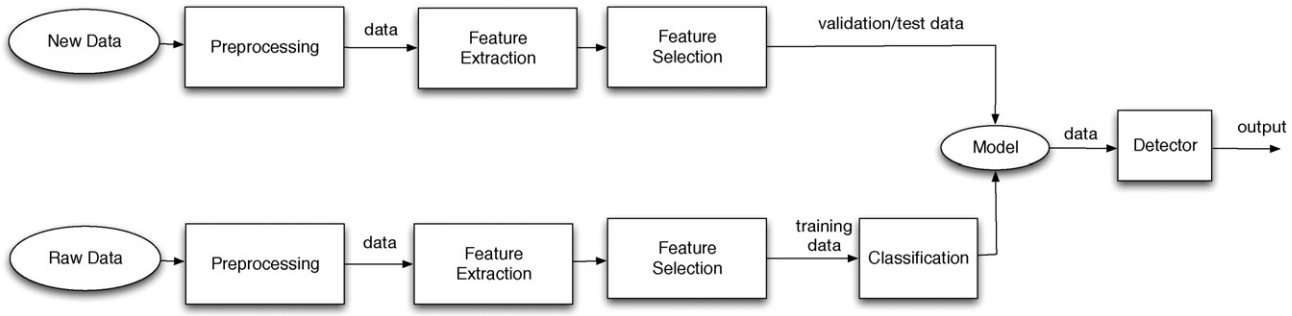


Fig. 3. Scheme for the event detection based on classification.

where $\mathbf{s}_{i,j}$ are the windowed segments of the i-th recording. Then, the data matrix is given by the signal matrices from all the available recordings, this is,

$$\mathbf{S} = \left[\mathbf{S}_1^T, \mathbf{S}_2^T, ..., \mathbf{S}_N^T\right]^T. \tag{3}$$

Note that this data matrix consists of samples from the same and different signals, which has to be taken into account for the machine learning free parameter search in order to avoid overfitting.

4.2. Feature extraction and selection stage

We developed a feature extraction framework in order to identify a set of relevant features from each row of matrix **S**, giving a feature vector $\mathbf{x}_{i,j} = g(\mathbf{s}_{i,j})$, where $g()$ is the feature extraction operator for which different options can be scrutinized, and they are summarized next.

In order to obtain matrices in the time domain, a simple data matrix was considered, corresponding to original matrix **S** with different segmentation window sizes w. In addition, another time data matrix corresponding to $\mathbf{t}_{i,j} = g_{MA}(\mathbf{s}_{i,j})$ was regarded, where $g_{MA}()$ is the signal operator squaring each sample and smoothing with a 1st order Moving Average (MA) filter (Chen and Chen, 2003). The main parameters of the MA filter considered the previous values of w, in order to maximize the variations among data and to cancel low values, and 50% of window overlapping. Then, our envelope-band yielded for each recording,

$$\mathbf{T}_i = \left[\mathbf{t}_{i,1}^T, \mathbf{t}_{i,2}^T, ..., \mathbf{t}_{i,j}^T\right]^T, \tag{4}$$

and the data matrix was often given by signal matrices $\mathbf{T}_i$ from all the available recordings, this is,

$$\mathbf{T} = \left[\mathbf{T}_1^T, \mathbf{T}_2^T, ..., \mathbf{T}_N^T\right]^T. \tag{5}$$

Both matrices had m rows and n columns, given by $m = N \times j$ instances and $n = w$ features.

In order to obtain matrices in the frequency domain, the Power Spectral Density (PSD) with Welch method was obtained for each row of matrix **S**, yielding feature vectors $\mathbf{f}_{i,j} = g_h(\mathbf{s}_{i,j})$, where g_h is the operator yielding the PSD of the row time vector. The main parameters of the Fast Fourier Transform (FFT) were 50% of window overlapping, each section was windowed with a Hamming window with length equal to the segmentation values, and 1024 points of FFT for frequency representation resolution, yielding

$$\mathbf{F}_i = \left[\mathbf{f}_{i,1}^T, \mathbf{f}_{i,2}^T, ..., \mathbf{f}_{i,j}^T\right]^T. \tag{6}$$

The feature matrix was then given by signal matrices $\mathbf{F}_i$ from all the available recordings. In order to have a moderate resolution in frequency and time, we limited to 512 points of FFT and maintained the other of parameters, yielding $\mathbf{g}_{i,j} = g_n(\mathbf{s}_{i,j})$, where g_n is the operator yielding the PSD of the row time providing with moderate resolution for each recording, i.e.,

$$\mathbf{G}_i = \left[\mathbf{g}_{i,1}^T, \mathbf{g}_{i,2}^T, ..., \mathbf{g}_{i,j}^T\right]^T, \tag{7}$$

and the data matrix was given by signal matrices $\mathbf{G}_i$ from all the available recordings. Both matrices had $m = N \times j$ instances, whilst the number of features was $n = \frac{n_f}{2} + 1$, where n_f are the points of the FFT, yielding 513 features for high resolution and 257 features for moderate resolution.

In order to obtain data matrices in the scale domain, a wavelet transform was applied to each row of matrix **S**, yielding $\mathbf{w}_{i,j} = g_w(\mathbf{s}_{i,j})$, where $g_w()$ is the operator applying a 10th order *symlet* as mother wavelet, by following the consideration of similarity (Ngui et al., 2013), and with details of decomposition with level 4, since this level has the main energy component of the signal. This yielded for each recording

$$\mathbf{W}_i = \left[\mathbf{w}_{i,1}^T, \mathbf{w}_{i,2}^T, ..., \mathbf{w}_{i,j}^T\right]^T, \tag{8}$$

and the data matrix was given by all the signal matrices $\mathbf{W}_i$ from all the available recordings. This matrix had $m = N \times j$ instances and n features, corresponding to the coefficients at 4th level of decomposition according to w value.

Note that, at this point, five types of feature matrices have been defined, two in the time domain (**S**, **T**), two in the frequency domain (**F**, **G**), and one in the scale domain (**W**).

Feature selection techniques are often used to identify the most relevant features, which are able to enhance classification performance. In this setting, we benchmarked two commonly used techniques, namely: filter and embedded. Filter algorithms are independent of the classifier by using a search criterion function, commonly a heuristic method resulting from practical procedures, based on general features like mutual information, correlation, or statistical dependence with the variable to predict. The embedded methods, in the other hand are immersed in the classification algorithm, which realizes a recursive partitioning of the data by splitting it into sub-sets based on features that are the most useful in distinguishing between different data classes. The process is termed recursive since each sub-set may in turn be split an indefinite number of times until a particular stopping criterion is reached.

Table 1
Matrix size in terms of the selected segmentation value w.

Matrix	$w = 5$ s	$w = 15$ s	$w = 30$ s
S, **T**	60,720 × 500	20,240 × 1500	10,120 × 3000
F	60,720 × 513	20,240 × 513	10,120 × 513
G	60,720 × 257	20,240 × 257	10,120 × 257
W	60,720 × 508	20,240 × 824	10,120 × 1621

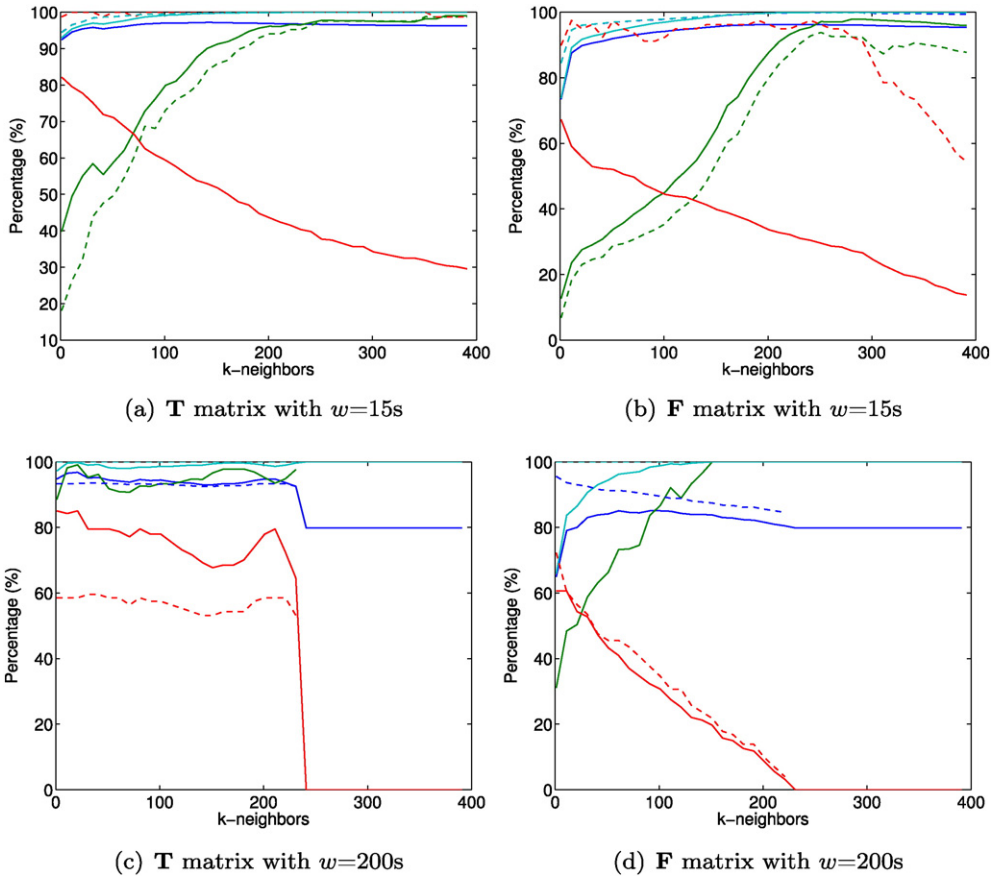


Fig. 4. Segmentation comparison as function of k for classification (solid line) and detection (dotted line), in terms of A (blue), P (green), R (red), and S (cian). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Both methods were applied in order in order to obtain matrices $\mathbf{S}'$, $\mathbf{T}'$, $\mathbf{F}'$, $\mathbf{G}'$, and $\mathbf{W}'$, which contain most of the discriminative information required to classify the events while avoiding overfitting.

We are using a pre-labeled dataset for our study, which was manually labeled by human analysts and provided to us by IGEPN. We are working with supervised learning, and therefore we need a known dataset for training the algorithms. We assumed that this dataset was correctly labeled, and we adopted it as ground truth being used for validating the learning algorithm.

Each segment $\mathbf{s}_{i,j}$ was automatically labeled as either class $y_{i,j} = +1$ or $y_{i,j} = -1$, depending if the signal contained by the segment is part or not of a LP event according to their timestamps and the corresponding label of the signal in the dataset, therefore consecutive segments $\mathbf{s}_{i,j}$ with label $+1$ will contain the entire LP event.

4.3. Classification algorithms

We considered supervised learning and used k-NN and DT as machine learning techniques, which are summarized next.

k-NN is one of the most basic instance-based classification methods, it assumes all instances corresponding to points in the space $\Re^n$, where n is the dimensionality of the input space belonging to well known classes or clusters, by forming a reference model in which a new instance vector will be assigned to the more frequent class or cluster belonging to their k nearest neighbors (Mitchell, 1997). The nearest neighbors of an instance are often defined in terms of the standard Euclidean distance. Let $\mathbf{x}$ be an arbitrary instance described by feature vector $[a_1(\mathbf{x}),a_2(\mathbf{x}),\ldots a_n(\mathbf{x})]^T$, where $a_r(\mathbf{x})$ denotes the value of the rth attribute of instance $\mathbf{x}$. The distance between two instances $\mathbf{x}_i$ and $\mathbf{x}_j$ is defined to be $d(\mathbf{x}_i,\mathbf{x}_j)$, giving

$$d(\mathbf{x}_i,\mathbf{x}_j) \equiv \sqrt{\sum_{i=1}^{n}\left(a_r(\mathbf{x}_i)-a_r(\mathbf{x}_j)\right)^2}. \tag{9}$$

The tuning of the k-NN classifier is carried out by calculating the optimal number of neighbors k, which is the free parameter for this classifier, which allows us to obtain the best performance in the validation set.

DT is a non-parametric supervised learning method used for classification and regression, and one of the most widely used methods for inductive inference (Bishop et al., 2006). The goal of this method is to create a tree model formed by a root and several leaves, based on conditions in order to categorize a set of consequently rules and to select the next branch, which can predict the value of a target variable by learning simple decision rules inferred from the data features. The paths from root to leaf represent the classification rules and the outcome of each leaf node is obtained after evaluating all the conditions along the path. DT classifies new instances by sorting them down the tree from the root to some leaf node, thereby providing the instance classification. Each node in the tree specifies a rule of some attribute of the instance, and each branch descending from that node corresponds to one of the possible values for this attribute. The measures include the average amount of information contained in each event. It is robust to noisy data, and searches in a completely expressive hypothesis space and, hence avoiding the difficulties of restricted hypothesis spaces. Small rather than large trees are preferred, in order to promote classifiers

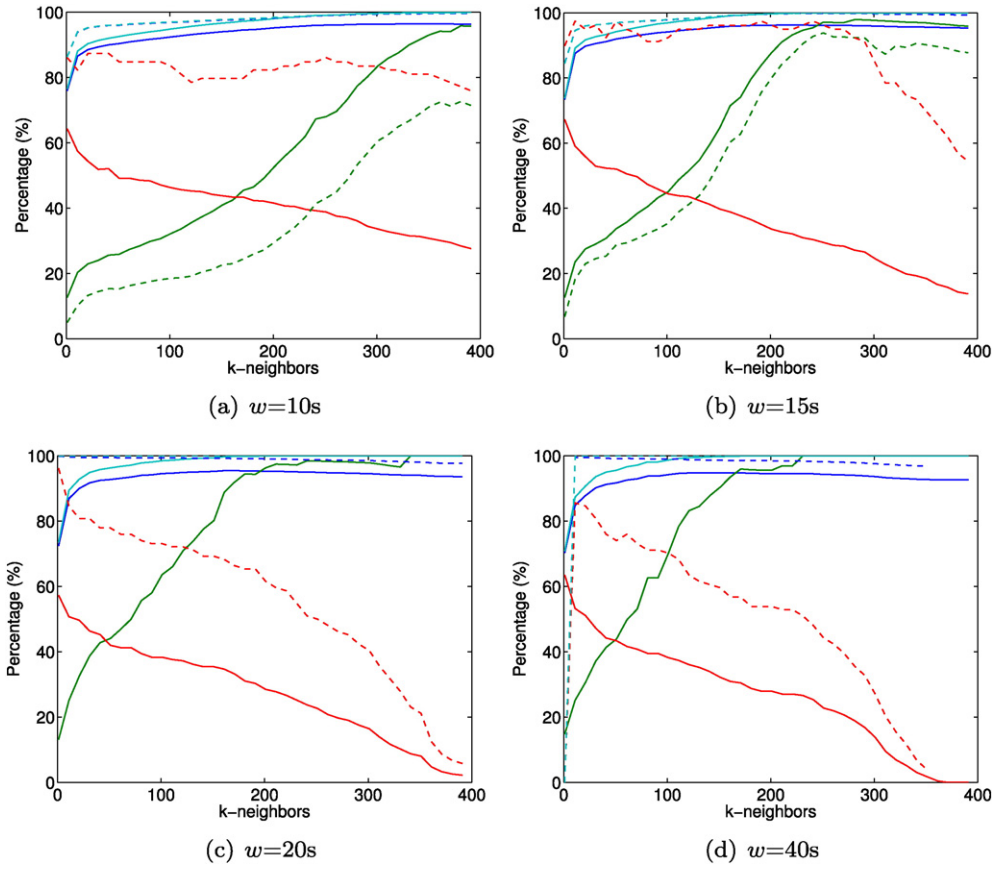


Fig. 5. Segmentation comparison with **F** matrix as function of k for classification (solid line) and detection (dotted line), in terms of A (blue), P (green), R (red), and S (cian). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

with good generalization capabilities. Currently, many algorithms have been developed for learning DT, which obtain a top-down, greedy search in the space of possible DT. The depth or *leafiness* of the tree is the free parameter for this machine learning technique, and it has to be adjusted for maximizing the classification performance in the validation set while avoiding overfitting to the training set.

4.4. Detection and performance

We developed an event detector based on the previously described classification algorithms, given that the output of the classifiers predicts a class y (either $+1$ or -1) for each segment $\mathbf{s}_{i,j}$, and then LP events will contain some $\mathbf{s}_{i,j}$ labeled with $y = +1$ depending on the w value. We created a detection algorithm in order to improve the classification performance, which was adjusted to detect LP events from background noise based on verifying the labels and by marking the start and end times of $+1$ or consecutive $+1$ output for signal windows in a given record. We identified a threshold of possible LP event duration, which maximized the detection performance of the system for LP events.

Once a prediction model has been constructed, the training data set is presented to the algorithm for calibration, and the classifiers will assign a value for each new data instance (w segment). This value will be either $+1$ for segments that the classifier considers are part of an LP event or -1 for segments that the classifier judges are not part of the LP event. A transition from a low to high value is considered the beginning of an event, and similarly, the transition from a high to a low value is considered the ending of the event. From these values the time duration of a given event can be calculated. In the training stage, the timestamps of possible events are compared with the real timestamps of the events annotated in the training set, and the algorithm automatically determines the time duration threshold that maximizes the accuracy of the classifier. Possible events with time duration lower than this time threshold will not be considered as real events, hence this post-processing stage (detector) will reduce the number of false positives.

When trying to fit models to a dataset, the common advice is to partition the data into three parts: training, validation, and test datasets, as illustrated in Fig. 3. The training set is used to train the model; the initial model parameters are selected by choosing the parameters that minimize the errors on the training set. After that, the parameters of the model are tuned by minimizing errors on the validation set. Therefore, a validation set, which is independent from the training set, is used for parameter selection and to avoid overfitting. Finally, the performance of the trained model is tested on the testing set. If the model is trained on a training set only, it is very likely to get 100% accuracy and overfit, resulting on a very poor performance on the testing set.

Hence, the proposed methodology was applied to the dataset from Cotopaxi Volcano. In order to make comparisons, the entire dataset was divided into training, validating, and testing sets, so that each set consisted of 253 records. For testing and validating sets, the dataset was split aleatory for each matrix by months, however, data independency, a necessary condition for machine learning algorithms, was guaranteed by assigning each $\mathbf{s}_{i,j}$ belonging to the same signal to different sets. Additionally, a comparison between k-NN and DT classifiers was performed. The experiments were carried out using Matlab R2013a, on a Core I5 PC with 3.1 GHz and 4 GB RAM.

The classification and detection performance was measured in terms of Accuracy (A), Precision (P), Sensitivity or Recall (R), and Specificity (S), which are defined as follows:

$$A(\%) = \frac{N_C}{N_T} \times 100, \tag{10}$$

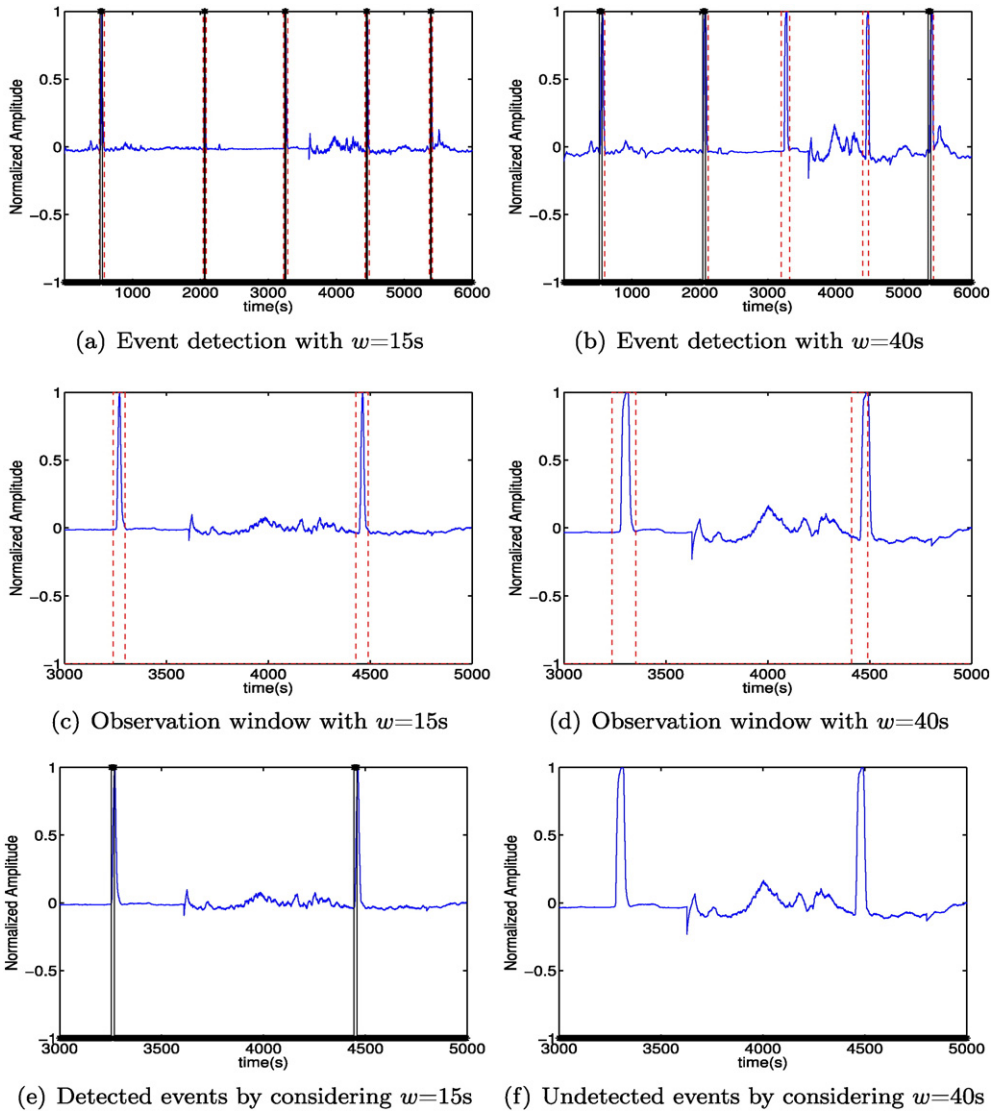


Fig. 6. Performance comparison of LP seismic detector for optimal observation window w selection when considering k-NN classifier, and with features in matrix **T**. Example considering 5 continuous events (blue solid line) within the seismogram and using 2 different observation windows (red dotted line). Insets (a), (c) and (e) show detection results using an observation window of $w = 15$ s, while insets (b), (d) and (f) show results with an observation window of $w = 40$ s. Detected events are denoted by the black solid line. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

$$P(\%) = \frac{N_{TP}}{N_{TP} + N_{FP}} \times 100, \tag{11}$$

$$R(\%) = \frac{N_{TP}}{N_{TP} + N_{FN}} \times 100, \tag{12}$$

$$S(\%) = \frac{N_{TN}}{N_{TN} + N_{FP}} \times 100, \tag{13}$$

Table 2
Experimental results for detection by using k-NN classifier, with $w = 15$ s, and validation/test sets. The optimal values of k reported in the table are the optimum values that maximize the metrics performance.

Matrix	k	A (%)	P (%)	R (%)	S (%)
S	11	99/96	100/97	18/12	100/99
T	351	97/96	100/60	40/18	100/99
F	241	97/99	78/93	39/98	99/99
G	271	97/99	87/92	33/92	99/99
W	11	96/99	60/62	18/40	99/99

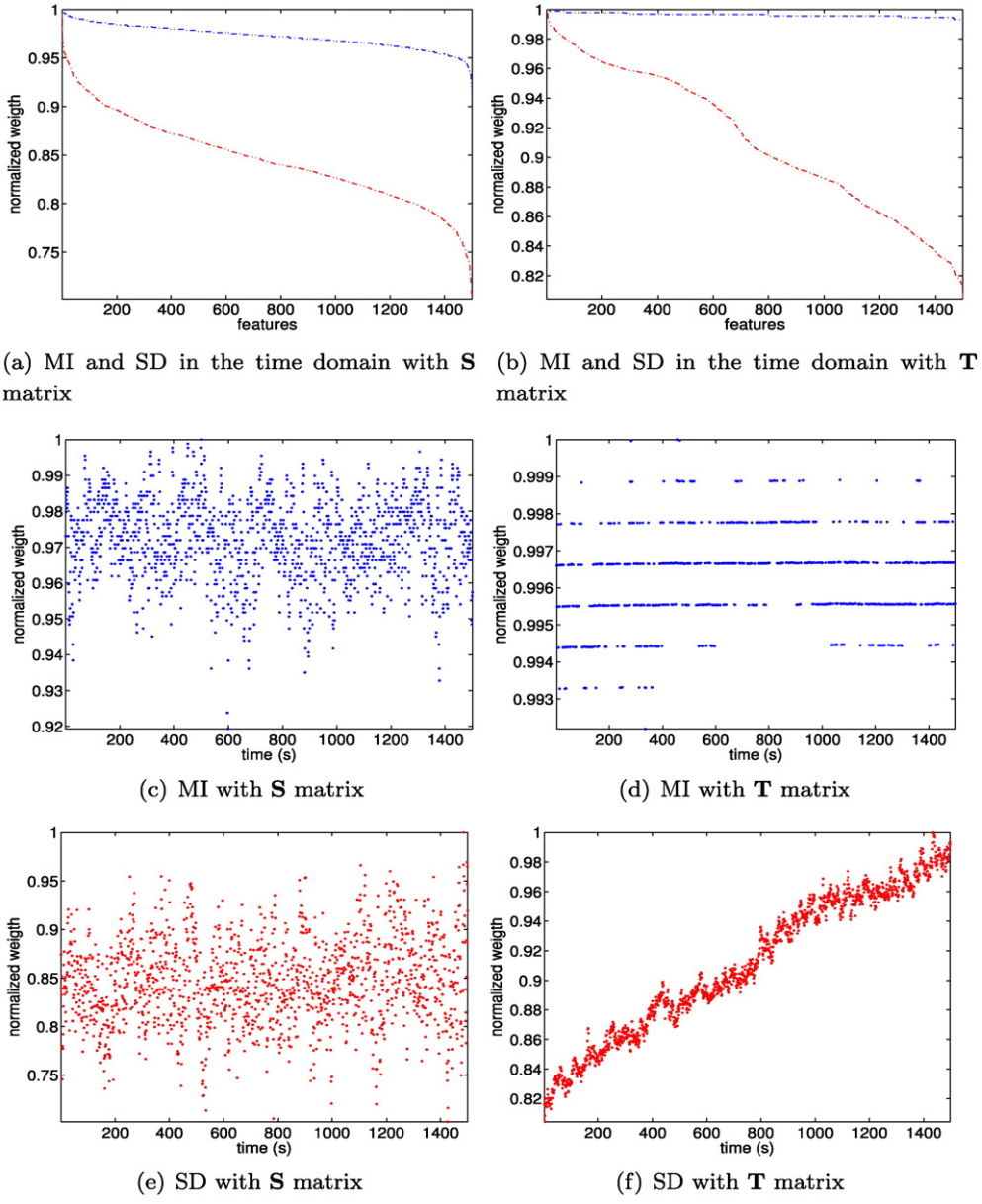
where N_C is the number of correctly classified patterns, N_T is the total number of patterns used to feed the classifier, N_{TP} is the number of true positives, N_{FN} is the number of false negatives, N_{TN} is the number of true negatives, and N_{FP} is the number of false positives. We calculated these performance measures for each validation and test folds.

5. Results

5.1. Segmentation tuning

We worked with different values of window w, so the number of segments for analysis directly depended of the chosen size of this window. For example, for a given record with a duration of 1200 s, if $w = 15$ s, then the number of segments is 80 for the entire record, meanwhile if $w = 200$ s is chosen, then the number of segments is only 6. Therefore, the sizes of the matrices will be different.

Table 1 shows some examples of this variation in terms of the original number of features and levels for different values of w. As it can be seen, the number of instances increases for smaller values of w, while the number of features depends directly on the value of w, except



(a) MI and SD in the time domain with **S** matrix

(b) MI and SD in the time domain with **T** matrix

(c) MI with **S** matrix

(d) MI with **T** matrix

(e) SD with **S** matrix

(f) SD with **T** matrix

Fig. 7. Normalized weights variation of recording segments in the time domain by using MI (blue dotted line) and SD (red dotted line) methods. Inset (a) shows continuous smooth changes on normalized weights as function of the relevant features; inset (b) shows that inflection changes are produced on normalized weights curve around the 200 and 600 features; insets (c) and (e) show the normalized weights variation according to the feature relevance in the time domain when using **S** matrix; insets (d) and (f) show the normalized weights variation in the time domain, after the feature selection method has been applied, when using **T** matrix. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

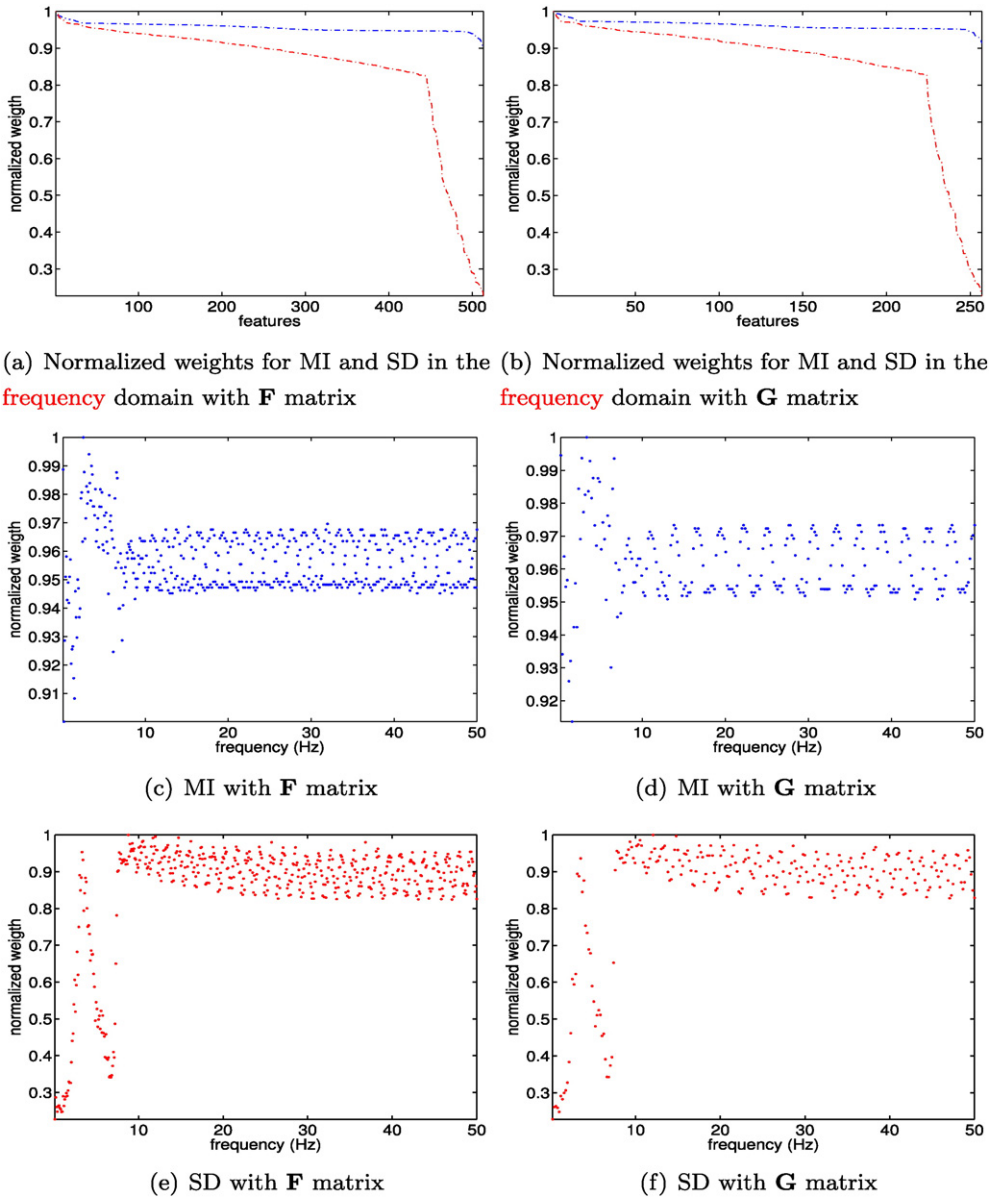
for **F** and **G** matrices, which are independent of the value of w since the characteristics are constant.

We explored different values of w ranging from 5 s to 200 s (100 time samples correspond to 1 s), in order to set up the optimal value maximizing the system performance. Fig. 4, for example, shows the results obtained when values of 15 s and 200 s were used for w. Performance was evaluated in terms of both classification and detection by using the k-NN classifier, however, the best results were obtained when only **T** and **F** matrices were used. With $w = 15$ s, the performance in classification and detection was better than when using $w = 200$ s, and values of less than 15 s were discarded since the algorithm took a long time for processing (around 10 min), which would be unacceptable in order to satisfy real-time requirements.

We considered P and R metrics for setting up the w value. In terms of classification, we obtained better results with **T** matrix and $w = 200$ s, where values between 60% to 84% were obtained for R when considering k of about 240, approximately, whilst P maintained its value near to 99%, as depicted in Fig. 4(c). Meanwhile, we obtained better results in terms of detection with **T** matrix and $w = 15$ s, as far as R obtained values above 95% and P increased with k, reaching values close to 99% (see Fig. 4(a)).

Considering all the performance metrics together, we concluded that $w = 15$ s should be used as a good option for segmentation in the time window. This value provides the best performance for both detection and correct classification, while allowing moderate computational burden. This assessment was corroborated by Fig. 5, where it can be observed that the performance metrics in detection were improved, and the main parameter which allowed us to take a decision was R in detection (dotted-red). Fig. 5 shows that R in detection reaches its maximum value when $w = 15$ s.

Nevertheless, we verified this value $w = 15$ s by additionally comparing these results with those obtained when $w = 40$ s, as seen in



(a) Normalized weights for MI and SD in the frequency domain with **F** matrix

(b) Normalized weights for MI and SD in the frequency domain with **G** matrix

(c) MI with **F** matrix

(d) MI with **G** matrix

(e) SD with **F** matrix

(f) SD with **G** matrix

Fig. 8. Normalized weights variations of recording segments in the frequency domain by using MI (blue dotted line) and SD (red dotted line) methods. Insets (a) and (b) shows inflection points on the normalized weights curve around 225 and 450 features when using the **F** and **G** matrices, respectively; insets (c) and (e) show a well defined frequency band from 2 to 7 Hz when using **F** matrix; insets (d) and (f) show that the same frequency band, but with less resolution, is also present when considering **G** matrix. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Fig. 6. In Figs. 6(a) and (b), we can see a detection example of 5 events, where we can determine a detection rate of 5/5 and 3/5 for each w, respectively. Figs. 6(c) and (d) are the zoomed portion from 3000 s to 5000 s, and we can observe that the event was better followed with $w = 15$ s compared to $w = 40$ s. With $w = 40$ s, the event seemed to move in the segmentation window, and this effect made the detector cannot identify the events, as shown in Fig. 6(e) and (f).

5.2. Results using k-NN classifier

We defined a minimum value of k following an empirical rule given in Duda et al. (2012), which states that the value of k must be equal to the squared root of n features. For our case, the minimum value of k was 11 when considering 1500 features of **S** matrix. Once the minimum value was defined, we selected a maximum k of 400 to have a wide enough observation range.

The training set is used to determine the optimal values of k for each matrix. Table 2 shows the results of detection based on classification when considering the validation and the test sets. We observed better results when using **T**, **F**, and **G**, than for the use of **S** and **W** matrices. We observed also that **S** and **W** matrices needed 11 nearest neighbors, whilst the rest of matrices used k above 200 nearest neighbors. The best results were obtained in the frequency domain, and the difference between **F** and **G** matrices was negligible in terms of A, P, and S, whilst for R the differences were around 6%.

We used the feature selection strategy in order to identify the most relevant p features that improved the processing time and its performance. We worked with Mutual Information (MI) and Statistical Dependence (SD) (Pohjalainen et al., 2015), which are statistical methods used

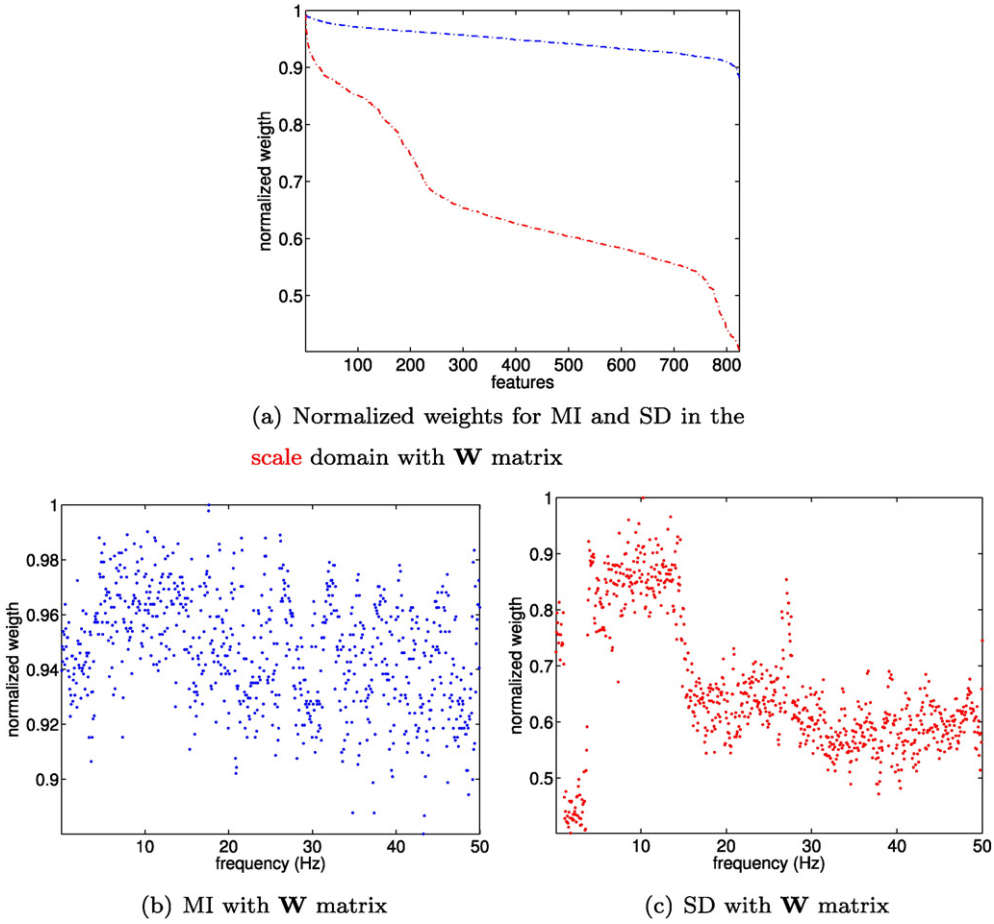


Fig. 9. Normalized weights variations of recording segments in the scale domain by using MI (blue dotted line) and SD (red dotted line) methods. Inset shows inflection points on the normalized weights around 200 and 600 features when using **W** matrix; inset (b) shows the normalized weight variation according to the feature relevance in the scale domain; inset (c) shows a well formed frequency band from 7 to 14 Hz when using **W** matrix. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

as filters in feature selection stage, which quantify the dependency between the class variable (Y) and the subset of selected features (X). Both methods are similarly defined, the difference being related to the way of evaluating this dependency. Hence,

$$\mathrm{SD} = \sum_{y\in Y}\sum_{z\in Z} p(y,z)\frac{p(y,z)}{p(y)p(z)} \tag{14}$$

$$\mathrm{MI} = \sum_{y\in Y}\sum_{z\in Z} p(y,z)\log\frac{p(y,z)}{p(y)p(z)}. \tag{15}$$

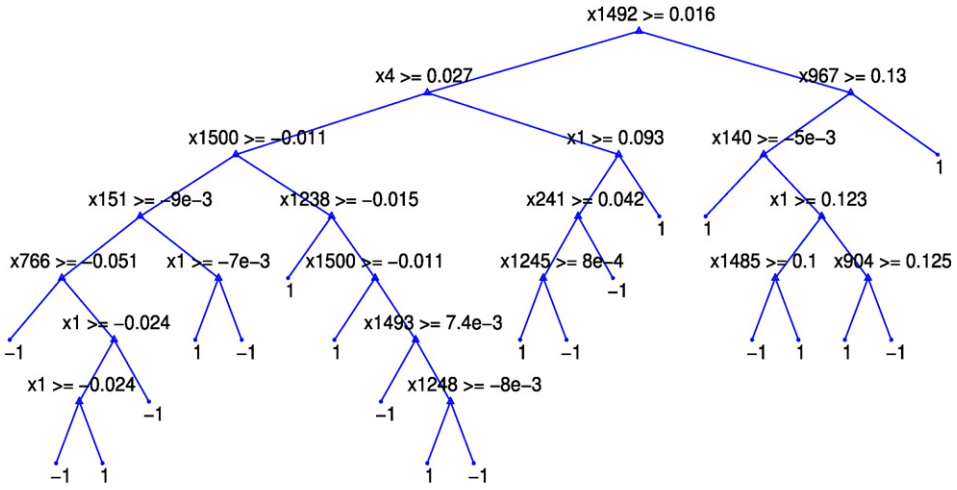
Accordingly, the larger the SD or MI, which are referred to as weights, the higher the dependency between the feature values and the class labels. Note that SD is more sensitive, due to the absence of logarithmic compression of MI.

Table 3
Experimental performance results for detection when applying a feature selection stage with k-NN classifier, $w = 15$ s, and validation/test sets, by considering the relevant features and the feature selection method.

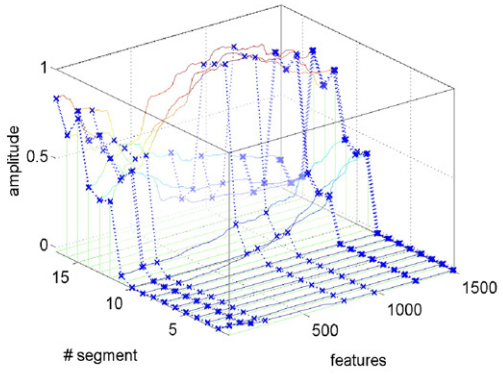
Matrix	n features–method	k	A (%)	P (%)	R (%)	S (%)
S	600–SD	11	99/99	100/100	25/31	100/100
T	200–SD	351	99/99	100/100	77/74	100/100
T	600–SD	351	99/99	100/100	77/74	100/100
F	50–MI	241	99/99	100/100	77/74	100/100
G	25–MI	271	99/99	100/100	77/71	100/100
W	200–SD	11	99/99	100/100	44/49	100/100
W	600–SD	11	99/99	100/100	49/53	100/100

Fig. 7 shows the normalized weights for MI and SD in the time domain for the **S** and **T** matrices. Figs. 7(a) and (b) are related to the normalized weights of the features. As it can be seen in Fig. 7(a), features have a very similar and continuous smooth changing behavior, for both MI and SD curves, in the time domain for **S** matrix, whilst for **T** matrix in Fig. 7(b) there are some small inflection points on the SD curve around 200 and 600 features. These changes in smoothness indicate that the relationships between the desired class label and the distribution of the subset containing such number of features change significantly after these points. Therefore, subsets with these features may be enough to containing the most relevant information related to the class, and hence these subsets should be used for testing. Fig. 7(c), (d), (e) and (f) shows the normalized weights according to feature relevance in the time domain, however, from these plots, it was not possible to determine which were the most relevant features for reducing the processing time.

Similarly, Fig. 8 shows the results in the frequency domain for high and moderate resolution matrices **F** and **G**, respectively. From Figs. 8(a) and (b), it is possible to distinguish a dramatic change in the traces around 450 and 225 features as the most significant when considering SD method. However, in terms of frequency, as it can be seen in Figs. 8(c), (d), (e) and (f), it is possible to identify significant changes in the normalized weights produced in the frequency band $f_b \in (2, 7)$ Hz, although these changes are produced for both SD and MI, the spectral components in MI have greater normalized weights values, therefore MI allows better discernment of the main features in frequency domain. This is consistent with the accepted spectral band for LP



(a) Main features retrieved by DT algorithm



(b) Features verified in each $\mathbf{s}_{i,j}$

Fig. 10. Results obtained by the DT classifier when considering **T** matrix. Inset (a) shows the 20 key features selected by DT algorithm given a total of 21 possible leafs (LP = 1 and background noise = −1); inset (b) shows that several points can be used in each segment to define it as LP or background noise.

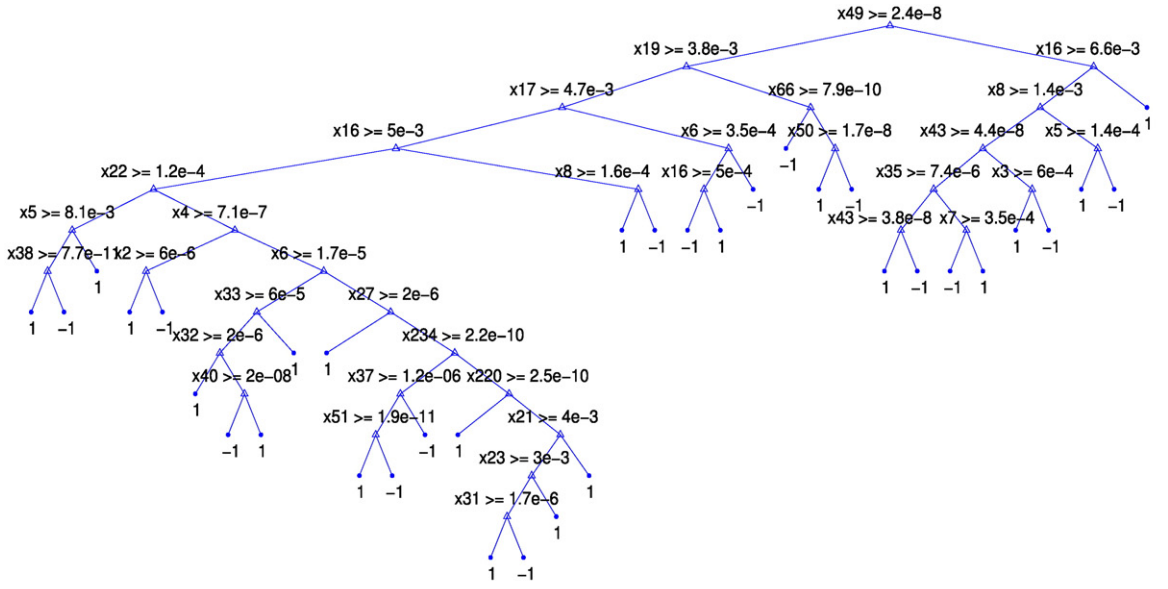
events. For the **F** matrix, the number of points corresponding to the normalized weights within the frequency band 2 to 7 Hz is 50, as illustrated in Fig. 8(c), whilst for the **G** matrix the number of points in this band is only 25, as it can be observed in Fig. 8(d). Therefore, we decided to use subsets containing these 50 features from the **F** matrix and 25 features from the **G** matrix for testing.

Finally, Fig. 9(a) shows the variation of normalized weights in the scale domain for **W** matrix. Similarly, to the analysis of Fig. 7, there are also some small inflection points on the SD curve around 200 and 600 features, therefore, and following the same logic as in the case of Fig. 7, subsets containing these number of features may be enough to containing the most relevant information related to the class, and hence subsets with these number of features should be selected for testing. On the other hand, when considering MI, features are strongly correlated among them, and therefore it is not possible to identify any significant changes in the normalized weights produced in the entire frequency spectrum, as illustrated on Fig. 9(b). Nevertheless, in Fig. 9(c) for SD it is possible to identify a region containing 150 features within the frequency band $f_w \in (7, 14)$ Hz where significant changes in the normalized weights can be observed. Hence a subset with these 150 features was chose for testing. However, this is not consistent with the spectral band accepted to contain the main information for LP events, therefore, we also tested subset containing the previously selected 200 and 600 features, in order to identify possible differences between the selected subsets of features.

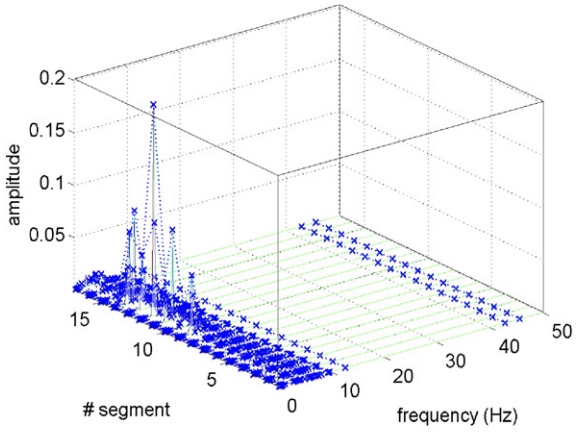
Table 3 summarizes the results of applying the feature selection stage. We obtained a noticeable improvement in most of the performance metrics when reducing the input space by using feature selection, which can be compared by analyzing the differences between Tables 2 and 3. For **S** matrix, the improvement was on all terms, with a remarkable improvement for *R* of 19%, meanwhile for **T** this improvement was better for *R* about 55%, by using 200 or 600 features. However, for **F** and **G**, the value of *R* got worse (around 24% and 21% respectively), even when for *P* it was improved in about 8%. Finally, for **W**, *P* and *R* were improved in 40% and 12%, respectively.

5.3. Results using DT classifier

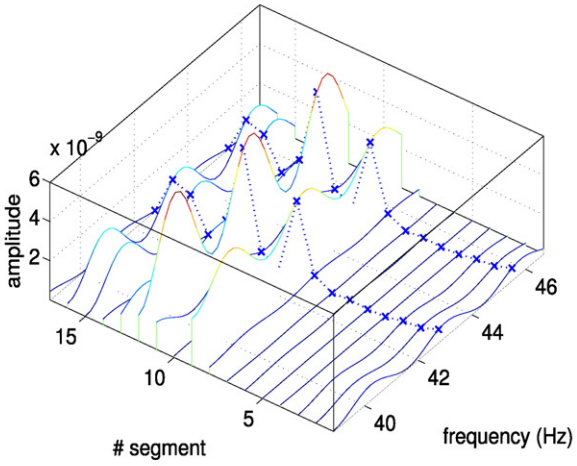
At this point, we decided to work with **T** and **G** matrices because both matrices provided the best results for the *k*-NN classifier. The goal of the DT algorithm is to create a model that predicts the value of a target variable based on several input variables. Fig. 10(a), shows an example of the top-down classification tree found by the DT algorithm for **T** matrix, in which each internal node is labeled with an input feature and each branch in the tree represent a decision rule, there are edges to children for each of the possible values of the feature. Each leaf of the tree is labeled with a class or a probability distribution over the possible classes. The algorithm learns the "best" possible tree by splitting the source set into subsets based on test attributes. This process is repeated on each derived subset in a recursive manner called recursive



(a) Main features retrieved by DT algorithm



(b) Features verified in each $\mathbf{g}_{i,j}$



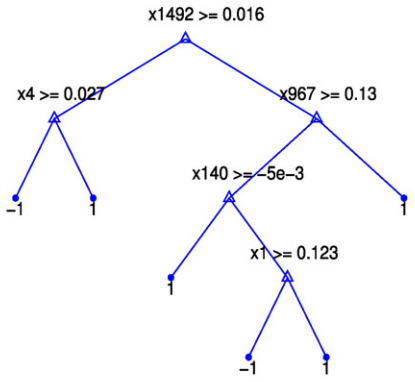
(c) Spectral content in frequency band (40, 45) Hz

Fig. 11. Results obtained by the DT classifier when considering **G** matrix: inset (a) shows the 34 key features selected by DT algorithm given a total of 35 possible leafs (LP = 1 and background noise = −1); inset (b) shows that several points can be used in each segment to define it as LP or background noise; inset (c) shows that 2 frequencies peaks around 43 Hz and 46 Hz can also be used for identification of the event.

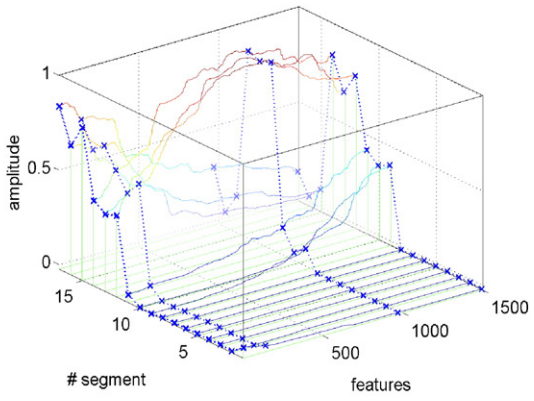
partitioning. The recursion is completed when the subset at a node has all the same value of the desired target variable, or when splitting no longer adds value to the predictions. To determine the performance of the tree, a measurement of the homogeneity of the target variable within the subsets, called Gini impurity Index, is used. The Gini impurity is a measure of the statistical dispersion of wrong classifications according to the distribution of labels in the subset. Refer to Bishop et al. (2006) for more details on the algorithm.

In the case of Fig. 10(a), for **T** matrix, the DT algorithm found that the main input features corresponding to the amplitude value at 1492 s, 4 s, and 967 s. Each of the chosen features were those that algorithm found have more information and were the most relevant for the classification. Similarly, the decision rules for each branch in the tree, for example the threshold value of 0.016 for feature amplitude in rule X_{1492}, were also found experimentally by the algorithm after several automatic optimizations based on the data information. As a result, the DT algorithm selected 20 key features, beginning from the top node with the rule $X_{1492} \geq 0.016$, making possible to classify any event into any one of the 21 possible classes we applied.

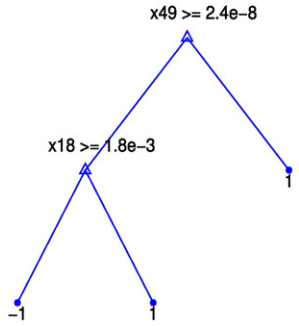
We considered 4 events, and most of these events consisted of 3 $\mathbf{s}_{i,j}$ with $w = 15$ s, giving a total of 12 segments. Fig. 10(b) shows a mesh representation of 18 segments: segments from 1 to 6 are examples of background noise, which were not part of the target events; segments from 7 to 10 correspond to the beginning of the events; segments from 11 to 14 correspond to the main part of the events; and segments from 15 to 18 contain the final part of the events. We observe that the DT algorithm retrieved about 20 amplitude features in each segment in order to predict the outcome. It is evident that the features that are being selected correspond to amplitude comparisons in different relative timings of the segmented time window, which is consistent with the usual increase in amplitude that can be observed in the envelope of a typical LP event.



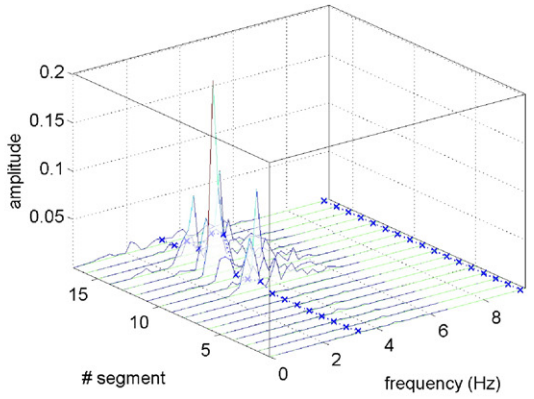
(a) Features retrieved with cross-validation and prunning



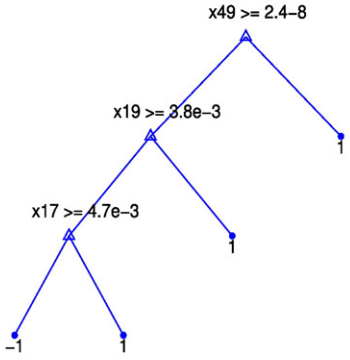
(b) Features verified in each $\mathbf{s}_{i,j}$



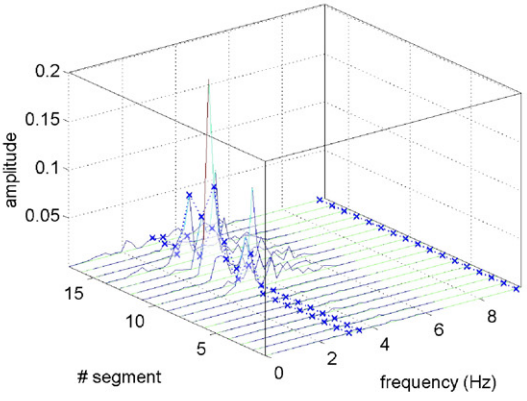
(c) Features retrieved with cross-validation



(d) Features verified in each $\mathbf{g}_{i,j}$



(e) Features retrieved with prunning



(f) Features verified in each $\mathbf{g}_{i,j}$

Fig. 12. Tree representation and main features retrieved by considering feature selection methods: inset (a) shows the 5 key features selected by the feature selection methods given a total of 6 possible leafs; inset (b) shows the equivalent 5 points corresponding to the features that are verified for each segment; inset (c) shows the 2 key features selected by cross-validation method given a total of 3 possible leafs; inset (d) shows the equivalent 2 frequency features that verified at 9.4 Hz and 3.3 Hz in each segment; inset (e) shows 3 key features selected by pruning method given a total of 4 possible leafs; inset (d) shows the equivalent 3 frequencies features that are verified at 9.4 Hz, 3.5 Hz, and 3.1 Hz in each segment.

Meanwhile for the **G** matrix, Fig. 11(a) shows the tree given by the DT algorithm, which selected 34 key features, beginning from the top node, with rule $X_{49} \geq 2.4e-8$, and classifying into one of the 34 features. We followed the same methodology used for the time domain to represent segments in mesh form, for instance, in Fig. 11(b) we considered 4 events, which were represented by 12 segments related to the events and 6 additional segments corresponding to background noise. We observed that the DT algorithm was able to verify 30 features in each $\mathbf{g}_{i,j}$ in order to predict the outcome. An interesting observation is that in this case, this representation shows that not only the band in the range (2,

Table 4
Experimental performance results for detection by applying feature selection stage with DT classifier, $w = 15$ s, and validation/test sets, by considering the relevant features and feature selection method.

Matrix	N. features–method	*A* (%)	*P* (%)	*R* (%)	*S* (%)
T	20–Default	99/99	100/100	95/95	100/100
T	5–Cross-validation	99/99	100/100	90/86	100/100
T	5–Pruning	99/99	100/100	90/88	100/100
G	34–Default	99/99	100/100	99/99	99/100
G	2–Cross-validation	99/99	100/100	94/94	100/100
G	3–Pruning	99/99	100/100	92/94	100/100

14) Hz is the relevant one since the classifier also found information related to the event contained in the high frequency band (40, 45) Hz, as illustrated in Fig. 11c. Although the source of this phenomenon is still unclear at the moment and requires further investigation, this information can also be used to differentiate real events from background noise, since the spectral content of the segments containing the event is greater than the spectral content of the noise.

5.3.1. Reduced number of features

Pruning and cross-validation are used as embedded methods (Esposito et al., 1997; Dai, 2013) in feature selection stage, which are in charge to control the *leafiness* of decision trees by removing the unnecessary leaf nodes (features) to classify the instances. These methods reduce the complexity of the tree, with its corresponding improvement in predictive accuracy. For Pruning method, trees are based on an optimal pruning scheme that first prunes branches giving less improvement in error cost. Meanwhile, cross-validation method consists of partitioning a dataset into *n* subsets and then running decision tree algorithm *n* times, each time using a different training set and validating the results on each subset, and corroborated with the improvement in error cost.

We applied pruning and cross-validation methods to **T** matrix in the time domain, in order to control leafiness in DT. The DT algorithm retrieved 5 key features with both methods, beginning from the top node, with rule $X_{1492} \geq 0.016$, which allowed classification into one of the 6 possible leafs, as depicted in Fig. 12(a).

Similarly, in the frequency domain, by considering the **G** matrix and by using cross-validation, the DT algorithm selected only 2 key features, beginning from the top node, with rule $X_{49} \geq 2.4e-8$ (this feature corresponding to the amplitude value at 9.4 Hz), and $X_{18} \geq 2.4e-8$ (corresponding to the amplitude value at 3.3 Hz), and allowed classification into one of the 3 possible leafs, as depicted in Fig. 12(c). Meanwhile, when using the pruning method, the DT algorithm selected 3 key features, the top node was kept with the same rule, and two more rules were defined, $X_{29} \geq 3.8e-3$ and $X_{17} \geq 4.7e-3$, which corresponded to the amplitude value at 3.5 Hz and 3.1 Hz, respectively, and the classification was into one of the 4 possible leafs, as depicted in Fig. 12(f).

Table 4 shows the results obtained with the validation set, which were confirmed with the test set. We obtained the best results with original features by using **G**, however, good performance was obtained by using methods to control depth (feature selection-embedded), just considering 5, 3, or 2 key features, depending of the case.

In order to validate the preprocessing block, we evaluated the performance of the proposed system with *k*-NN classifier, and only using one of the considerations in the preprocessing block at the time (either using filter or normalization). The results, in each case, got worse performance than when all the considerations in the preprocessing block were used, the results for *P* value were in the order of 10% lower for both classification and detection stages.

6. Discussion and conclusions

Previous works have demonstrated the success and usefulness of machine learning techniques applied to the problem of classifying events from a volcano, without considering real-time requirements (Falsaperla et al., 1996; Langer et al., 2003). Most reports do not fully detail their methodology, which makes hard to identify the main parameters to be considered for a real-time strategy (Langer et al., 2006; Curilem et al., 2009; Messina and Langer, 2011; Esposito et al., 2008; Ohrnberger, 2001; Ruano et al., 2014).

In our case, several possible features were extracted from the seismological signal in order to detect and classify events. In this regard, in terms of temporal features, Fig. 10(b) shows that the most relevant features are those located at the beginning, middle and end of each segment, therefore a subsampling amplitude detector in the observed window may be enough for event detection, according to the main featured retrieved by the DT algorithm. Whereas in the case of the spectral features, as illustrated by the Fourier representation from Fig. 11(b), the most relevant features are concentrated within the 4 to 10 Hz frequency baseband, which agrees with the frequency bands mentioned in previous studies, however it is also possible to observe some relevant information related to the event around the 40 Hz, which has not been noted in previous studies and therefore require a further analysis. Observing Fig. 12(b), it is clear that the amplitude of features is more or less uniformly distributed for the sampling windows in the band between 4 and 10 Hz, therefore using a sample amplitude detector in such frequency band may be sufficient for event detection.

LP events and VT earthquakes have proven to be key elements for monitoring any volcano, including Cotopaxi, since they provide important information about the volcano status. With an appropriate monitoring and detection, a real-time system will allow launching an effectiveness early warning. The dataset provided by IGEPN contains mostly LP events from Cotopaxi Volcano. At this stage we were interested in develop a detection system based on classification, and for that purpose, we have to define a characterization of the most prevalent events, in this case, LP events. However, in future works we are interested in extending the current scope towards an automatic recognition system for LP and VT. However we will require a data set containing enough VT events for training the algorithms when developing such system.

Our proposed approach detects LP events with high accuracy and reduced computational requirements, and it is developed towards its use in real-time analysis, rather than data in blocks or off-line based approaches. We considered LP event detection based on classification, and in this case, the use of detection criteria instead of classification criteria gave significant advantage in the free parameters tuning. We worked with supervised classification, since this is a highly user-defined and application driven solid approach.

Our experiments have shown that the best results can be obtained in the frequency domain, by using **F** or **G** matrices, instead of in the time and scale domains, and by using DT classifier instead of *k*-NN classifier. The results also show that the system has reached an accuracy of 99% in the detection stage. Regarding the scheme for the event detector based on classification proposed in Fig. 3, the feature selection block, by using filter methods, yielded a significant improvement with *k*-NN classifier, in terms of *P* at least in 10%, whilst for *R*, without feature selection block, it presented different values for the validation and test sets (33% and 92%, respectively), a difference of around 60%. Meanwhile, by using this block for validation and test sets, detection reached an *R* below 77%, with a noticeable reduction of features from 513 to 50 for **F** matrix, and from 257 to 25 for **G** matrix, mainly due to the identification of a frequency band $f_b \in (2,7)$ Hz, which contains the most relevant features. Moreover, embedded methods got worse performance with DT classifier, in terms of *R* in 5%, the rest of the parameters resulted similar by considering **G** matrix, and a considerable reduction of features was achieved, resulting in 5 and 2 main features in the time and frequency domain, respectively. If the frequency content of LP events was to change over time, it still would be possible to distinguish LP from background noise, since the spectral content of the noise has a small amplitude compared to the LP spectral content. This feature

selection strategy has permitted reducing the processing time from 3 min to 2 min, approximately.

As future work, we are interested in developing a new strategy by extracting features of each $\mathbf{s}_{i,j}$ in the time, frequency, and scale domain, in order to identify the main features for reducing the processing time avoiding the overfitting, we also plan to include in our experiments Support Vector Machine (SVM) (Schölkopf et al., 2000; Chang and Lin, 2001), to allow discrimination between LP events from VT earthquakes. We want to improve the classification rate by using Digital Communication and Codification Theory (Zhang, 2011), specifically, turbo codification techniques for error control in the classifiers.

Acknowledgment

The authors gratefully acknowledge the contribution of Universidad de las Fuerzas Armadas ESPE for the economical support in the development of this project by Research Grants 2013-PIT-014 and 2015-PIT-004. This work has been partly supported by Research Projects S203/ICE-2933 (Comunidad Autónoma de Madrid), TEC2013-48439-C4-1-R (Spanish Government), and by the Prometeo Project of the Secretariat for Higher Education, Science, Technology and Innovation of the Republic of Ecuador.

References

Álvarez, I., García, L., Cortés, G., Benítez, C., De la Torre, A., 2012. Discriminative feature selection for automatic classification of volcano-seismic signals. IEEE Geosci. Remote Sens. Lett. 9 (2), 151–155.
Baubron, J.-C., Allard, P., Sabroux, J.-C., Tedesco, D., Toutaln, J.-P., 1991. Soil gas emanations as precursory indicators of volcanic eruptions. J. Geol. Soc. 148 (3), 571–576.
Bean, C.J., De Barros, L., Lokmer, I., Métaxian, J.-P., O'Brien, G., Murphy, S., 2014. Long-period seismicity in the shallow volcanic edifice formed from slow-rupture earthquakes. Nat. Geosci. 7 (1), 71–75.
Behnke, S.A., Thomas, R.J., McNutt, S.R., Schneider, D.J., Krehbiel, P.R., Rison, W., Edens, H.E., 2013. Observations of volcanic lightning during the 2009 eruption of Redoubt Volcano. J. Volcanol. Geotherm. Res. 259, 214–234.
Bishop, C.M., et al., 2006. Pattern Recognition and Machine Learning. vol. 1. Springer, pp. 663–666.
Bonaccorso, A., Bonforte, A., Guglielmino, F., Palano, M., Puglisi, G., 2006. Composite ground deformation pattern forerunning the 2004–2005 Mount Etna eruption. J. Geophys. Res. 111 (B12).
Cárdenas-Peña, D., Orozco-Alzate, M., Castellanos-Dominguez, G., 2013. Selection of time-variant features for earthquake classification at the Nevado-del-Ruiz Volcano. J. Comput. Geosci. 51, 293–304.
Chang, C.-C., Lin, C.-J., 2001. Training ν-support vector classifiers: theory and algorithms. Neural Comput. 13 (9), 2119–2147.
Chen, H., Chen, S., 2003. A moving average based filtering system with its application to real-time QRS detection. Computers in Cardiology, pp. 585–588.
Chouet, B.A., 1996. Long-period volcano seismicity: its source and use in eruption forecasting. Nature 380 (6572), 309–316.
Chouet, B.A., Matoza, R.S., 2013. A multi-decadal view of seismic methods for detecting precursors of magma movement and eruption. J. Volcanol. Geotherm. Res. 252, 108–175.
Córdova Regalado, E.A., 2013. Estudio de Micro-Sismicidad Para Los Proyectos Geotérmicos: Chacana y Chachimbiro (Master's thesis) Escuela Politécnica Nacional.
Cortés, G., García, L., Álvarez, I., Benítez, C., de la Torre, Á., Ibáñez, J., 2014. Parallel system architecture (PSA): an efficient approach for automatic recognition of volcano-seismic events. J. Volcanol. Geotherm. Res. 271 (0), 1–10.
Curilem, G., Vergara, J., Fuentealba, G., Acuña, G., Chacón, M., 2009. Classification of seismic signals at Villarrica Volcano (Chile) using neural networks and genetic algorithms. J. Volcanol. Geotherm. Res. 180 (1), 1–8.
Cusano, P., Palo, M., West, M., 2015. Long-period seismicity at Shishaldin Volcano (Alaska) in 2003–2004: indications of an upward migration of the source before a minor eruption. J. Volcanol. Geotherm. Res. 291, 14–24.
Dai, Q., 2013. A competitive ensemble pruning approach based on cross-validation technique. Knowl.-Based Syst. 37, 394–414.
Duda, R.O., Hart, P.E., Stork, D.G., 2012. Pattern Classification. John Wiley & Sons, pp. 174–177.
Dvorak, J.J., Dzurisin, D., 1997. Volcano geodesy: the search for magma reservoirs and the formation of eruptive vents. Rev. Geophys. 35 (3), 343–384.
Dzurisin, D., 1980. Electronic tiltmeters for volcano monitoring: lessons from Mount St. Helens. J. Monit. Volcanoes 90, 69–83.
Esposito, F., Malerba, D., Semeraro, G., Kay, J., 1997. A comparative analysis of methods for pruning decision trees. IEEE Trans. Pattern Anal. Mach. Intell. 19 (5), 476–491.
Esposito, A., Giudicepietro, F., D'Auria, L., Scarpetta, S., Martini, M., Coltelli, M., Marinaro, M., 2008. Unsupervised neural analysis of very-long-period events at Stromboli Volcano using the self-organizing maps. Bull. Seismol. Soc. Am. 98 (5), 2449–2459.
Falsaperla, S., Graziani, S., Nunnari, G., Spampinato, S., 1996. Automatic classification of volcanic earthquakes by using multi-layered neural networks. J. Nat. Hazards 13 (3), 205–228.
Galle, B., Oppenheimer, C., Geyer, A., McGonigle, A.J., Edmonds, M., Horrocks, L., 2003. A miniaturised ultraviolet spectrometer for remote sensing of SO2 fluxes: a new tool for volcano surveillance. J. Volcanol. Geotherm. Res. 119 (1), 241–254.
Hall, M., Mothes, P., 2008. The rhyolitic–andesitic eruptive history of Cotopaxi Volcano, Ecuador. Bull. Volcanol. 70 (6), 675–702.
Ibáñez, J.M., Benítez, C., Gutiérrez, L.A., Cortés, G., García-Yeguas, A., Alguacil, G., 2009. The classification of seismo-volcanic signals using hidden Markov models as applied to the Stromboli and Etna Volcanoes. J. Volcanol. Geotherm. Res. 187 (3), 218–226.
Kenneth, B., 2001. The Seismic Wavefield, Volume I: Introduction and Theoretical Development. Cambridge Univ. Press, USA.
Langer, H., Falsaperla, S., Thompson, G., 2003. Application of artificial neural networks for the classification of the seismic transients at Soufriere Hills Volcano, Montserrat. Geophys. Res. Lett. 30 (21), 1–6.
Langer, H., Falsaperla, S., Powell, T., Thompson, G., 2006. Automatic classification and a-posteriori analysis of seismic event identification at Soufriere Hills Volcano, Montserrat. J. Volcanol. Geotherm. Res. 153 (1), 1–10.
Lewicki, J., Connor, C., St-Amand, K., Stix, J., Spinner, W., 2003. Self-potential, soil CO2 flux, and temperature on Masaya Volcano, Nicaragua. Geophys. Res. Lett. 30 (15), 1–6.
Lyons, J.J., Haney, M.M., Fee, D., Paskievitch, J.F., 2014. Distinguishing high surf from volcanic long-period earthquakes. Geophys. Res. Lett. 41 (4), 1171–1178.
McNutt, S.R., 1996. Seismic Monitoring and Eruption Forecasting of Volcanoes: A Review of the State-of-the-art and Case Histories. Springer, pp. 99–146.
McNutt, S.R., 2000. In: Sigurdsson, H., Houghton, B., SR, McNutt, Rymer, H., Stix, J. (Eds.), Seismic Monitoring, Encyclopedia of Volcanoes, pp. 1–95.
Mery, D., Medina, O., 2004. Automated visual inspection of glass bottles using adapted median filtering. J. Image Anal. Recognit. 818–825.
Messina, A., Langer, H., 2011. Pattern recognition of volcanic tremor data on Mt. Etna (Italy) with KKAnalysis—a software program for unsupervised classification. J. Comput. Geosci. 37 (7), 953–961.
Mitchell, T.M., 1997. Machine Learning. vol. 45. McGraw Hill, Burr Ridge, IL, pp. 230–236.
Newman, T.S., Jain, A.K., 1995. A survey of automated visual inspection. J. Comput. Vis. Image Underst. 61 (2), 231–262.
Ngui, W.K., Leong, M.S., Hee, L.M., Abdelrhman, A.M., 2013. Wavelet analysis: mother wavelet selection methods. Appl. Mech. Mater. 393, 953–958.
Ohrnberger, M., 2001. Continuous Automatic Classification of Seismic Signals of Volcanic Origin at Mt. Merapi, Java, Indonesia (Ph.D. thesis) University of Potsdam.
Ortiz Erazo, H.D., 2013. Estudio de Los Efectos de Sitio Para La Construcción de Un índice de Actividad sísmica en El volcán Cotopaxi (Master's thesis) Escuela Politécnica Nacional.
Papadimitriou, P., Kapetanidis, V., Karakonstantis, A., Kaviris, G., Voulgaris, N., Makropoulos, K., 2015. The Santorini Volcanic complex: a detailed multi-parameter seismological approach with emphasis on the 2011–2012 unrest period. J. Geodyn. 85, 32–57.
Pohjalainen, J., Räsänen, O., Kadioglu, S., 2015. Feature selection methods and their combinations in high-dimensional classification of speaker likability, intelligibility and personality traits. Comput. Speech Lang. 29 (1), 145–171.
Ruano, A., Madureira, G., Barros, O., Khosravani, H., Ruano, M., Ferreira, P., 2014. Seismic detection using support vector machines. J. Neurocomputing 135 (0), 273–283.
Scarpetta, S., Giudicepietro, F., Ezin, E., Petrosino, S., Del Pezzo, E., Martini, M., Marinaro, M., 2005. Automatic classification of seismic signals at Mt. Vesuvius Volcano, Italy, using neural networks. Bull. Seismol. Soc. Am. 95 (1), 185–196.
Schölkopf, B., Smola, A.J., Williamson, R.C., Bartlett, P.L., 2000. New support vector algorithms. Neural Comput. 12 (5), 1207–1245.
Sicali, S., Barberi, G., Cocina, O., Musumeci, C., Patanè, D., 2015. Volcanic unrest leading to the July–August 2001 lateral eruption at Mt. Etna: seismological constraints. J. Volcanol. Geotherm. Res. 304, 11–23.
Sparks, R., 2003. Forecasting volcanic eruptions. Earth Planet. Sci. Lett. 210 (1), 1–15.
Syahbana, D.K., Caudron, C., Jousset, P., Lecocq, T., Camelbeeck, T., Bernard, A., et al., 2014. Fluid dynamics inside a wet volcano inferred from the complex frequencies of long-period (LP) events: an example from Papandayan Volcano, West Java, Indonesia, during the 2011 seismic unrest. J. Volcanol. Geotherm. Res. 280, 76–89.
Trombley, R., Toutain, J.-P., 2005. Eruption pro 10.5—the new and improved long-range eruption forecasting software. Caribb. J. Earth Sci. 39, 3–8.
Voight, B., Hoblitt, R., Clarke, A., Lockhart, A., Miller, A., Lynch, L., McMahon, J., 1998. Remarkable cyclic ground deformation monitored in real-time on Montserrat, and its use in eruption forecasting. Geophys. Res. Lett. 25 (18), 3405–3408.
Withers, M., Aster, R., Young, C., Beiriger, J., Harris, M., Moore, S., Trujillo, J., 1998. A comparison of select trigger algorithms for automated global seismic phase and event detection. Bull. Seismol. Soc. Am. 88 (1), 95–106.
Zhang, Z., 2011. Theory and applications of network error correction coding. Proc. IEEE 99 (3), 406–420.