A Machine Learning Toolbox for Automating Development of Personalized Epileptic Seizure Detection Algorithms

G. Saulnier-Comte, J. Pineau, M. Lévesque, M. Avoli

School of Computer Science, McGill University, Montréal, Québec, Canada
Montreal Neurological Institute, McGill University, Montréal, Québec, Canada

Abstract

Objective: A toolbox for automating the development of real-time personalized epileptic seizure detection algorithms is presented. The toolbox contains modules that cover feature extraction, feature selection, classifier training and performance evaluation using cross-validation.

Methods: A large pool of features is extracted from the training dataset of a given patient using different signal processing methods. Then, the feature selection modules picks an efficient subset of features. Next, a high-level machine learning classifier is created to automatically classify EEG data as seizure or non-seizure.

Results: The toolbox performance was evaluated using 3-fold cross-validation on multiple patients from three publicly available datasets. The overall sensitivities were between 74.2% and 92.7% with median false positive rates below 2 per day.

Conclusion: The toolbox was able to create individualized detection algorithms with suitable sensitivities and low false positive rates for most of the patients.

Significance: The performance of the toolbox confirms its potential to be used in clinical settings, raising alarms when patients suffer from seizures. Moreover, it can pre-process EEG recordings by finding seizure occurrences. The modularity of the toolbox enables its components to be used in the design of new algorithms tailored for different tasks.

Keywords: Epilepsy, EEG, Seizure Detection, Machine Learning

1. Introduction

Epilepsy is a chronic neurological disorder affecting 1% of the world population. It is characterized by recurrent seizures that are refractory to anti-epileptic drugs in 20% to 30% of patients (Loscher, 1997). The diagnosis of epilepsy is based on the analysis of the electroencephalographic (EEG) signal, because seizures are accompanied by hypersynchronous electrical discharges that can be recorded using scalp or depth electrodes. Seizures are unpredictable events; therefore, one of the main challenges in the diagnosis and the treatment of this disorder rests on detecting and predicting their occurrence. This problem may be solved by performing computational analysis of the EEG signal.

Automated seizure detection is indeed the process of identifying and reporting the occurrence of ictal (seizure) events in the EEG with an online algorithm (running in real time). Creating such algorithms is often a complex procedure as it requires multiple components to be well designed in order to achieve good performance. This complexity comes from multiple factors.

First, epileptic disorders are not well understood: the mechanisms underlying seizure generation are mostly unknown. This is partly caused by the heterogeneity of the disorder and interindividual variability regarding the symptoms, EEG activity and neuropathological changes.

Second, there are technical difficulties associated with EEG recordings. The recordings are made from small electrodes that capture a noisy state of the brain activity at a given location. The origin of the seizures being unknown for many patients, the electrode placement may not be optimal, yielding weaker electrical signals. Therefore, the automated seizure detection system needs to make abstraction of the noise and filter out irrelevant information with respect to the type of seizure and its manifestations.

Finally, only a small part of the EEG recordings are reviewed by epileptologists in order to annotate ictal events. This task is time consuming and sometimes requires an analysis of behavioral symptoms in order to confirm the occurrence of an ictal event. The relatively low rate of occurrence of seizures and their short duration, compared to the amount of inter-ictal (between seizures) data, makes the EEG recordings highly unbalanced.

An automated seizure detection system must take into account all these challenges in order to be efficient. Multiple algorithms have been designed in the past few years. As an example, Shoeb & Guttag (2010) designed a new type of feature vector that accounted for both spectral and spatial information of a patient’s disorder characteristics and used a support vector machine (SVM) with a non-linear kernel to detect ictal events. Chan et al. (2008) used spectral and temporal features com-
bined with a SVM in order to localize seizure onset times. Saab & Gotman (2005) used wavelet decomposition combined with a Bayesian model for seizure detection. Zandi et al. (2010) implemented a new feature called the combined seizure index and monitored its increase using a robust statistic in order to raise alarms. The features used in those methods are either new designs or hand selected by the authors. The feature parametrizations are sometimes automated on a per patient basis or chosen according to prior knowledge.

We introduce a toolbox designed to ease and automate the various phases of building a personalized epileptic seizure detection algorithm. It contains a large pool of features and parametrizations, and a feature selection module that tailors the algorithm to a specific patient. A high-level machine learning classifier is used for the detection of ictal events. We also introduce a new dataset containing EEG recordings from rats suffering from epilepsy. Finally, we use a sound methodology to evaluate the performance of the toolbox on a total of 3 different public datasets.

The existence of efficient seizure detection algorithms could help tackle more complicated problems such as prediction, prevention and early abortion. It could also increase the quality of life of patients suffering from epilepsy by alerting nurses or relevant authorities that the patient is currently suffering from a seizure. It could also help alleviate the analysis of EEG recordings for epileptologists by indicating points of interests.

2. Methods

The toolbox was created to facilitate the process of designing an epileptic seizure detection algorithm, such that it becomes feasible to build personalized seizure detectors with minimal need for design choices. The toolbox applies equally well to intracranial and scalp EEG signals. The main requirements of the system are a sample of EEG data from the patient, containing a small number (e.g. 3) of hand labelled seizures.

The main components of a seizure detection algorithm include: feature extraction from the multichannel EEG signal, personalized feature selection, robust event classification, and performance analysis. If desired, the toolbox offers multiple design options for each phase. In terms of feature extraction, our toolbox offers many of the univariate and bivariate features identified in the literature (Mormann et al. (2007), and references therein). The feature selection is achieved through regularized logistic regression, which provides the ability to select a small number of robust features from a minimal amount of pre-recorded training data. Classification of seizure events can be performed directly within the toolbox using the regularized logistic regression or, for a richer hypothesis class, extremely randomized trees (Extra-Trees). The classification can also be performed by other machine learning toolboxes, such as Weka (Hall et al., 2009), by exporting the feature data. Performance of the system is characterized by the sensitivity of the classifier, false positive rate, and detection latency; the toolbox provides tools for calculating these metrics, and indications for tuning the parameters to achieve different trade-offs between sensitivity/accuracy and false positive rate. A flow chart describing the use of the toolbox is depicted in Figure 1.

2.1. Data Description

EEG recordings consist of multivariate time-series data where each measured variable comes from a different electrode. The electrodes are usually sampled between 200Hz to 2000Hz and downsampled to 256Hz for computational reasons. The analog data is digitized to 16 bits of resolution.

The toolbox uses a simple text format to ease the preprocessing of databases. Each time-series, or channel, of the EEG recording corresponds to a single text file where each measured value is on a different line. An EEG recording is represented by a folder containing all the channel text files within. Each channel of the EEG recording should have the same sampling rate and the same duration. A simple XML file encapsulates all the information required to process the multivariate time-series, such as the record name, the sampling rate, the name and filenames of each channel file, and the labels of the different events present in the recording.

2.2. Feature Extraction

The goal of the feature extraction module is to process the multi-channel EEG recordings and extract a database of univariate (single channel) and multivariate (channel pair) features. A large number of features have been proposed in the literature to achieve automatic seizure detection and/or prediction. The toolbox provides direct access to a well-tested implementation of these features. It is worth noting that many of the features depend on some parameters (e.g. window length over which the feature is calculated); the toolbox extracts these features over a wide range of parameter values (this range can be modified if desired, by reducing the range to accelerate computation, or to possibly improve performance by extending the range).

To account for the non-stationarity of the EEG recordings and to detect changes in the brain state, the features are extracted on windows of EEG data that are moving in time. This information is used to look at the similarities of the new window to previously observed windows containing either ictal or interictal data and decide to which category the new window belongs to. Multiple lengths of window sizes are used to get a different resolution of the feature values with respect to time. When moved by a small amount of time, feature values will vary faster when the window is short, but the feature values in longer windows will be more stable to small changes in the brain activity. The toolbox is able to extract any features on a set of window length $W$ with a spacing of $\delta$ between the end of 2 consecutive windows.

Let $E$ represent an EEG recording (multivariate time-series data) and let $c \in E$ be a channel present in the EEG. The vector $c$ represents a time-series consisting of the recordings of the electrode across time. We will denote by $c_i$ the value of the $i^{th}$ sample of the time-series $c$. A window of length $w$ ending with sample $i$ will be written as $c_i^n$. Thus, $c_i^n$ corresponds to the $j^{th}$ value of $c_i^n$ for $0 \leq j < w$. Note that all indices are starting at 0. We use the notation $(a \cdot b)_i = a_i b_i$, $\forall i$ to denote pointwise
multiplication of the vectors \(a\) and \(b\). An example of two consecutive sets of moving windows with \(W = \{200, 400, 1000\}\) and \(\delta = 200\) is illustrated in Figure 2.

2.2.1. Univariate Features

Univariate features are extracted from a single channel \(c \in C\) at a time.

- Mean (\(\mu\)):

\[ \mu(c^w_k) = \frac{1}{w} \sum_{l=0}^{w-1} c^w_{kl} \]

The mean captures shifts in the base line of the EEG recordings.

- Variance (\(\sigma^2\)):

\[ \sigma^2(c^w_k) = \frac{1}{w-1} \sum_{l=0}^{w-1} (c^w_{kl} - \mu(c^w_k))^2 \]

The variance is positively correlated to the amplitude of the measurements made by the electrodes.

- Line-Length (\(L\)):

\[ L(c^w_k) = \sum_{l=1}^{w-1} |c^w_{kl} - c^w_{kl-1}|, \]

where \(|\cdot|\) denotes the absolute value. The line-length was introduced by Olsen et al. (1994) and is positively correlated to the high-frequency components contained in the signal and the signal’s amplitude.

- Fast Fourier Transform (FFT):

The spectral characteristics of the brain activity are a natural component to extract from EEG recordings. A windowing function \(w\) such as a Hann or Hamming (Harris, 1978) is applied on the window \(c^w_k\) to prevent the creation of artifacts due to the non-periodicity of the signal.

Let \(w\) be a windowing function such that \(|w| = w\). We denote the FFT of \((h \circ c^w_k)\) as

\[ \text{FFT}(h \circ c^w_k) = \sum_{j=0}^{w-1} (h \circ c^w_k)_j \cdot e^{-\frac{2\pi i j}{w}}, \]

with \(l \in \{1, 2, \ldots, \frac{w}{2}\}\) and \(i\) representing the imaginary number. Given the sampling rate \(f_s\) of the EEG recording and a frequency \(f\), the linear weighted average of the magnitude of the spectrum between \(l_1 = \lfloor \frac{f}{f_s} \rfloor\) and \(l_2 = \lceil \frac{f}{f_s} \rceil\) is returned as the amplitude of \(f\). The analysis of the spectral components of an EEG recording is a standard procedure used as the brain activity is characterized in wave bands.

- Mean of the absolute Convolution (MAC):

Let \(g\) be a finite impulse response (FIR) filter. The convolution of \(g\) and \(c^w_k\) is defined as

\[ (g \ast c^w_k)_l = \sum_{j=0}^{w+|g|-1} g_{|g|-j} c^w_{kj}, \]

where \(l \in \{0, 1, \ldots, w + |g| - 1\}\) and the convolution operator is denoted by \(\ast\). The values for out of bounds indices are assumed to be 0. The mean of the absolute values of the convolved signal is defined as

\[ \text{MAC}(c^w_k, g) = \frac{1}{w + |g| - 1} \sum_{l=0}^{w+|g|-1} |(g \ast c^w_k)_l|. \]

The absolute value of the convolution accounts for changes in the polarity of the signal. The convolution of the signal with a FIR filter at each point in time is related to the overlapping area under the signal and the FIR filter at that time. It is also equivalent to the signal resulting from the pointwise multiplication of the original signal and the FIR filter in the frequency domain. This feature was taken from Osorio et al. (1998).

2.2.2. Bivariate Features

Bivariate features are extracted over a pair of EEG channels \(c, d \in E\) in order to measure the relations between them. The features are extracted over widows of the same size at the same time location, i.e. \(c^w_k, d^w_k\). The following bivariate features were selected from Mormann et al. (2007).

- Linear Coherence (LC):

Let \(g\) be a weighted average filter of size \(|g| < \frac{w}{2}\). For larger \(|g|\), the statistical significance of the measure is increased at the cost of spectral distortion (Kristensen & Kirkegaard, 1986). Let

\[ G(c^w_k, d^w_k, g, h) = g \ast \left[ \text{FFT}(h \circ c^w_k) \ast \text{FFT}(h \circ d^w_k) \right]^* \]

be the sample cross-spectrum of \(c^w_k\) and \(d^w_k\) convolved with \(g\) where \(a^*\) denotes the complex conjugate of \(a\). This estimation of the spectrum is shown to be roughly equivalent to the Welch method with \(M = |g|\) when \(g\) is uniform (Kristensen & Kirkegaard, 1986). We then define the linear coherence as

\[ \text{LC}(c^w_k, d^w_k, g, h)_l = \left| \frac{G(c^w_k, d^w_k, g, h)_l}{\sqrt{G(c^w_k, c^w_k, g, h)_l \cdot G(d^w_k, d^w_k, g, h)_l}} \right|, \]

where \(l \in \{|g| - 1, |g|, \ldots, \frac{w}{2}\}\). Then, for a given sampling rate \(f_s\) and a frequency \(f\), the value of the measure is located at index \(l = \text{round}(\frac{f}{f_s}) + |g| - 1\). The value at \(l\) gives the linear coherence between \(c^w_k\) and \(d^w_k\) at the frequency band defined by \(g\) centered at \(f\). The linear coherence has a value of 1 when there is a perfect linear synchronization and 0 when there is no synchronization. This feature as been used by Quiroga et al. (2002).
- **Maximum Cross-Correlation (MCC):**

Let

\[
C(c^w_k, d^w_k, \tau) = \left\{ \frac{1}{w} \sum_{t=0}^{w-1} c^w_k(t+\tau) d^w_k(t) \right\}_{\tau \geq 0},
\]

\[
C(d^w_k, c^w_k, -\tau) = \left\{ \frac{1}{w} \sum_{t=0}^{w-1} d^w_k(t) c^w_k(t+\tau) \right\}_{\tau < 0},
\]

define the standard linear cross-correlation. Then,

\[
\text{MCC}(c^w_k, d^w_k, r) = \max_{\tau \in \{-r, r+1, \ldots, r\}} \frac{C(c^w_k, d^w_k, \tau)}{\sqrt{C(c^w_k, c^w_k, 0) \cdot C(d^w_k, d^w_k, 0)}},
\]

measures the maximum normalized lag synchronization of the two signals in the range \([-r, -r + 1, \ldots, r]\). The cross-correlation measures the linear similarity of the amplitude of two signals as a function of lag. Therefore, the maximum cross-correlation returns the highest similarity obtained across the range given by \(\tau\). This feature as been used by Quian Quiroga et al. (2002).

- **Nonlinear Interdependence (NI):**

Let

\[
\psi(c^w_k, d, \tau) = (c^w_k(\tau^d+1), \ldots, c^w_k(\tau^d+(d-1)))
\]

be the delay embedding of \(c^w_k\) in \(d\) dimensions with lag \(\tau\). Note that valid values for \(l\) range from 0 to \(w - (d - 1)\tau - 1\). We define the set of indices corresponding to the \(r\) nearest neighbours of \(l\) in \(\psi(c^w_k, d, \tau)\) as

\[
\text{NN}(c^w_k, d, \tau, r) = \arg\min_{|A|=r} \|\psi(c^w_k, d, \tau) - \psi(c^w_k, d, \tau)\|_2,
\]

where \(\|v\|_2\) denotes the Euclidean norm (length) of the vector \(v\) and \(\|v\|_2^2\) naturally denotes the square of \(\|v\|_2\).

If we let

\[
R(c^w_k, d^w_k, d, \tau, r) = \frac{1}{r} \sum_{j \in \text{NN}(c^w_k, d, \tau, r)} \|\psi(c^w_k, d, \tau) - \psi(c^w_k, d, \tau)\|_2^2,
\]

and define \(N = |\psi(c^w_k, d, \tau)| = w - (d - 1)\tau\), one can compute the following measures of non-linear interdependence

\[
\text{NN}_2(c^w_k, d^w_k, d, \tau, r) = \frac{1}{N} \sum_{j=0}^{N-1} R(c^w_k, d^w_k, d, \tau, r),
\]

\[
\text{NN}_2(c^w_k, d^w_k, d, \tau, r) = \frac{1}{N} \sum_{j=0}^{N-1} \log R(c^w_k, d^w_k, d, \tau, r),
\]

where higher values implies higher degrees of non-linear interdependence. This measures a notion of generalizations synchronization that captures an asymmetric relation of dependence between \(c^w_k\) and \(d^w_k\). These features were suggested by Arnhold et al. (1999).

- **Phase Synchrony (PS):**

Let \(a \in [0, 1]\) and \(h\) be a width parameter, we define

\[
\sigma(h, a) = \frac{h}{\sqrt{2} \cdot \text{erf}^{-1}(a)},
\]

where

\[
\text{erf}(\lambda) = \frac{2}{\sqrt{\pi}} \int_0^\lambda e^{-t^2} dt,
\]

is the error function. Then,

\[
G(f, h, a, f_s) = \frac{1}{f_s} e^{2\pi i (f + \frac{1}{2} \lambda)(e^{2\pi i (a)})^2}
\]

is a complex valued function with a frequency response corresponding to a normal distribution centered at \(f\) where \(a\) of the area of its pdf is located between \([f - h, f + h]\). Note that \(f_s\) is the sampling rate of the EEG recording and \(l \in Z\).

Let \(\mathbb{I}_{\exp}\) be the indicator function: having a value of one when the \(\exp\) is satisfied and zero otherwise. Then, we can define the Hilbert transform in the time domain. For \(w\) even, we define

\[
\mathcal{H}(w_j) = \mathbb{I}_{j=\lfloor \frac{w}{2} \rfloor} + \frac{j}{w} \cot\left(\frac{j - \frac{w}{2}}{w} \pi\right) - \cos\left(\left(j - \frac{w}{2}\right)\pi\right) \sin\left(\frac{j - \frac{w}{2}}{w} \pi\right)
\]

when \(j - \frac{w}{2}\) is even and

\[
\mathcal{H}(w_j) = 0
\]

when \(j - \frac{w}{2}\) is odd. For \(w\) odd, the Hilbert transform is defined as

\[
\mathcal{H}(w_j) = \mathbb{I}_{j=\lfloor \frac{w}{2} \rfloor} + 2 \frac{j}{w} \sin\left(\frac{j - \frac{w}{2}}{w} \pi\right) \cos\left(\frac{j - \frac{w}{2}}{w} \pi\right)
\]

for all \(j\)’s. The real part of \(\mathcal{H}(w) \ast c^w_k\) is \(c^w_k\) and the imaginary part corresponds to the Hilbert transform of \(c^w_k\).

Choosing \(g\) to be either of the two discrete complex filters defined above (\(G\) or \(H\)), we can compute the phase of the signal by

\[
\phi(c^w_k, g) = \text{arctan}\left(\frac{\text{Im}(g \ast c^w_k)}{\text{Re}(g \ast c^w_k)}\right),
\]

in which \(j \in [0, \ldots, w-1]\) and \(\ast\) denotes the convolution operator defined above. We will define the phase difference between \(c^w_k\) and \(d^w_k\) as

\[
\Delta(c^w_k, d^w_k, g) = \phi(c^w_k, g) - \phi(d^w_k, g).
\]

Three different features can be extracted from the recordings. They are measures of synchronization that do not depend on the amplitude of the signals, but instead are related to their respective phases. The first is the mean phase synchrony,

\[
\text{PS}_{\mu}(c^w_k, d^w_k, g) = \frac{1}{w} \sum_{j=0}^{w-1} e^{j \mu \Delta(c^w_k, d^w_k, g)}.
\]
which represents the average phase difference between the two signals. 

Alternatively, an index based on the conditional probability (PS\textsubscript{cp}) and another index based on the Shannon entropy (PS\textsubscript{sh}) can be computed. To define these, we first need to separate the interval [0, 2\pi] into equidistant bins. The number of such bins is given by 

\[ L(w) = e^{0.626+0.4\ln(w-1)} \]

as defined by Rosenblum et al. (2001).

Let 

\[ M(c^w, g)_l = \left\{ j : \phi(c^w_j, g)_l \in \left[ \frac{l}{L(w)2\pi}, \frac{l+1}{L(w)2\pi} \right] \right\}, \]

where \( j \in \{0, 1, \ldots, w-1\} \), \( l \) is a bin index and \( M(c^w, g)_l \) is an index set containing the indices of the elements of \( \phi(c^w_j, g) \) that fall into the \( l \)th bin. Then, for each bin, we compute the following value 

\[ \lambda(c^w_j, d^w_j, g)_l = \frac{1}{|M(c^w, g)_l|} \sum_{j \epsilon M(c^w, g)_l} e^{\phi(d^w_j, g)}, \]

which enables us to compute the index based on conditional probability, 

\[ \text{PS}_{cp}(c^w_j, d^w_j, g) = \frac{1}{L(w)} \sum_{l=0}^{L(w)-1} |\lambda(c^w_j, d^w_j, g)_l|. \]

This index is related to the probability that \( \phi(d^w_j, g)_l \) has a certain value given that \( \phi(c^w_j, g)_l \) fell in a certain bin.

As for the index based on the Shannon entropy, we bin the phase differences as 

\[ P(c^w_j, d^w_j, g)_l = \left( \left| \left\{ j : \Delta(c^w_j, d^w_j, g)_l \in \left[ \frac{l}{L(w)2\pi}, \frac{l+1}{L(w)2\pi} \right] \right\} \right| \right), \]

where \( j \in \{0, 1, \ldots, w-1\} \). Then the index is computed as 

\[ \text{PS}_{sh}(c^w_j, d^w_j, g) = 1 + \frac{1}{\ln[L(w)2\pi]} \sum_{l=0}^{L(w)-1} P(c^w_j, d^w_j, g)_l \ln[P(c^w_j, d^w_j, g)_l]. \]

The PS\textsubscript{sh} index represents the Shannon entropy of the binned phase differences between the two signals.

2.3. Computational Complexity

The choice of which features and what parametrizations to use is often related to the time available for their computation. Therefore, it is important to understand the computational complexity of each feature. We will use the Big-Oh notation \( O(\cdot) \) which shows how the computational cost grows in the worst case. We assume that \( |g| \leq w \) in all the possible parametrizations. Note that these are crude bounds that are not necessarily tight with the toolbox implementation.

The univariate features are computed on a single channel of the EEG recording at a time. Therefore, we need to multiply the computational costs in Table 1 by \(|C|\) to account for all the channels.

The bivariate features depend on pairs of channels present in the EEG recordings. Since there exists \( \binom{C}{2} \) such pairs, we need to multiply the costs in Table 2 by this value.

Any computational complexity containing terms higher or equal to \( O(w^2) \) is excessive for values of \( w \sim 1000 \) (5 seconds at a sample rate of 200Hz).

2.4. Feature selection

A detailed survey indicates, among other findings, that seizure characteristics can vary substantially between patients (Mormann et al., 2007). This could be caused by the inherent heterogeneity in either the type of epilepsy, the kind of manifestations or even the treatments taken among individuals (Talathi et al., 2008). Therefore it is essential to select a personalized set of features to maximize detection accuracy for each patient.

To find a set of optimal features and corresponding parametrization for an individual, a large pool of features can be created using multiple different parametrizations for each of them. If this pool is large enough, we can assume that some subset of features will be good for this individual. The task of feature selection is to find such subset.

The machine learning literature provides a large number of methods for automatic feature selection; most of these are computationally intensive and not practical for dealing with hundreds or thousands of different features (Guyon et al., 2006). An alternate approach is to select features by imposing a complexity constraint on learning a simple classifier, which is a standard approach in statistical analysis (Hastie et al., 2009). This complexity constraint creates a trade-off between the number of selected features and the overall performance of the simple classifier. Therefore, only features that increase the performance of the classifier substantially are selected.

We used a logistic regression as our simple classifier, which is defined as 

\[ p(y=1|x; \theta) = \sigma_\theta(x) = \frac{1}{1 + e^{\theta^T x}}. \]

where \( x \) is the feature vector, \( y \) is the corresponding binary class of \( x \) and \( \theta \) are the parameters of the logistic regression. Given a set of training samples \( X \) and corresponding set of labels \( Y \), the optimal set of parameters of a logistic regression is found by maximizing the likelihood of the data given the labels, i.e.

\[ \min_{\theta} \sum_{x \epsilon X} -\log \left[ p(y|x; \theta) \right]. \]

The optimal solution of this optimization might be a \( \theta \) with no zero entries; using all the available features in \( x \). To reduce the number of features used by the logistic regression, we can use an interesting complexity constraint called the l1-norm regularization 

\[ ||\theta||_1 = \sum_{i=0}^{n} |\theta|_i, \]
which measures the density of $\theta$. We can modify the optimization to include the regularization function,

$$\min_{\theta} \sum_{i=1}^{n} -\log [p(y_i| x_i; \theta)] + \lambda||\theta||_1,$$

where $\lambda$ is a parameter that controls the complexity penalty. The parameter $\lambda > 0$ indicates how much to penalize the parameter vector $\theta$ according to its $1$ norm. Therefore, for larger values of $\lambda$, sparse $\theta$ vectors are preferred. Once the optimization is complete, we can look at the non-zero entries of $\theta$ and use these as the selected features. We can then optimize a more complex classifier using only this subset of features. The main advantage of this method is that highly correlated features are not selected by the regression, thus providing a more diverse feature set. Yet, a rich function class can still be considered in the classification step.

Note that iterative methods (Lee et al., 2006) are often used to solve the regularized optimization because of the large number of features for which we want to learn the set of parameters $\theta$.

2.5. Classification

The next step is to train a classifier using the selected feature set. To do so, each feature vector $x$ must be labelled by either positive, when it is located during an ictal event, or negative, when it occurs during inter-ictal data. The classifier then learns to distinguish between the two different classes of data vectors. Once the classifier is trained, its task is to correctly label feature vectors with unknown labels.

Two types of classifiers are currently offered in the toolbox.

i. Extremely Randomized Trees: A forest of Extremely Randomized trees (Extra-Trees) (Geurts et al., 2006) can be used to classify each feature vector. Extra-Trees are an ensemble of randomized decision trees. Each tree in the ensemble is built using the entire training data. At each node, $K$ candidate tests are chosen randomly such that each contains an element of the feature vector and a random cut point. The output of a test $T$ is

$$T(x) = \begin{cases} 
  \text{True} & \text{if } x_i' \geq t \\
  \text{False} & \text{otherwise}
\end{cases}$$

where $f$ is the feature and $t$ is the threshold picked by test $T$. Then, a score based on the information gain is calculated for each candidate test. Let $X$ be a set of training feature vectors with its corresponding set of labels $Y$. We denote the possible outputs of a test by $B = \{\text{True}, \text{False}\}$ and the set of possible labels by $L$. The score $S(T, X, Y)$ of test $T$ on dataset $X$ is defined as

$$S(T, X, Y) = \frac{2I(T(X); Y)}{H(T(X)) + H(Y)},$$

where $H(T(X))$ is the log entropy of the output of the tests on the vectors in $X$, $H(Y)$ is the log entropy of the labels in $Y$ and $I(T(X); Y)$ is the mutual information of the output of the tests on vectors in $X$ and their labels $Y$. Let $p(a)$ denotes the probability of event $a$, then

$$H(T(X)) = \sum_{b \in B} p(T(x) = b) \log[p(T(x) = b)],$$

$$H(Y) = \sum_{l \in L} p(y = l) \log[p(y = l)],$$

and

$$I(T(X); Y) = \sum_{b \in B, y \in L} p(T(x) = b, y = l) \log \left[ \frac{p(T(x) = b, y = l)}{p(T(x) = b)p(y = l)} \right].$$

When the score of each candidate test has been calculated, the one with the highest value is retained and the others are discarded. The data is then split according to the test and children nodes are built. The process continues until either the split dataset becomes correctly separated (i.e. all feature vectors in the set are from the same class) or its size reaches a minimum $n_{\text{min}}$. The label of a leaf is set to be the majority class of the split data set.

ii. Linear/Logistic Regression: Linear and logistic regression can be performed through the integration of the Vowpal Wabbit toolbox (Langford et al., 2012). Vowpal Wabbit was designed for efficiency on large datasets, using a modified online gradient descent technique for learning. This method only looks at each feature vector once. Since the amount of data used in the training set of each patient is small, we use the option that enables multiples passes over the data while optimizing, therefore seeing each sample more than once. We also use an option that calculates the exact adaptive norm for each feature, improving the quality of the resulting solution.

2.6. Performance Analysis

A complete seizure detection algorithm consists of the concatenation of the feature extraction, feature selection and classification steps. To assess performance of the seizure detection algorithm, it is always recommended to use a hold-out dataset, rather than the data that was used in the feature extraction/selection and classification steps. This prevents over-fitting the detection algorithm to the data at hand, and is a procedure commonly referred to as cross-validation (Hastie et al., 2009)). In our experiments, the training was performed using 3-folds cross-validation for each patient. The dataset is separated into 3 folds which each contain ictal and inter-ictal segments. Then, the classifier is trained on two of the three folds and its performance is evaluated on all the data not contained in the training folds. These steps are repeated three times, once for each possible permutation of the training data. The overall performance of the algorithm is then gathered from the three testing phases. The set of data used to create the training folds of each patient contained all ictal segments and twice as much inter-ictal data. The non-seizure data was selected uniformly at random from the inter-ictal segments. Figure 3 shows an example of the 3-fold cross-validation performed in the experiments.
Recall that the output of the classification step is a label of positive or negative for each feature vector and that such a feature vector consists of the concatenation of the features values computed on windows ending at the same time step. We call a set of contiguous positive classifications an alarm. Since the types of classifiers used do not take into account their previous classifications while classifying a new feature vector, their output might be highly sensitive to small changes in the EEG recordings. This would yield a large amount of false detections. To remedy this problem, an alarm is issued only if a minimum number of consecutive positive outputs are observed from the classifier. We call this minimum value the minimum trigger length (MTL) and it is a tunable parameter of the toolbox. Figure 4 shows an example of the output of a classifier with a MTL of 5 on EEG data containing a seizure.

We consider the three following performance metrics:

i. Detection: An alarm issued by the algorithm is deemed a true detection if it overlaps an ictal event and the first positive classification occurred no more than 30 seconds before the start of the ictal event. If a single alarm overlaps 2 ictal events, only the first one is deemed detected.

ii. False Positives: Each alarm present in an inter-ictal segment (2 minutes segment not containing any ictal activity) counts as a false positive.

iii. Latency: The latency is the time between the start of the ictal event and the start of the earliest true detection. Note that if the alarm began before the start of the seizure, the latency is considered to be 0.

2.7. Experimental Methodology

Three datasets were used to assess the toolbox performance. The toolbox could easily be used with other datasets by converting them to a simple text format as described in Section 2.1. We now outline the experimental procedure and present the key results.

2.7.1. Preparation

To lower the memory requirements of the toolbox and to ease the execution of the cross-validation, the EEG recordings for each subject were cut into ictal and inter-ictal segments. First, the EEG recordings were split into non-overlapping 2 minutes segments. Then, segments containing ictal activity were merged back together with an extra 2 segments at the beginning and end (4 minutes padding), when possible. The segments containing ictal activity will be called ictal segments and the other 2 minutes segments will be called inter-ictal segments.

2.7.2. Toolbox Configuration

• Feature Parametrization: The following feature parametrization was used. Let \( f_s \) be the sampling frequency of the signal. Features were extracted from a set of window lengths \( W = \{f_s, 2f_s, 5f_s\} \) with a delay of \( \delta = f_s \) between the end of each consecutive windows.

\[- \sigma(c^w_k) \]

\[- \mathcal{L}(c^w_k) \]

\[- (h \circ c^w_k), \text{ where } h \text{ is a Hann window and } l \in \{1, 2, \ldots, \left\lceil \frac{T}{2} \right\rceil \}. \]

\[- \text{MAC}(c^w_k, g) \text{ where } g \text{ is a filter such that a convolution yields a Daubechies 4 wavelet decomposition at levels } \{1, \ldots, 5\}. \]

\[- \text{LC}(c^w_k, d^w_k, g, h) \text{ where } g = (1, 4, 6, 4, 1), h \text{ is a Hann window and } l \in \{1, 2, \ldots, \left\lceil \frac{T}{2} \right\rceil \}. \]

\[- \text{MCC}(c^w_k, d^w_k, r) \text{ where } r = f_s - 1. \]

\[- \text{PS}_\text{r}(c^w_k, d^w_k, g), \text{PS}_\text{r}^\text{c}(c^w_k, d^w_k, g), \text{PS}_\text{r}^\text{c}(c^w_k, d^w_k, g), \text{ where} \]

\[ g \in \{ \mathcal{H}(w) \}
\[ \mathcal{G}(2, 2, 0.95, f_s) \]
\[ \mathcal{G}(5.5, 1.5, 0.95, f_s) \]
\[ \mathcal{G}(10, 3, 0.95, f_s) \]
\[ \mathcal{G}(14, 1, 0.95, f_s) \]
\[ \mathcal{G}(22, 8, 0.95, f_s) \]
\[ \mathcal{G}(37, 7.5, 0.95, f_s) \]
\[ \mathcal{G}(72.5, 27.5, 0.95, f_s) \] \]

For some of the datasets below, the computation of LC, MCC and PS was omitted because of the increased time requirements incurred by the higher number of channels.

• Feature Selection: Feature selection was performed using a \( l_1 \)-norm regularized logistic regression with \( \lambda = 10^{-5} \). The optimization of the logistic regression was performed using a total of 10 passes over the training data. We started with a learning rate of 1 and decayed it such that on the \( n \)-th pass, the learning rate was given by \( 0.95^{n-1} \). This rate affects how much the current parameters are modified according to the gradient calculated over the training vectors.

• Classification: We consider two types of classifiers. First, we built forests of \( M = 200 \) Extra-Trees with \( K = \lceil \sqrt{|F|} \rceil \), where \( F \) is the set of features, and \( n_{\text{min}} = 3 \) using either all the features or the one selected. We also used the \( l_1 \)-norm regularized logistic regression resulting from the feature selection step.

2.7.3. Dataset 1: Montréal Neurological Institute (MNI)

A rat pilocarpine model of temporal lobe epilepsy was used to collect EEG data (Levesque et al., 2011, 2012). This animal model of epilepsy is highly isomorphic to human epilepsy (Curi et al., 2008). A one hour status epilepticus (continuous stage 5 seizures (Racine, 1972)) was induced in six Sprague-Dawley rats (250-300g) by intraperitoneal injection of pilocarpine (380mg/kg). Then, three days after, surgery was performed to place intracranial bipolar electrodes in the CA3 region of ventral hippocampus, the medial entorhinal cortex, the
ventral subiculum, and the dentate gyrus for rats 45-5, 46-5, and 50-9. For rats 38-5, 39-3 and 39-8, electrodes were placed in both the left and right regions of the CA3, the medial entorhinal cortex and the amygdala. All procedures were approved by the Canadian Council of Animal Care and all efforts were made to minimize the number of animals used and their suffering. The recordings were downsampled to 200Hz without filtering and digitized using a 16 bit analog-to-digital converter. Recordings between the 4th and 15th day after injection were used, for a total of 472 hours of data containing 137 seizures (see Table 3 for the data distribution per rat).

2.7.4. Dataset 2: Freiburg

The Freiburg epilepsy dataset (Freiburg University, 2012) consists of patients suffering from medically intractable epilepsy at the Epilepsy Center of the University Hospital of Freiburg, in Germany. Recordings from three focal and three extra-focal intracranial electrodes from subjects undergoing presurgical monitoring are available. The EEG signals are sampled at 256Hz and digitized using a 16 bit converter. No filters were applied to the recordings. We analyzed data from 18 of the 21 available patients due to a lack of seizures for patients 002, 008 and 013 (less than 3 seizures). A total of 459 hours of EEG recordings containing 79 seizures were processed for the experiments (see Table 4 for the data distribution).

2.7.5. Dataset 3: CHB-MIT

The CHB-MIT dataset (Shoeb, 2009), freely available on Physionet (Goldberger et al., 2000), contains EEG recordings from 24 pediatric patients from the Children’s Hospital of Boston. Non-invasive scalp recordings were made during the monitoring of patients after their withdrawal from anti-seizure medication in order to characterize their seizures and assess the possibility of surgical intervention. The signals were sampled at 256Hz with a 16 bit resolution. For some patients, the set of available electrodes changed between recordings. Therefore we chose the largest subset of electrodes common to a maximum number of recordings for the analysis. On average, 21 channels were used per patient. Patient chb07 was dropped because less than 3 seizures were available. A total of 741 hours of data containing 182 seizures were analyzed (see Table 5 for the data distribution).

3. Results

The effect of the minimum trigger length on the trade-off between the detection rate and the false positive rate for all 3 datasets is illustrated in Figures 5, 7 and 9. As expected, an increase of the MTL decreases both the false positive rate and the detection rate. Effectively, increasing the MTL requires the algorithm to detect epileptiform activity for a longer period of time before raising an alarm. By fixing the MTL according to a maximum acceptable false positive rate, we are able to obtain the corresponding detection rate of the algorithm. In Figures 5, 7 and 9, the marks correspond to the smallest MTL such that the median false positive rate is less than 2 per day, or 0.08 per hour. We can then observe the corresponding detection rate.

Figures 5, 7 and 9 also demonstrate the superior performance of the Extra-Trees, both for the detection and false positive rate, compared to the logistic regression.

3.1. MNI

The Extra-Trees using feature selection are able to detect correctly 127 out of the 137 seizures present in the dataset, which represents a 92.7% accuracy. The corresponding median false positive rate per hour is 0.071 which is about 1.704 false positives per day. The latency of the algorithm is of 21.81 seconds. Figure 6 contains a per rat analysis where we can observe no false positives for rats 45-5 and 50-9. We also obtain a perfect detection rate (no missed seizures) for rats 39-3 and 45-5, although the false positive rate of rat 39-3 is relatively high. Rat 46-5 has a lot of false positives per day, indicating that the algorithm has difficulty isolating differences in the feature patterns of ictal and inter-ictal activity. The feature selection performed using the logistic regression chose an average of 122 features per fold, out of the available 3834.

3.2. Freiburg

The Extra-Trees using feature selection detect correctly 67 out of the 79 seizures, for an accuracy of 84.8% with a median false positive rate of 0.052 per hour, yielding 1.248 false positives per day. The latency of the Extra-Trees using feature selection is of 14.85 seconds. A per patient analysis of the performance of Extra-Trees using feature selection is provided in Figure 8. We observe a perfect detection rate on 9 patients out of 18. Moreover, no false detections occurred in 7 patients. Patients pat006, pat014 and pat015 had high false positive rates. The feature selection algorithm selected an average of 293 features per fold, out of the possible 1944.

3.3. MIT

Extra-Trees with feature selection are able to detect 135 out of the 182 seizures, for an accuracy of 74.2%. Its false positive rate per hour is of 0.076 which is about 1.824 false positives per day. The latency is of 15.16 seconds. A per patient analysis is provided in Figure 10. The algorithm did not detect any seizures for patients chb06 and chb16. Removing these two patients from the computation of the overall detection rate yields a total of 135 detected seizures out of 162, for a detection rate of 83.3%. The algorithm also detected all the seizures in 16 out of the 23 patients. No false positive rates were observed on 5 of the patients, while a high false positive rate was observed in 6 of them. The feature selection algorithm picked an average of 196 features per fold, out of an average of 6790 available.

3.4. Related Work

We show that the toolbox provides state of the art results in seizure detection by comparing it to other published methods. Table 6 contains relevant information about the statistics of the datasets. The comparison of different methods is difficult as only the work by Shoeb was performed on a freely available dataset (CHB-MIT).
For the intracranial EEG (iEEG) recordings, we compare our work to the one performed by Chan. They obtain a sensitivity of 89.4% on their dataset, which is comparable to the 92.7% we have on the MNI dataset. It is important to note that our average false positive rate per hour is 2.46 times lower. For the Freiburg dataset, which also contained iEEG recordings, our sensitivity is a little bit lower at 84.8%, but our average false positive rate per hour is also 3.63 times smaller. The sensitivity of our method could be increased if more seizures were available in the dataset. Indeed, to train their algorithm, they used 10 seizures per patient which is much more than the 2 to 4 seizures that were available for training per patient (recall that a third of the seizures are always withheld for performance evaluation).

Comparing the techniques developed for scalp EEG recordings, we see that the sensitivity of our toolbox (74.2%) is on par with both Saab & Gotman (2005) and IdentEvent (Kelly et al., 2010). The work by Zandi et al. (2010) has higher sensitivity at 90.5%, but an average false positive rate of 0.51 which is 1.7 times higher than the 0.30 obtained by our toolbox. Only the work by Shoeb & Guttag (2010) provides much better results, with a sensitivity of 96% and a median false positive rate of 0.08 per hour. This could be partly explained by the way the algorithm was trained. They performed a full leave-one-out cross-validation, training on all the data except for a single one hour segment and assessing the performance on the remaining segment. Doing so, the sensitivity and latency were calculated when the omitted segment contained a seizure and the false positive rate was calculated when the segment was seizure free. This implies that the algorithm was able to train on almost the complete dataset for each of the folds, therefore lowering the chances of committing errors. Our classifier was trained on average on a mere 12.7% of the available data for each patient.

4. Discussion

The good results obtained using the toolbox on the three different datasets justify the design choices made during its creation. Indeed, we have shown that many techniques used in machine learning are useful for seizure detection.

The use of many different features combined with different parametrization enables to extract a large pool of features for each patient. The results observed for the Extra-Trees without feature selection over the three different databases show that the pool is large and general enough to capture the different properties of ictal events present in different patients.

One of the concerns when using such a large pool of features is the computation complexity, which translates to long computation times during the feature extraction phase. We remediated this problem by using feature selection through a $L_1$-regularized logistic regression. This feature selection technique was shown to reduce the number of features used per patient significantly. Only 3.18% of the available features were used on average for the rats present in the MNI dataset, 15.07% for patients present in the Freiburg dataset and 2.89% for those in the CHB-MIT dataset. Though the reduction in features used was substantial, the Extra-Trees that were trained using only the selected features had a detection rate that was a little bit higher than those without feature selection. This can be explained by the fact that the selected features react more to ictal events: which can also increase the false positive rates. Feature selection not only improves the detection rate of the Extra-Trees, but it also reduces the time required for the extraction of the features, enabling the algorithm to classify data in real-time.

Even though the feature selection helps the performance of the algorithm, the choice of classifier is even more important. As expected, the performance of the Extra-Trees, a high level machine learning classifier, was much better than the simpler logistic regression model. This is not a surprise as the trees can represent much more complex classification boundaries than a logistic model. The default parameters of the training process were effective across the three databases. Increasing the size of the forest of extremely randomized trees would only increase the stability of the decisions made for each feature vector. Therefore, the chosen size becomes a compromise between stability and memory/speed requirements. Also, the choice of $n_{\text{min}} = 3$ is close to the default of 2 suggested in Geurts et al. (2006). We increased the value to 3 in order to eliminate ambiguities when picking the majority labels in a leaf. Higher $n_{\text{min}}$ values reduce the effect of the noise present in the training data. Setting $n_{\text{min}} = 3$ showed to be effective on all 3 datasets. It is important to note that while the Extra-Trees are able to represent much more complex structures, the time used for their creation and during classification is still small.

As with most classifiers, we cannot specify a trade-off between detection and false positive rates directly in the Extra-Trees. This is the reason behind the introduction of the MTL. The tunability of the parameter enables the toolbox to be tailored for the context it is used in. Tasks demanding lower false positive rate can use a higher MTL, whereas tasks that can tolerate a higher false positive rate can lower the MTL, increasing the detection rate.

An interesting property of the toolbox is its set of default parameters. Indeed, the default parametrization of the feature extraction, feature selection and training of the Extra-Trees used, without modifications, on all patients across the three datasets, provided good results. This essentially makes the algorithm “parameter-free” for seizure detection, but the existence of the parameters enables the toolbox to be used under different environments, such as seizure prediction. This could be achieved by tailoring the parametrization of the feature extraction component to extract pertinent information from the pre-ictal phases of the EEG recordings. Then, the Extra-Trees could be trained to detect the pre-ictal phases, thus predicting the occurrence of seizures.

The separation of the toolbox into four distinct components (feature extraction, feature selection, classification and performance analysis) that can be used separately or jointly facilitates the modular development of new methods for seizure detection or prediction. For example, the feature extraction phase can be applied without modification by researchers interested in testing new classification algorithms.

The good detection rate combined with the low false positive rate of our algorithm on the majority of the patients enable its use in multiple contexts. Nursing care facilities could benefit...
from such a system by alerting relevant authorities that a patient is undergoing a seizure. The small amount of false positives per day would not be sufficient for the personnel to ignore the system. Moreover, this algorithm could help for EEG recording analysis by detecting ictal events, reducing the amount of time spent looking for such information. It could also help automate the analysis of new databases by annotating the signals.

Moreover, automated seizure detection is the first step towards seizure prediction. The selection of features on a per patient basis enables the algorithm to discover patterns that are proper to a patient’s seizures. One can then look at their evolution through time in order to assess if they contain the information about imminent seizures. The task of seizure prediction can be daunting; by looking at the trade-offs between detection and false positive rates, with the induced latency, even automated seizure detection is complex. Therefore, the development of good generic algorithms for detection are a necessity before tackling seizure prediction.

We want to highlight the importance of testing new techniques on freely available datasets with a sound methodology in order to ease the comparison of methods used for seizure detection and prediction. This is the reason for which the MNI datasets, containing EEG recordings of rats, is made publicly available at www.cs.mcgill.ca/~geaauln/files/mni_dataset/.

As of now, the toolbox contains very few classification algorithms: linear regressions and Extra-Trees. This is the main area for further development. Though, because of the modularity of the toolbox, the output of the feature extraction/selection steps can readily be used in other machine learning toolboxes such as Weka (Hall et al., 2009). It is also important to note that the toolbox can be used on a wide variety of time-series signals, such as accelerometer data as was performed by Moghaddam et al. (2011).

To conclude, we have presented a novel toolbox for the development of personalized seizure detection algorithms with modest data requirements. Combining a large feature pool with feature selection, one can develop patient specific algorithms without any parameter tuning. We show that the toolbox provides state of the art results following evaluation on three different publicly available datasets.

### 5. Acknowledgements


<table>
<thead>
<tr>
<th>Univariate Feature</th>
<th>Computational Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mu$ ($c_k^u$)</td>
<td>$O(w)$</td>
</tr>
<tr>
<td>$\sigma^2$ ($c_k^u$)</td>
<td>$O(w)$</td>
</tr>
<tr>
<td>$\mathcal{L}$ ($c_k^u$)</td>
<td>$O(w)$</td>
</tr>
<tr>
<td>FFT ($h \circ c_k^w$)</td>
<td>$O(w \log w)$</td>
</tr>
<tr>
<td>MAC ($c_k^w$, $g$)</td>
<td>$O(w \log w)$</td>
</tr>
</tbody>
</table>

**Table 1:** Computational cost of univariate features on a single channel $c_k^u$.

<table>
<thead>
<tr>
<th>Bivariate Feature</th>
<th>Computational Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>LC ($c_k^w$, $d_k^r$, $g$, $h$)</td>
<td>$O(w \log w)$</td>
</tr>
<tr>
<td>MGC ($c_k^w$, $d_k^r$, $r$)</td>
<td>$O(rw)$</td>
</tr>
<tr>
<td>NL$_1$ ($c_k^w$, $d_k^r$, $d$, $\tau$, $r$)</td>
<td>$O(dw^2 + w^2 \log w)$</td>
</tr>
<tr>
<td>NL$_1$ ($c_k^w$, $d_k^r$, $d$, $\tau$, $r$)</td>
<td>$O(dw^2 + w^2 \log w)$</td>
</tr>
<tr>
<td>PS$_{c_k^w}$ ($c_k^w$, $d_k^r$, $g$)</td>
<td>$O(w \log w)$</td>
</tr>
<tr>
<td>PS$_{c_k^w}$ ($c_k^w$, $d_k^r$, $g$)</td>
<td>$O(w \log w + wL(w))$</td>
</tr>
<tr>
<td>PS$_{c_k^w}$ ($c_k^w$, $d_k^r$, $g$)</td>
<td>$O(w \log w + wL(w))$</td>
</tr>
</tbody>
</table>

**Table 2:** Computational cost of bivariate features for a single channel pair ($c_k^w$, $d_k^r$).


Table 3: Distribution of EEG data (MNI)

<table>
<thead>
<tr>
<th>Patient</th>
<th>Ictal Segments</th>
<th>Inter-ictal Segments</th>
</tr>
</thead>
<tbody>
<tr>
<td>pat001</td>
<td>4</td>
<td>720</td>
</tr>
<tr>
<td>pat003</td>
<td>5</td>
<td>720</td>
</tr>
<tr>
<td>pat004</td>
<td>5</td>
<td>720</td>
</tr>
<tr>
<td>pat005</td>
<td>5</td>
<td>720</td>
</tr>
<tr>
<td>pat006</td>
<td>3</td>
<td>720</td>
</tr>
<tr>
<td>pat007</td>
<td>3</td>
<td>738</td>
</tr>
<tr>
<td>pat009</td>
<td>5</td>
<td>716</td>
</tr>
<tr>
<td>pat010</td>
<td>4</td>
<td>732</td>
</tr>
<tr>
<td>pat011</td>
<td>4</td>
<td>720</td>
</tr>
<tr>
<td>pat012</td>
<td>4</td>
<td>1488</td>
</tr>
<tr>
<td>pat014</td>
<td>4</td>
<td>714</td>
</tr>
<tr>
<td>pat015</td>
<td>4</td>
<td>720</td>
</tr>
<tr>
<td>pat016</td>
<td>5</td>
<td>720</td>
</tr>
<tr>
<td>pat017</td>
<td>5</td>
<td>721</td>
</tr>
<tr>
<td>pat018</td>
<td>5</td>
<td>746</td>
</tr>
<tr>
<td>pat019</td>
<td>4</td>
<td>731</td>
</tr>
<tr>
<td>pat020</td>
<td>5</td>
<td>767</td>
</tr>
<tr>
<td>pat021</td>
<td>5</td>
<td>718</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>79</strong></td>
<td><strong>13831</strong></td>
</tr>
</tbody>
</table>

Table 4: Distribution of EEG data (Freiburg)

<table>
<thead>
<tr>
<th>Patient</th>
<th>Ictal Segments</th>
<th>Inter-ictal Segments</th>
</tr>
</thead>
<tbody>
<tr>
<td>chb01</td>
<td>7</td>
<td>1017</td>
</tr>
<tr>
<td>chb02</td>
<td>3</td>
<td>990</td>
</tr>
<tr>
<td>chb03</td>
<td>7</td>
<td>930</td>
</tr>
<tr>
<td>chb04</td>
<td>4</td>
<td>4357</td>
</tr>
<tr>
<td>chb05</td>
<td>5</td>
<td>1020</td>
</tr>
<tr>
<td>chb06</td>
<td>10</td>
<td>1225</td>
</tr>
<tr>
<td>chb08</td>
<td>5</td>
<td>450</td>
</tr>
<tr>
<td>chb09</td>
<td>4</td>
<td>1746</td>
</tr>
<tr>
<td>chb10</td>
<td>7</td>
<td>1080</td>
</tr>
<tr>
<td>chb11</td>
<td>3</td>
<td>959</td>
</tr>
<tr>
<td>chb12</td>
<td>27</td>
<td>330</td>
</tr>
<tr>
<td>chb13</td>
<td>12</td>
<td>750</td>
</tr>
<tr>
<td>chb14</td>
<td>8</td>
<td>570</td>
</tr>
<tr>
<td>chb15</td>
<td>20</td>
<td>780</td>
</tr>
<tr>
<td>chb16</td>
<td>10</td>
<td>390</td>
</tr>
<tr>
<td>chb17</td>
<td>3</td>
<td>540</td>
</tr>
<tr>
<td>chb18</td>
<td>6</td>
<td>900</td>
</tr>
<tr>
<td>chb19</td>
<td>3</td>
<td>810</td>
</tr>
<tr>
<td>chb20</td>
<td>8</td>
<td>660</td>
</tr>
<tr>
<td>chb21</td>
<td>4</td>
<td>869</td>
</tr>
<tr>
<td>chb22</td>
<td>3</td>
<td>840</td>
</tr>
<tr>
<td>chb23</td>
<td>7</td>
<td>526</td>
</tr>
<tr>
<td>chb24</td>
<td>16</td>
<td>278</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>182</strong></td>
<td><strong>22017</strong></td>
</tr>
</tbody>
</table>

Table 5: Distribution of EEG data (CHB-MIT)


Figure 1: Flowchart describing the steps taken by the toolbox. The modularity of the toolbox enables the interchangeability of any subcomponents.

Figure 2: Two consecutive sets of moving windows with lengths $W = \{200, 400, 1000\}$ and $\delta = 200$ are illustrated in the above figure. The horizontal lines correspond to the segment of data on which the features are extracted and the dashed lines show the synchronization of the endpoints of a set of windows.

Figure 3: Example of 3-folds cross-validation for a fictitious patient. All the available data is depicted in the figure and the dashed lines separate the different segments. The segments containing an ictal event are in gray with a shaded background. The other segments with a shaded background were selected uniformly at random across the inter-ictal segments and were put randomly in one of the three folds $\{a, b, c\}$. The training was performed on all segments contained in a pair of folds and testing was performed on all the segments that were not contained in these folds.

Figure 4: Minimum Trigger Length of 5. The alarm is only raised at 31, when the 5th consecutive positive classification is made.
<table>
<thead>
<tr>
<th>Method</th>
<th>Type</th>
<th>Patients</th>
<th>Seizures</th>
<th>Duration (h)</th>
<th>Sensitivity</th>
<th>False Detection Rate (h⁻¹)</th>
<th>Latency (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zandi et al. (2010)</td>
<td>Scalp</td>
<td>14</td>
<td>63</td>
<td>75.8</td>
<td>90.5</td>
<td>0.51 (µ)</td>
<td>7 (m)</td>
</tr>
<tr>
<td>Saab &amp; Gotman (2005)</td>
<td>Scalp</td>
<td>16</td>
<td>69</td>
<td>360</td>
<td>76</td>
<td>0.34 (µ)</td>
<td>10 (m)</td>
</tr>
<tr>
<td>Saab &amp; Gotman (2005) (TM)</td>
<td>Scalp</td>
<td>16</td>
<td>69</td>
<td>360</td>
<td>43</td>
<td>0.78 (µ)</td>
<td>16 (m)</td>
</tr>
<tr>
<td>IdentEvent (Kelly et al., 2010)</td>
<td>Scalp</td>
<td>55</td>
<td>46</td>
<td>1200</td>
<td>79.5</td>
<td>0.09 (µ)</td>
<td>-</td>
</tr>
<tr>
<td>Shoeb et al. (2004) (CHB-MIT)</td>
<td>Scalp</td>
<td>36</td>
<td>139</td>
<td>60</td>
<td>94.2</td>
<td>0.25 (µ)</td>
<td>8 (µ)</td>
</tr>
<tr>
<td>Shoeb &amp; Guttag (2010) (CHB-MIT)</td>
<td>Scalp</td>
<td>24</td>
<td>173</td>
<td>916</td>
<td>96</td>
<td>0.08 (m) (Offline)</td>
<td>4.6 (µ)</td>
</tr>
<tr>
<td>Chan et al. (2008)</td>
<td>iEEG</td>
<td>6</td>
<td>1792</td>
<td>166.6</td>
<td>89.4</td>
<td>0.69 (µ)</td>
<td></td>
</tr>
<tr>
<td>Extra-Trees using Feature Selection, MNI dataset</td>
<td>iEEG</td>
<td>6</td>
<td>137</td>
<td>472</td>
<td>92.7</td>
<td>0.071 (m), 0.277 (µ)</td>
<td>21.81</td>
</tr>
<tr>
<td>Extra-Trees using Feature Selection, Freiburg dataset</td>
<td>iEEG</td>
<td>18</td>
<td>79</td>
<td>459</td>
<td>84.8</td>
<td>0.052 (m), 0.185 (µ)</td>
<td>14.85</td>
</tr>
<tr>
<td>Extra-Trees using Feature Selection, MIT dataset</td>
<td>Scalp</td>
<td>23</td>
<td>182</td>
<td>741</td>
<td>74.2</td>
<td>0.076 (m), 0.296 (µ)</td>
<td>15.16</td>
</tr>
</tbody>
</table>

Table 6: Related Work. Medians are indicated by (m) and averages by (µ). All authors used private datasets with the exception to Shoeb, who used the MIT dataset.
Effect of the MTL on the median false positive rate

Minimum Trigger Length (MTL)

- Extra-Trees using FS
- Extra-Trees
- Logistic Regression

Effect of the MTL on the detection rate

Minimum Trigger Length (MTL)

- Extra-Trees using FS
- Extra-Trees
- Logistic Regression

Figure 5: MNI: Effect of the Minimum Trigger Length. The curves in the two graphs correspond to the evolution of the median false positive rate and the detection rate with respect to the increase of the MTL. The marks show the smallest MTL such that a false positive rate of less than 2 per day is achieved.

Figure 6: MNI: Performance of Extra-Trees trained on features selected by $l_1$-regularized logistic regression, using a MTL of 12. A per patient analysis is presented on the left and the overall statistics across all patients is presented to the right. The error bars correspond to the 95% confidence interval.
Figure 7: Freiburg: Effect of the Minimum Trigger Length. The curves in the two graphs correspond to the evolution of the median false positive rate and the detection rate with respect to the increase of the MTL. The marks show the smallest MTL such that a false positive rate of less than 2 per day is achieved.

Figure 8: Freiburg: Performance of Extra-Trees trained on features selected by $l_1$-regularized logistic regression, using a MTL of 6. A per patient analysis is presented on the left and the overall statistics across all patients is presented to the right. The error bars correspond to the 95% confidence interval.
Figure 9: MIT: Effect of the Minimum Trigger Length. The curves in the two graphs correspond to the evolution of the median false positive rate and the detection rate with respect to the increase of the MTL. The marks show the smallest MTL such that a false positive rate of less than 2 per day is achieved.

Figure 10: MIT: Performance of Extra-Trees trained on features selected by $l_1$-regularized logistic regression, using a MTL of 7. A per patient analysis is presented on the left and the overall statistics across all patients is presented to the right. The error bars correspond to the 95% confidence interval.