Detection of EEG burst-suppression in neurocritical care patients using an unsupervised machine learning algorithm

G. Narula a,⇑, M. Haeberlin b,1, J. Balsiger b, C. Strässle a, L.L. Imbach b, E. Keller a

a Neurocritical Care Unit, Department of Neurosurgery and Institute of Intensive Care Medicine, University Hospital Zürich, Zürich, Switzerland
b Department of Epileptology, Neurology Clinic, University Hospital Zürich, Zürich, Switzerland

ARTICLE INFO

Article history:
Accepted 20 July 2021
Available online 5 August 2021

Keywords:
Burst suppression
EEG
Unsupervised
Machine learning
Neurocritical care

HIGHLIGHTS

• A novel burst suppression detection algorithm that doesn't require annotated data.
• The algorithm adapts to each patient, is fast and provides confidence scores.
• We report competitive performance compared to supervised deep neural networks.

ABSTRACT

Objective: The burst suppression pattern in clinical electroencephalographic (EEG) recordings is an important diagnostic tool because of its association with comas of various etiologies, as with hypoxia, drug related intoxication or deep anesthesia. The detection of bursts and the calculation of burst/suppression ratio are often used to monitor the level of anesthesia during treatment of status epilepticus. However, manual counting of bursts is a laborious process open to inter-rater variation and motivates a need for automatic detection. METHODS: We describe a novel unsupervised learning algorithm that detects bursts in EEG and generates burst-per-minute estimates for the purpose of monitoring sedation level in an intensive care unit (ICU). We validated the algorithm on 29 hours of burst annotated EEG data from 29 patients suffering from status epilepticus and hemorrhage. RESULTS: We report competitive results in comparison to neural networks learned via supervised learning. The mean absolute error (SD) in bursts per minute was 0.93 (1.38). CONCLUSION: We present a novel burst suppression detection algorithm that adapts to each patient individually, reports bursts-per-minute quickly, and does not require manual fine-tuning unlike previous approaches to burst-suppression pattern detection. SIGNIFICANCE: Our algorithm for automatic burst suppression quantification can greatly reduce manual oversight in depth of sedation monitoring.

© 2021 International Federation of Clinical Neurophysiology. Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

1. Introduction

A pattern of neural activity often observed in EEG monitoring in patients in deep coma in the intensive care environment is the “burst suppression” pattern. It consists of a periodic alternation between high amplitude activity and complete or near complete background suppression (Derbyshire et al., 1936; Niedermeyer et al., 1999; Swank and Watson, 1949). It has been defined by The International Federation of Societies for Electroencephalography and Clinical Neuropysiology (IF-SECN) as a “pattern characterized by theta and/or delta waves, at times intermixed with faster waves, and intervening periods of relative quiescence” (Fürbass et al., 2016). Burst suppression patterns are associated with comas of various etiologies, such as severe cerebral hypoxia after cardiac arrest (Zaret, 1985), drug intoxication (Weissenborn et al., 1991), encephalopathies, hypothermia (Pagni and Courjon, 1964) or deep anesthesia (Amzica, 2015). Burst suppression monitoring in the EEG is used to adapt the depth of sedation in a thiopental-induced coma for the treatment of a severe status epilepticus or an increased intracranial pressure. For an adequate treatment of the underlying condition, it is necessary to monitor some derivative measure of burst suppression activity, usually the number of bursts per minute (BPM) or the burst suppression ratio (BSR), to monitor the percentage of time within an interval spent in the suppressed state. The amount of time spent in a suppression phase is an indicator of worse outcomes for patients suffering anoxic brain injury (Cloostermans et al., 2012; Hallberg...
et al., 2010; Young et al., 2004) making it a clinically relevant measure. In clinical practice, to enable the calculation of measures such as bursts per minute, one needs to segment the EEG into burst and suppression phases. When done manually, this can be a very time-consuming procedure, and subject to interrater variability. This motivates the need for automated burst suppression pattern detection.

Here, we present a novel and completely unsupervised burst suppression detection algorithm based on representing EEG data via covariance matrices computed on short time windows, and clustering said covariance matrices into burst and suppression clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters. We study a cohort of 29 patients from the neurocritical care unit who suffered from intracranial hemorrhage with increased intracranial pressure or status epilepticus. We report clusters.

2. Materials and methods

2.1. Patient characteristics

Continuous long-term EEG recordings (cEEG) were collected from N = 29 patients at the Neurocritical Care Unit, University Hospital Zurich. Patient characteristics are detailed in Table 1. Thiopental (dosage range 0-1200 mg/hour), some patients were additionally treated with Midazolam (dosage range 0–60 mg/hour) and/or Ketamine (0–400 mg/hour). The indication for deep sedation in patients with epilepsy was refractory non-convulsive status epilepticus. The indication for deep sedation in patients with sub-arachnoid, epidural or intracerebral hemorrhage was increased intracranial pressure and vasospasms refractory to conventional treatment. This explanatory study, part of the project “ICU Cockpit”, was approved by the local ethics committee. Written consent was given by legal representatives, as all patients were incapable of judgment.

2.2. Rules used for burst annotation

Bursts were annotated by Dr. Haeberlin (M.H) and the MD student Balsiger (J.B). One of the annotators (M.H) has 2 years’ experience of reading over 2000 EEG recordings, the other annotator (J.B) was new to reading EEG and was instructed and controlled by an experienced specialist. Bursts were identified as high-voltage slow or sharp waves of different frequencies, with clear suppression phases both before and after a burst. Per definition, the suppressed phases have to be longer than the episodes with bursts. There is no evidence about the minimal duration of the suppressed phases between two bursts. The experience of the neurologists at our ICU showed that a complete suppression between two bursts requires a difference of at least 2.5 seconds between the bursts. A burst is defined to occur in all recorded channels. Artifacts can be present in all or only some of the recorded channels and show different frequencies and morphology than bursts. All burst annotations were performed using the software EEG Focus (BESA GmbH, Gräfelfing, Germany). Our motivation for including patients without bursts (or extremely low amplitude bursts) was to find out if the algorithm reported only suppression when it learned clusters on such a patient. Ground truth here simply consists of 100 % suppression and 0% burst.

2.3. Signal acquisition and pre-processing

EEG waveforms were recorded using two EEG devices. In 9 patients we used the Component Neuromonitoring System (CNS – 210, Moberg Research Inc., Ambler PA, USA) and in 20 patients Nihon Kohden EEG-1100 recorder (Nihon Kohden, Irvine CA). For 17 of 29 patients, the sampling rate was 200 Hz, in 3 patients it was 500 Hz and in the remaining 9 it was 256 Hz. We did not resample the data to a single sampling rate for our algorithm because its computational steps do not depend on sampling rate. Eight channels along with a reference electrode and electrocardiogram were recorded in a Frontal montage sampling scheme, namely electrodes Fp1, Fp2, F7, F8, T3, T4, T5, T6. Six derived montage channels were extracted (Fp1-F7, F7-T3, T5-T3, Fp2-F8, F8-T4, T6-T4). We band-pass filtered the six montage channels between 0.5 and 15 Hz to remove baseline drifts and high frequency noise. For band-pass filtering, we used 18th order Chebyshev type I filter (using the scipy.signal.iirfilter function from the python scientific computing library Scipy (Virtanen et al., 2020)). The filtering was performed using the sosfiltfilt command to avoid any numerical issues with filter construction and as well as avoiding phase delays due to filtering.

2.4. Data windowing

EEG waveforms were then segmented into windows of 2 secs (400/512/1000 samples depending on the sampling rate) with no overlap between successive windows. Thus, one window of data was shaped as a [T\ samples × 6 channel] matrix, where T\ samples = 400 (sampling rate = 200 Hz) or 512 (sampling rate = 256 Hz) or 1000 (sampling rate = 500 Hz). The data windows corresponding to the first 15 min of data were considered the “training” data subset and the rest of the data was only used for prediction.

2.5. Artifact suppression

To eliminate artifacts, we calculated the maximum absolute value of the EEG signal in each data window separately for each of the six channels. If the maximum absolute value was > 200 µV, we counted the window in that particular as an artifact window. We replaced the data in such (window, channel) pairs with Gaussian white noise with mean 0 and standard deviation 1.

2.6. Covariance matrix estimation

In each 2-sec window, we computed the empirical covariance matrix of the [T\ samples × 6] data matrix. Briefly, we denote data
in window \( i \) as \( X_i \), a collection of vectors \( X_i = \{ x_i^1, x_i^2, \ldots, x_i^N \} \) where each \( x_i \in \mathbb{R}^6 \) and hence, \( X_i \in \mathbb{R}^{6 \times N} \), with \( T = 400/512/1000 \). Thus, the empirical covariance in each epoch was computed as:

\[
C_i = \frac{1}{T-1} \sum_{t=1}^{T} (x_i^t - \bar{x})(x_i^t - \bar{x})^T
\]

where \( C_i \) is of size \( [6 \times 6] \). Here, \( x_i^t = \frac{1}{2} \sum_{c_i} x_i^t \) is the estimate of the average vector in the epoch. To avoid any singular covariance matrices due to numerical issues, we add a small constant \( (1 \times 10^{-5}) \) to the diagonal of each covariance matrix. This allows for invertibility.

2.7. Distance and similarity between covariance matrices

Förstner and Moonen (2003) provide a metric to calculate the distance between two positive semi-definite symmetric matrices, which contains the space of covariance matrices. For any two covariance matrices \( C_1, C_2 \), the metric is defined as follows:

\[
d(C_1, C_2) = \sqrt{\sum_{i=1}^{N} (\ln z_i(C_1, C_2))^2}
\]

where \( \ln \) is the natural logarithm and \( z_i \) are the eigenvalues of the generalized eigen-value problem \( C_1 v = \lambda C_2 v \). Förstner and Moonen prove that this is in fact a Riemannian metric in the manifold of positive semi-definite matrices and is invariant to affine transformations.

Pairwise similarity between a collection of \( N \) matrices \( \{C_i\}_{i=1}^N \) was estimated as follows:

\[
S(C_i, C_j) = \exp \left( -\frac{d(C_i, C_j)^2}{2\gamma^2} \right)
\]

Here, the value of \( \gamma = \text{median}(d(\ldots)) \).

2.8. Spectral clustering

We use the spectral clustering algorithm to cluster the similarity matrix \( S \) into two clusters. The spectral clustering algorithm is a method that clusters data using a so-called “similarity graph”. A similarity graph contains vertices (data points) and edges, i.e. connections between data points. Edges can either be unweighted (connected or not connected) or weighted, where the weights are non-negative. Consider a dataset with \( N \) points that are connected via a weighted adjacency matrix (in our case this is exactly the similarity matrix \( S = (w_{ij})_{i,j=1,\ldots,N} \geq 0 \)). By construction, the matrix \( W \) is symmetric. The degree \( d \) of a vertex \( x_i = \sum_{i=1}^{N} w_{ij} \). This lets us construct the degree matrix \( D = \text{diag}(d_1, d_2, \ldots, d_N) \). The clustering algorithm according to Shi and Malik (2000) proceeds as follows.

1. Compute the unnormalized graph Laplacian given by \( L = D - W \).
2. Compute the first two eigenvectors of the generalized eigen-value problem \( LU = \lambda Du \).
3. Let \( U \in \mathbb{R}^{N \times k} \) be the matrix containing the column vectors \( u_1, u_2, \ldots, u_k \). Then, for each data point \( y_i \in \mathbb{R}^k \), the row in \( U \) containing the coordinates of the data point.
4. Cluster the points \( \{ y_i \}_{i=1,\ldots,N} \) using the k-means algorithm into two clusters (Lloyd, 1982).

We set \( k = 2 \) for our algorithm. We used the Python package sklearn.cluster.SpectralClustering to perform the above steps. The random seed of the algorithm was set to 99.

2.9. Cluster label assignment

The spectral clustering algorithm returns cluster identities \( Y_i \) for each window ‘i’ of the training data (consisting of \( N \) total windows). Since the clustering method does not what cluster identity \( 1 \) or \( 0 \) actually means, we needed a method to automatically map cluster identity to burst or suppression labels. To map the cluster identity to the label (i.e. “burst” = cluster 1 and “suppression” = cluster 0 or vice versa), we estimated the “burst-ness” of each cluster i.e. if a cluster contained data windows that provided more evidence of being more like bursts, our approach assigned a higher probability of labelling that cluster as burst, and in the same vein, for windows that provided more evidence for being suppression-like, we increased the probability of assigning the cluster to suppression. We used the following rules to characterize the “burst-ness” of a cluster: 1. The burst cluster should have higher energy in the signal than suppression and 2. There should be fewer “burst” data windows than “suppression” data windows. These rules are constructed based on how a neurologist would distinguish a burst from a suppression window. We now formalize these rules.

First, we compute the median energy \( Z \) of the signal in each cluster. Formally

\[
Z_1 = \text{median}(E_i | Y_i = 1, \ i = 1, \ldots, N)
\]

\[
Z_0 = \text{median}(E_i | Y_i = 0, \ i = 1, \ldots, N)
\]

Here, \( E_i \) is a function that refers to the total energy of the signal over all channels and time steps in a single window ‘i’:

\[
E_i = \sum_{c=1}^{C} \sum_{t=1}^{T} (X_{ci}^t)^2
\]

Here, ‘i’ indexes time and ‘c’ indexes channel number. Second, we compute the number of data windows assigned to each cluster \( (N_c = \text{number of windows in cluster} 0, N_1 = \text{number in cluster} 1) \). We then compute the probability that cluster 0 is the burst cluster using the following formula:

\[
\Pr(\text{burst} = \text{cluster} 0) = \frac{1}{2} (P_1 + P_2)
\]

where \( P_1 = \frac{Z_0}{N_0 + N_1} \) and \( P_2 = 1 - \frac{N_0}{N_0 + N_1} \). The probability \( P_1 \) is the proportion of energy in cluster 0. If \( P_1 \) increases, we hypothesize that this provides more evidence that the burst cluster is cluster 0 based on the assumption that bursts should have more signal energy than suppressions. The probability \( P_2 \) is the proportion of windows assigned to cluster 1. Here, the assumption is that bursts are rarer than suppressions. Thus, the more skewed toward larger values \( P_2 \) gets, the more likely that cluster 0 is the burst cluster. We combine these two probability measures by simply taking an average. The decision to assign the label “burst” to cluster 0 is taken if \( \Pr(\text{burst} = \text{cluster} 0) > 0.5 \).

2.10. Clustering confidence assessment

In addition to the classification of time windows, our algorithm can compute a percent score (values between [0, 1] or equivalently 0–100%) that indicates how confident the algorithm is in labelling clusters as burst or suppression after clustering all the data windows. To do so, we first compute the probability \( p \) that cluster 0 is the burst cluster, i.e. \( \Pr(\text{burst} = \text{cluster} 0) = p \). Then, confidence is computed as follows:

\[
\text{Confidence} = \begin{cases} 0.5 - p & \text{if } p < 0.5 \\ p - 0.5 & \text{if } p \geq 0.5 \end{cases}
\]
Thus, if the algorithm is highly unsure about assigning cluster labels (see Cluster label assignment), $Pr(burst = cluster0)$ will be near 0.5. Plugging that as the value of $p$ in the equation above will give a confidence near 0%.

2.11. Prediction on new data

We used a K-nearest neighbor classifier with 5 neighbors to predict the label of new data windows following the unsupervised training data period. Consider a dataset with N training data points $D = \{(x_1,y_1), (x_2,y_2), \cdots (x_N,y_N)\}$ where $x_i$ are the inputs to the classifier and $y_i$ are the class labels. Given a distance metric $d$, a K-nearest neighbor method computes the distance $d(x,x_i)$ between a test data point $x$ and all the training data points ($x_i \in D$). Then, the K (=5 in our algorithm) nearest training data points are retrieved and the predicted class $y$ of the test data point is the majority vote over the class labels of these K-nearest neighbors. We use the Python package sklearn.neighbors.KNeighborsClassifier from the scikit-learn library (Pedregosa et al., 2011) to perform the classification.

2.12. Supervised learning with a deep neural network model

We wanted to compare our unsupervised approach which adapts to each patient and does not learn any parameters to a more advanced supervised approach. We trained two deep convolutional neural networks to predict the burst or suppression labels of data windows using a) the ground truth burst annotations from the expert (abbreviated SUPERVISED-NET) and b) the burst annotations provided by our burst-detection algorithm (abbreviated UNSUPERVISED-NET). The input six channel EEG signal was resampled to 200 Hz in subjects with a higher sampling rate, such that the output of convolutions performed by the network were consistent across all subjects. The network consisted of 4 layers of downsampling 2-D convolutions with filter kernel size = 4, dilation = 1, padding = 0 and stride = 3. The number of filters used in each layer (in order of layer depth) was = [80, 60, 40, 20]. Each convolutional...
layer was followed by Layer Normalization (Ba et al., 2016) and LeakyReLU (leak = 0.2) activation functions. The convolutional layers were followed by a single hidden layer (with 50 neurons) dense neural network. The first layer projected the flattened output of the convolutions to 50 dimensions, followed by Batch Normalization (Ioffe and Szegedy, 2015) and ReLU activation. The 50-dimensional hidden layer activations were then reduced to one output dimension which acted as the “logit” for classification. Binary cross entropy was used as a loss function and the Adam optimizer (learning rate = 1e-4, beta = (0.1,0.99), weight decay = 0.) with batch size 32 was used to perform stochastic gradient descent on the loss function with respect to the parameters. We used the Pytorch library for training and evaluating neural networks.

2.13. Algorithm training and testing procedure

We recorded EEG signals in a bipolar frontal montage from 29 patients undergoing deep sedation with Thiopental for treatment of status epilepticus or an increased intracranial pressure after SAH/EDH/ICH or epilepsy (see Patient characteristics section 2.1 and Table 1 for details). Short segments of data from two example patients are shown in Fig. 1A. After recording the EEG data, we preprocessed the signals by band-pass filtering the bipolar frontal montage channels (see Methods section 2.3, Signal acquisition and pre-processing). We then segmented the 6 channel time series into short windows. From preliminary analysis with two patients, we found that a window length of 2 sec (in comparison to 1 or 3 sec) was appropriate for the detection of bursts and suppression phases. We detected large artifacts by using a threshold (200 µV) on the maximum value of the signal in a window. The artifacts in these windows were suppressed by replacing them with low variance Gaussian noise.

Our algorithm (abbreviated BSUPP), depicted as a flow chart in Fig. 1B, proceeds by first pre-processing and segmenting the data from each patient into windows as stated above. Then, we computed the empirical covariance matrix in each window, followed by computing distances between covariance matrices in the subset of the data used for training (first 15 min of data that contains a burst-suppression pattern). Distances were computed using the Förstner-Moonen metric (see Methods section 2.7, Distance between covariance matrices). The (N training windows × N training windows) distance matrices were transformed into similarity matrices using the squared exponential function. Then, we used the Spectral Clustering algorithm to cluster the similarity matrix into two potential clusters. To map the clusters (with value 0 or 1) to the labels “burst” or “suppression”, we compared the total energy of the signals in cluster 1 against cluster 2 (see Methods section 2.9, Cluster label assignment). The cluster with higher total energy was labelled the “burst cluster”. With the labels identified by clustering and labelling the training data subset, we used a K-nearest neighbor (n = 5 neighbors) classifier to predict the labels on new data windows after they were transformed to covariance matrices using the same metric as before. We then apply a rule used by neurologists at our ICU to count bursts in a burst-suppression: bursts that occur with 2.5 seconds of each other are counted as a single burst. This is done to avoid overcounting bursts without a clear suppression phase between them. M.H and J.B annotated a total of 29 hours (1740 min) of EEG recordings. Notably, the whole process of pre-processing, covariance computation, clustering and labelling requires approximately 2 seconds per minute of EEG data. In contrast, manual annotation by experts can take anywhere from 5 to 10 minutes per hour of data depending on the level of ambiguity in the presented burst suppression pattern. This implies around 5–10 seconds per minute of EEG. Furthermore, this

Fig. 2. Algorithm performance on two representative patients. From top, 6 rows: 6 frontal montage channels were used to perform clustering and classification of bursts, scale bar y-axis = 50 microvolt. These patients show contrasting burst characteristics. Left, clear suppression periods are not visible. Right, burst durations are short and bursts have low amplitude. Bottom row, model burst predictions are shown as positive (thick) rectangular pulses while ground truth annotations are shown as negative (thin) pulses. Note that, model predictions are constant for an entire window of 2 sec and are shown as short pulses placed in the middle of the window for display purpose only.
estimate does not include the time taken to export the data, import it into a visualization software and export again for use. Three minutes of band-pass filtered EEG data from two example patients with ground truth burst annotations and estimated bursts from the algorithm are shown in Fig. 2A. To mark the burst times, we represent the window with the burst label by a short rectangular pulse in the middle of the window. Note that, we have collapsed all bursts occurring within 2.5 secs of each other into one burst. This was done recursively until the condition of no consecutive bursts within 2.5 sec was satisfied.

3. Results

We tested our novel, unsupervised burst-suppression detection algorithm on EEG data from 29 intensive care patients who were treated at the Neurocritical Care Unit, University Hospital Zurich (see Methods section 2.1, Patient characteristics for details on the patient pool). Our algorithm proceeds as described in Methods section 2.12. The output from the algorithm for a short subset of the data is shown in Fig. 2 for two representative patients.

3.1. Model performance quantification

We used our algorithm to compute bursts per minute (abbreviated BPM) estimates for each patient. Since the algorithm clusters and predict bursts individually from each patient, it is valid to quantify the mean absolute error (MAE) of BPM (mean(|ground truth BPM – estimated BPM|)) for each annotated minute of data from each patient separately. Overall, the average (SD) MAE over 29 patients was 0.93 (1.38). The MAE for each patient is plotted in Fig. 3A. We also computed classification accuracy metrics such as sensitivity (proportion of true bursts identified), specificity (proportion of true suppressions identified), area under the receiver-operator curve (abbreviated AUROC, an overall accuracy measure, 0.5 = random guessing, 1.0 = perfect prediction) and negative predictive value (accuracy in estimating suppression), described in Table 2. Our dataset included three patients where there were no bursts identified by the expert, thus measures of sensitivity and specificity do not apply. Our motivation for including patients without bursts (or extremely low amplitude bursts) was to find out if the algorithm reported only suppression when it learned clusters on such a patient. Ground truth here simply consists of 100 % suppression and 0% burst. However, on patients where there were more than zero instances of bursts (n = 26), the average (SD) of sensitivity was 0.81 (0.21), specificity was 0.81 (0.24) and AUROC of 0.82 (0.14). The MAE as well as sensitivity, specificity, negative predictive value and AUROC of each patient are tabulated in Appendix Table 1.

The median absolute error in BPM estimation was much lower (0.38) than the mean, because the mean error is dominated by a few patients where the algorithm failed acutely. The MAE for three patients was greater than 2 BPM. Fig. 3C and D show a short segment of EEG from two of the patients with the worst model performance. In one case, the patient’s data was corrupted by large amounts of noise in some channels and complete suppression in others (Fig. 3C), while in the other, the burst amplitudes were almost indistinguishable from suppression. In such a situation, it is important that the algorithm produces some kind of warning to the attendant physicians or nurse in case it is not confident about segmenting the data into burst and suppression clusters. To do so, we constructed a Confidence score (0–100%) using our cluster labelling method (see Methods section 2.10, Clustering confidence assessment). Fig. 3B shows the relationship between

![Figure 3](image_url)

Fig. 3. Model performance and failure analysis. (A) Mean absolute error (MAE) of bursts per minute (BPM) estimation for n = 29 patients. Two patients where the algorithm failed (MAE > 3) are highlighted with colored arrows. (B) MAE of BPM vs algorithm confidence in clustering assessment. A clear negative relationship (Pearson correlation coefficient r = -0.62, p < 0.001, n = 29) is visible. (C) Short segment of data from patient 11, indicated by the red arrow in figure panel A. The data contained very high levels of noise. (D) Short segment of data from patient 12, indicated by the green arrow in figure panel A. The data contained bursts of very low amplitude.
Table 2

Unsupervised burst-suppression detection algorithm performance under different metrics. Abbreviations: BPM = Bursts per minute, AUROC = Area under receiver-operator curve.

| Performance metric          | Mean   | Standard deviation |
|-----------------------------|--------|--------------------|
| Sensitivity                 | 0.81   | 0.21 (n = 26)      |
| Specificity                 | 0.81   | 0.24 (n = 26)      |
| AUROC                       | 0.82   | 0.14 (n = 26)      |
| Negative predictive value   | 0.97   | 0.029 (n = 29)     |
| Mean Absolute Error (BPM)   | 0.93   | 1.38 (n = 29)      |

Table 3

Comparison of SUPERVISED-NET (deep network trained with ground truth labels), UNSUPERVISED-NET (deep network trained with labels from our burst suppression detection algorithm) and BSUPP (our burst suppression detection algorithm).

| Performance metric | BSUPP (mean over n = 29 Patients) | SUPERVISED-NET (N = 6 Test patients) | UNSUPERVISED-NET (n = 6 test patients) |
|---------------------|-----------------------------------|--------------------------------------|----------------------------------------|
| Sensitivity         | 0.81                              | 0.48                                 | 0.73                                   |
| Specificity         | 0.81                              | 0.96                                 | 0.85                                   |
| AUROC               | 0.82                              | 0.91                                 | 0.88                                   |
| Negative Predictive Value | 0.97                      | 0.97                                 | 0.98                                   |

this score and the MAE for bursts per minute. There is a clear negative correlation (Pearson correlation coefficient between MAE and Confidence: $r = -0.62, p = 0.0001, n = 29$), showing that errors are larger when the algorithm is not confident about its clustering.

3.2. Comparison to an advanced supervised method

We wanted to compare our unsupervised approach which adapts to each patient and does not learn any parameters to a more advanced supervised approach. Specifically, we trained a deep convolutional neural network (see Methods section 2.12, Supervised learning with a deep neural network model) on our band-pass filtered data windows to classify the windows as burst or suppression. Naturally, we used the ground truth annotations as labels to train the deep network model (SUPERVISED-NET). We also trained another network with same architecture on the labels produced by our burst-suppression detection algorithm (UNSUPERVISED-NET). Hyper-parameter optimization was avoided because our aim was to compare the two networks and avoid unnecessary over-optimization and over-fitting. We randomly split the patients into training and test datasets (23 patients in the training set, 6 in the test set which is an 80%-20% training-test split). The classification loss reduced gradually across 10 epochs of training, after which we evaluated the networks on the test dataset. In Table 3, we compare the results of our unsupervised burst-detector algorithm (BSUPP) against SUPERVISED-NET and UNSUPERVISED-NET. We show only classification metrics and not MAE for BPM. Clearly, the performance of the SUPERVISED-NET is better overall than the BSUPP algorithm. Better performance can be explained due to the well-known capacity of neural networks to approximate any function (Cybenko, 1989). Surprisingly, the SUPERVISED-NET’s performance is only marginally better than the UNSUPERVISED-NET. This indicates that we can combine the relatively simple, non-parametric BSUPP algorithm with a neural network trained on labels generated by BSUPP on some set of “training patients”, in order to improve prediction performance for new patients.

4. Discussion

We have presented a novel, unsupervised, adaptive and fast burst-suppression detection algorithm. The algorithm uses covariance matrices to represent short segments (2 sec windows) of data, a distance metric appropriate to find pairwise distances between these covariance matrices (namely, the Foerstner-Moonen metric) and a clustering algorithm and auto-labelling procedure to generate burst and suppression labels for each data segment. We evaluate the algorithm on 29 hours of data from 29 patients and find that mean absolute error in estimating bursts-per-minute is approximately 1 burst. From a clinical perspective, this error lies in a tolerable range for monitoring coma depth. We characterize the instances where the algorithm fails acutely, and find that this occurs if the noise in the EEG data is very high or if the burst amplitudes are almost at the same level as the average suppression period amplitude. As a possible remedy against such a situation in clinical use, we also construct a confidence score (0–100%) from our model, which shows a negative correlation with error in bursts-per-minute estimation. In an actual use case, the confidence score would be calculated by the algorithm after receiving the training subset of data, and if the score was lower than some threshold (say 70%) an alarm could be emitted by the algorithm to warn the attending physician or nurse that burst-suppression detection in that patient may falter and EEG traces must be controlled visually. Finally, we compare our method to an advanced supervised learning approach using convolutional neural networks. Our algorithm, though simple and non-parametric, does not perform much worse than a neural network learned using ground truth burst annotations and instead provides additional advantages such as being fast, adaptive and not requiring supervision to train.

Previously, several works have described algorithms for automated burst suppression pattern segmentation and BSR estimation. Of note are recent models from Särkelä et al. (2002), where the authors use amplitude thresholds on features such as the Non-linear Energy Operator (Mukhopadhyay and Ray, 1998). Westover et al. (2013) apply a recursive algorithm for signal variance estimation and thresholding to identify bursts. Chemali et al. (2013) introduce a Bayesian approach to explain sequences of observed Burst suppression probabilities using a state space model. These and related models (Leistritz et al., 1999; Lipping et al., 1995) have typically used data that has already been annotated by experts to learn parameters such as segmentation thresholds (Lipping et al., 1995; Särkelä et al., 2002; Westover et al. 2013) or model parameters, such as those of a Bayesian sequence model (Chemali et al., 2013) or a Neural Network (Leistritz et al., 1999). Because these methods use the annotations for parameter estimation, all of these methods fall under the class of supervised learning, and in some cases, require a moderate amount of manual fine-tuning to new patients (Westover et al., 2013). The problem with such approaches is two-fold: first, these methods do not account for dataset shifts that can occur in the particular circumstances of each patient and second, any amount of further annotation or manual fine-tuning is an additional source for errors and a burden on neurologists treating several patients. Finally, even within the same patient, the burst suppression pattern may be non-stationary, thus a fine-tuning would be required again and again as time in the ICU progresses. The advantage of our algorithm is that it can be “re-learned” every 15 minutes, which is the training data duration. Moreover, it can be re-learned with temporal overlap between successive training data, and in parallel, such that there is no time lag induced between training and prediction. Thus, if there is non-stationarity in the burst suppression pattern, one can train a new detector to deal with it.

Furthermore, most of the prior studies on burst suppression detection were significantly underpowered in terms of patient numbers, or were only applied to data from healthy subjects under anesthesia for surgery or on neo-natal EEG. Clustering based approaches have a rich history in the computational analysis of
EEG (Agarwal et al., 1998) and the closest work to ours in terms of methodology is a clustering based method from Fürbass and colleagues (Fürbass et al., 2016). The authors show very competitive results on a large dataset (88 patients). However, they also utilize several hard thresholds for various parameters in their algorithm which then would suffer from pattern definition issues across patient populations.

One of the main drawbacks of our approach is the use of a fixed window size of 2 s which was determined using preliminary analysis on two patients. This imposes a strong constraint on what can be considered as a single burst, explicitly excluding bursts that last longer than 2 seconds. This is partly alleviated by the collapsing of bursts that occur within 2.5 s of each other which corresponds to the idea that bursts should not be followed by bursts, and should be alternated with clear suppression phases. In future work, we intend to remove this constraint while preserving the unsupervised nature of the algorithm.

In conclusion, we provide a novel unsupervised algorithm for online quantitative assessment of coma depth in ICU patients. The algorithm performance was stable and with high accuracy in a heterogeneous group of patients with variable etiologies of coma, medication levels and using different EEG recording devices. Most importantly, the algorithm does not require manual (human) input for establishing the burst-suppression classifier. When validated in a prospective manner, this algorithm could be implemented as an easy-to-implement online classifier for ICU EEG-neuromonitoring.

Author contributions

G.N, M.H, L.I and E.K conceived the project. M.H and L.I acquired data. G.N developed and implemented the algorithm. M.H and J.B annotated patient data. C.S provided software support. G.N, M.H, L.I and E.K wrote the manuscript.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The research was funded by Innosuisse project Nr. 34960.1.

Appendix A. Supplementary material

Supplementary data to this article can be found online at https://doi.org/10.1016/j.clinph.2021.07.018.

References

Agarwal R, Gotman J, Flanagan D, Rosenblatt B. Automatic EEG analysis during long-term monitoring in the ICU. Electroencephalogr Clin Neurophysiol 1998;107(1):44–58. https://doi.org/10.1016/S0168-7643(98)00009-1.

Amzica F. What does burst suppression really mean? Epilepsy Behav 2015;49:234–7. https://doi.org/10.1016/j.yebeh.2015.06.012.

Ba JL, Kiros JR, Hinton GE. Layer Normalization. ArXiv160706450 Cs Stat 2016.

Chemali J, Ching S, Purdon PL, Solt K, Brown EN. Burst suppression probability algorithms: state-space methods for tracking EEG burst suppression 056017. J Neural Eng 2013;10(5). https://doi.org/10.1088/1741-2550/10/5/056017.

Cloostermans MC, van Meulen FB, Eertjes CJ, Hom HW, van Putten MJAM. Continuous electroencephalography monitoring for early prediction of neurological outcome in postanoxic patients after cardiac arrest: A prospective cohort study. Crit Care Med 2012;40(10):2867–75. https://doi.org/10.1097/CCM.0b013e318250f40f.

Cybenko G. Approximation by superpositions of a sigmoidal function. Math Control Signals Syst 1989;2(4):303–14. https://doi.org/10.1007/BF02551224.

Derbyshire AJ, Rempel B, Forbes A, Lambert EF. The effects of anesthetics on action potentials in the cerebral cortex of the cat. Am J Physiol 1936;116(3):577–96.

Fürbass W, Moonen B. A metric for covariance matrices. In: Grafarend EW, Krumm Förstner W, Jäntti V, Yli-Hankala A, Hartikainen K. Adaptive segmentation of burst-suppression patterns in critical care EEG monitoring. J Clin Monit Comput 1999;15:357–67. https://doi.org/10.1023/A:1009990629797.

Lipping T, Jantzi V, Maliakala J, Hartikainen K. Adaptive segmentation of burst-suppression pattern in isoflurane and enfuran anesthesia. Int J Clin Monit Comput 1995;12(3):161–7. https://doi.org/10.1007/BF02372609.

Lloyd S. Least squares quantization in PCM. IEEE Trans Inf Theory 1982;28(2):129–37. https://doi.org/10.1109/TIT.1982.1056485.

Mukhopadhyay S, Ray GC. A new interpretation of nonlinear energy operator and its efficacy in spike detection. IEEE Trans Biomed Eng 1998;45:180–7. https://doi.org/10.1109/10.661266.

Niedermeyer E, Sherman DL, Geocadin RJ, Hansen HC, Hanley DF. The Burst-Suppression Electroencephalogram. Clin Electroencephalogr 1999;30(3):99–105. https://doi.org/10.1177/155005949903000305.

Pagni CA, Courjon J. Electroencephalographic modifications induced by moderate and deep hypothermia in man. Acta Neurochir 1964;13:35–49.

Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: Machine Learning in Python. J Mach Learn Res 2011;12:2825–30.

Shi J, Malik J. Normalized Cuts and Image Segmentation. IEEE Trans Pattern Anal Mach Intell 2000;22:18.

Swank RL, Watson CW. Effects of barbiturates and ether on spontaneous electrical activity of dog brain. J Neurophysiol 1949;12(2):137–60. https://doi.org/10.1152/jnen.1949.12.2.137.

Virtanen P, Gommers R, Oliphant T, Haberland M, Reddy T, Cournapeau D, et al. Scipy 1.0: fundamental algorithms for scientific computing in Python. Nat Methods 2020;17(3):261–72. https://doi.org/10.1038/s41592-019-0686-2.

Westover MB, Wilkens H, Haumann E, Degen PH. Burst suppression EEG with baclofen overdose. Clin Neurosi Neurosurg 1991;91(1):77–80. https://doi.org/10.1016/0303-8467(91)90015-4.