

[Re] Deep Convolution Neural Network and Autoencoders-Based Unsupervised Feature Learning of EEG Signals

Bruno Aristimunha¹, Diogo E. L. Alves¹

¹ Center of Mathematics, Computation and Cognition - CMCC
Federal University of ABC - UFABC

{a.bruno, diogo.eduardo}@aluno.ufabc.edu.br

Abstract. *This paper describes our efforts to implement, acquire similar results and improvement of the ones achieved by the authors of article [Wen and Zhang 2018]. We follow the steps and models described in their article and the same public data sets of EEG Signals. Epilepsy affects more than 65 million people globally and EEG Signals are very important to analyze and recognize epilepsy. Although the efforts in the last years it is still very difficult to extract useful information from this signals and select useful features from numerous them in diagnostic application. We construct a deep convolution network and autoencoders-based model (AE-CDNN) in order to perform unsupervised feature learning. We use the AE-CDNN to extract the features of the available data sets, and then we use some common classifiers to classify the features. The results obtained demonstrate that the proposed AE-CDNN to perform the traditional feature extraction based classification techniques by achieving better accuracy of classification.*

1. Introduction

Epilepsy is a chronic neurological disorder and it is becoming one of the most common neurological diseases in the world. Approximately 1% of world's population is affected by epilepsy representing more than 65 million people affected. This disorder is characterized by the occurrence of spontaneous convulsions due to the abnormal synchronous firing of the cortical neurons. This physical reaction can generate many problems to patients, including physical harm caused by the loss of consciousness, shame and discrimination.

Frequent seizures are dangerous conditions because, at the moment of disruption of the body can occur falls, fractures, burns, car accidents, and other serious physical injuries. Epilepsy can be defined as a permanent predisposition in the brain to cause epileptic seizures.

A person is diagnosed with epilepsy if they have two unprovoked seizures (or one unprovoked seizure with the likelihood of more) that were not caused by some known and reversible medical condition like alcohol withdrawal or extremely low blood sugar. Even when correctly diagnosed and treated, the epileptic patient still suffers side effects and sporadic seizures. The epileptic seizures can cause even irreversible damage to brain, then we can visualize the importance of analyzing epilepsy to improve the life quality and the medical treatments for these patients.

To confirm the diagnostic, epileptologists should usually visually inspect the long-term electroencephalograms of the scalp (EEG). Electroencephalogram (EEG) is a measure of the voltage fluctuation generated by the ion current of neurons in the brain, which reflects the activity of the brain's bio-electricity and may contain many physiological and disease information.

When it was discovered that during a patient's seizure the brain activity changes the EEG has become the most common epilepsy diagnostic tool. Many studies have been made and the general problem consists in acquiring methods to efficiently classify the patients' EEG signals. However, this costly task still presents several challenges for automatic crisis detection, among them: The scarce number of public data sets; The lack of standardization in seizure classification methodologies; The lack of standardization of data preprocessing; The cost of a specialist to label time intervals; The unbalance of the time series given the rare occurrence of the event; The difficulty of reproducing the works in the literature.

With this problem in hand, this paper reproduces the results obtained in [Wen and Zhang 2018], with public data labeled and preprocessed. In addition, we get new results by combining the proposal classifiers into a classifier by set voting, and we add new metrics.

The remainder of this paper is organized as follows: Section 2 presents few works related to classification of epileptic seizures in EEG. Section 3 introduce the methodological proposal employed, and their differences with the work of [Wen and Zhang 2018]. Section 4 lists the experimental validation process using epilepsy datasets. Section 5 presents the corresponding results and analyzes our approach. Finally, Conclusions were summarized in Section 6.

2. Related Work

Several papers use automated methods for detecting seizures. We can extract several discriminative characteristics of the signals, among them, we mention the autocorrelation, probability of synchronization, functional connectivity network properties, EEG morphol-

ogy and the reconstructed powers of the time series.

The oscillatory characteristics present in the time series of patients with epilepsy were extensively studied by temporal frequency analysis for classification [Saab and Gotman 2005, Kuhlmann et al. 2009, Shoeb et al. 2004, Shoeb et al. 2011]. Using this technique, we mention the Discrete Wavelet Transform (DWT), which, despite requiring hand-designed parameters, it is the most used [Ullah et al. 2018].

In the literature, there is no common rule for manually label seizures in databases, which makes it difficult to compare the results of these methods. Besides the few papers that use the same sets, there are few that are looking for the same task.

In [Chua et al. 2011] it is used High Order Spectra (HOS) and spectrum-based energy resources for the automated detection of epilepsy. The proposed method yields good results when using Gaussian Mixture (GMM) for classification (93.11% and 88.78%). In [Nicolaou and Georgiou 2012] we have that the authors extracted the entropy of the permutation of the signals, and employed these in an SVM for classification. The result of this methodology, acquired 93.55% for our first database, in task A vs E.

Some researchers have applied deep belief networks (DBNs) to the detection of seizures [Acharya et al. 2018]. In the line of deep learning algorithms, CNNs attract a growing interest in the literature. In [Hussein et al. 2018] they propose a CNN that learns based on the spectral information of each channel and a LSTM network with a single layer to classify the channels of the objects.

Paper [Xun et al. 2016] propose an approach unusual when decoding each window of possible interval as an “EGG word” from the “EEG” dictionary. They explore temporal knowledge by learning context information from EEG fragments (Context-EEG). The authors obtained a 22.93% error rate in the control classification vs epileptic crisis using the second dataset present in [Wen and Zhang 2018].

Paper [Emami et al. 2019] obtained 100% sensitivity with a simpler methodology than our. For each channel, it builds a autoencoder and through the error of reconstruction is classified. The dataset was self and with no access available. The [Ullah et al. 2018] propose a pyramidal model of one dimension for convolution (P-1D-CNN). The method obtains 99.1 ± 0.9 in our first dataset.

3. Methodology Proposal

In this section we describe implementation details, since the core here is the reproducible aspect of our reference article. We introduce the idea and implementation of autoencoder/feature learning and our version of the model in [Wen and Zhang 2018], explaining the differences we have made to the original model.

From the long methodology, we keep the autoencoder and feature learning as proposed. However, in the classification, we no longer consider classifiers individually, these methods now make up a large ensemble learning classifier, which decides by majority vote the object class.

3.1. Implementation Details

We decided to reproduce the implementation described in the article using Keras [Chollet et al. 2018] and back-end in TensorFlow [Abadi et al. 2016]. Our repository includes the list of all the required libraries employed in acquiring the datasets and running the model (the original and the proposed one). According to the methodology proposed in [la Fuente and Aduviri 2019] we store all the checkpoints for the trained models, for reproduction purposes. Besides that, the training logs can be visualized using TensorBoard tool.

Given the lack of information about implementation in the original paper, some assumptions or cuts are made: The number of epoch in the auto-enconder is assumed to be 5000; The number of samples per batch size is assumed to be 500, due our time constraint; A column of the first database is removed, there is disagreement in the literature on the total instances, 4097 or 4096. In the specific database we use there is 4097. The removed attribute is at the endpoint of each object; The classifier presented in the final subsection was not reproduced for lack of information; The loss function presented in equation 12 of the [Wen and Zhang 2018] was not implemented given the complexity in the chosen library; The value of the seeds selected in all classifiers was 42; The percent in train-test was 80%-20%; The test data were used for validation in Auto-Enconder;

The experiments were performed using a CPU with Intel Core i7-5930K with 3.50 GHz and two GPUs: Nvidia Quadro K5200 and GeForce GTX 970. Some experiments were also run using Nvidia Titan X.

3.2. Autoencoders

The autoencoder implemented is a specific case of neural network structure. It is formed by three layers, the input layer, output layer and a hidden layer. The training is done in order to set the weights of hidden layer to force the input layer and output layer to be as close to each other as possible. Our features are extracted from the hidden layer, which reduces the dimension of data.

Therefore we have a encoding process and a decoding process and we obtain the hidden layer h by applying the encoding function:

$$h = \text{encoder}(x) = g(W * x + b), \quad (1)$$

where W is the weight matrix between input layer and hidden layer. In the decoding function the hidden layer h is the input and $y = \text{decoder}(h)$ as output, the function is defined as follows:

$$y = \text{decoder}(x) = g(W' * x + b'), \quad (2)$$

where W' is the weight matrix between hidden layer and output layer. Since we want input and output to be as close as possible, we have the object function for the model training process:

$$\min \sum |y^{(i)} - x^{(i)}|,$$

where $y^{(i)}$ is the output signal and $x^{(i)}$ is the input signal.

3.3. Feature Learning Model

In this subsection, in order to be brief we will omit equations and minor details, for complete information read [Wen and Zhang 2018, Shueb and Guttag 2010, Emami et al. 2019]. Since we have the dimension reduced by autoencoder we focus on the next challenge: how to obtain effective features from EEG signals. The AE-CDNN implemented follows the steps:

- Encoder: sample input, convolution layer, down-sampling layer, reshape operation, full connection layer, and the feature coding.
- Decoder: feature coding as input, full connection layer, reshape operation, deconvolution layer, up-sampling layer and the reconstruction samples.

Basically, the convolution layer acts as our feature extractor. It performs many successive convolution calculations of the input data and the expectation is to maintain the main components of the input data. The pooling layer is a down-sampling method which reduces data dimension. It uses windows to slid and extract the feature maps. These intervals do not overlap each other and with then we obtain the pooled feature maps. The feature sizes tested were $m = 2, 4, 8, 16, 32, 64, 128, 256^1$.

The convolution and pooling operations can be iterated multiple times. Reshape operation uses the pooled feature maps to construct a one-dimension vector and full connection layer transform this one-dimension vector.

Considering x as the input and y as the output, now we need to re-transform the one-dimension vector which will generate the y output, recall we want to minimize the difference between x and y , we have the following equation to calculate loss:

¹Size $m = 256$ has not been tested in [Wen and Zhang 2018]

$$\text{Loss} = \sum_{i=1}^N |x^{(i)} - y^{(i)}|/N.$$

3.4. Classification

Since we have extracted the features with reduced dimension we use supervised learning models on these features in order to classify the EEG signals. We avail each classifier and then we compare the results obtained with each one. The classical classifiers used are: K-Nearest Neighbors (K-NN), Support-Vector Machine - Linear Kernel and Radial Basis kernel (SVM1, SVM2), Decision Tree (DT), Random Forest (RT), Multilayer Neural Network (MLP), Adaptive Boosting algorithm (ADB) and Gaussian Naive Bayesian (GNB).

The proposed modification combines these classifiers, and creates a single classifier that decides by voting. In short, the classifiers were combined by ensemble learning, and the result of the classification became the classification most voted by the classifiers.

4. Experimental Methodology

In this paper as in our reference paper [Wen and Zhang 2018] we use unsupervised learning method in EEG signals in order to obtain useful features. This process is needed because the original data is high-dimensional and using the autoencoder we can extract features with reduced dimension.

4.1. Bonn University EEG database

We can use different approaches to detect epileptic crisis. Then in order to acquire a comparative measure we verify our outputs using the method described in 3 and the original one showed in [Wen and Zhang 2018]. This database is public and was published by [G. Andrzejak et al. 2002]. The study groups were the control, inter-ictal and ictal distributed into five sets (denotated A-E). Each containing 100 records of 23.6 seconds duration and frequency of 173.6 Hz on a single channel, with 12 bit resolution. The data segments has 4097 samples. These recordings underwent a pre-processing in which the signals had a band filter between 0.53 to 40 Hz. There was also the removal of artifacts such as muscle movements or flicker movements.

Using labels A, B, C, D and E for the subsets we have that A and B contain records of 5 healthy volunteers. Set A corresponds to open-eye activity, and subset B to closed-eye activity. The subsets C and D have signals during the absence (interictal epileptiform activity) of 5 epileptic patients. And E records the signals during epileptic patients' seizure (ictal intervals). According to [Kamath 2015], this dataset is a compilation of recordings under different conditions.

4.2. Performance Measures

According to [Roy et al. 2019], most of the state-of-the-art systems for epilepsy use the metrics defined below, the adaptation of these metrics for evaluating our system contributes to fair comparison with state-of-the-art systems. The definitions of these metrics are given in Table 1.

Accuracy	Precision	Specificity	Sensitivity	F-Measure
$\frac{TP+TN}{TP+TN+FP+FN}$	$\frac{TP}{TP+FP}$	$\frac{TN}{TN+FP}$	$\frac{TP}{FN+TP}$	$\frac{2 \cdot \text{Precision} \cdot \text{Sensitivity}}{\text{Sensitivity} + \text{Precision}}$

Table 1. Metrics and Definition use in our paper. Only the Accuracy was considered in [Wen and Zhang 2018].

where False Negatives - FN is the number of epileptic cases, which are predicted as control, True Positives - TP is the number of epileptic cases, which are predicted as epileptic, True Negative - TN is the number of control case that is predicted as control and False Positives - FP is the number of control cases that are identified as epileptic by the system. In addition, there was also the AUC-ROC (Area Under The Curve - Receiver Operating Characteristic) defined as the cumulative distribution function of the true positive rate vs. the false negative rate denoted by a threshold.

5. Results and Discussion

The results obtained in our reproduction experiment are presented in Table 2. We employed the same methodology as the one used in original paper, performing a 5-fold cross validation of the accuracy for each classifier, and we show the mean values. We observe in the last column the values obtained by joining the classifiers by voting (**Ense**). In addition, we exploited an additional value of m , i.e. the size of the latent feature vector of AE-CDNN-L1, which is $m = 256$. The values observed in Tables 3, 4, 5 demonstrate the results obtained for the proposed additional metrics that we consider important for understanding the behavior of the classifiers.

m	K-NN	SVM-1	SVM-2	DT	RF	MLP	ADB	GNB	AVG	Ense
2	0.678	0.6	0.738	0.714	0.66	0.6	0.744	0.576	0.664	0.73
4	0.75	0.6	0.782	0.724	0.764	0.684	0.732	0.582	0.702	0.774
8	0.774	0.6	0.784	0.726	0.766	0.608	0.776	0.556	0.699	0.782
16	0.734	0.6	0.762	0.71	0.75	0.632	0.698	0.53	0.677	0.752
32	0.798	0.702	0.852	0.738	0.79	0.784	0.8	0.632	0.762	0.802
64	0.8	0.672	0.79	0.77	0.788	0.762	0.81	0.602	0.749	0.802
128	0.806	0.758	0.702	0.698	0.776	0.876	0.784	0.594	0.749	0.846
256	0.78	0.72	0.804	0.69	0.718	0.848	0.764	0.598	0.74	0.854

Table 2. Classification Accuracy Results of AE-CDNN-L1 for Dataset 1

Our results are different of the ones showed in [Wen and Zhang 2018]. However, we do not have full information about the implementation used to obtain the article results.

Based on original article results we have two points of attention: one classifier had a result 36% below the expected and other one had 22% above the expected.

The Figure 1 shows the difference between our results and the original ones. The relation between the vector's size and the classifiers accuracy is showed in Figure 2. In this figure we have on the X axis the number of dimensions and on the Y axis the accuracy of the classifiers.

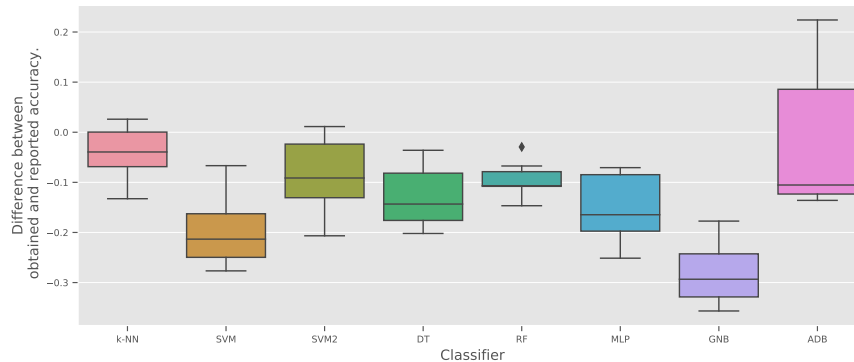


Figure 1. Accuracy difference between original results and our results.

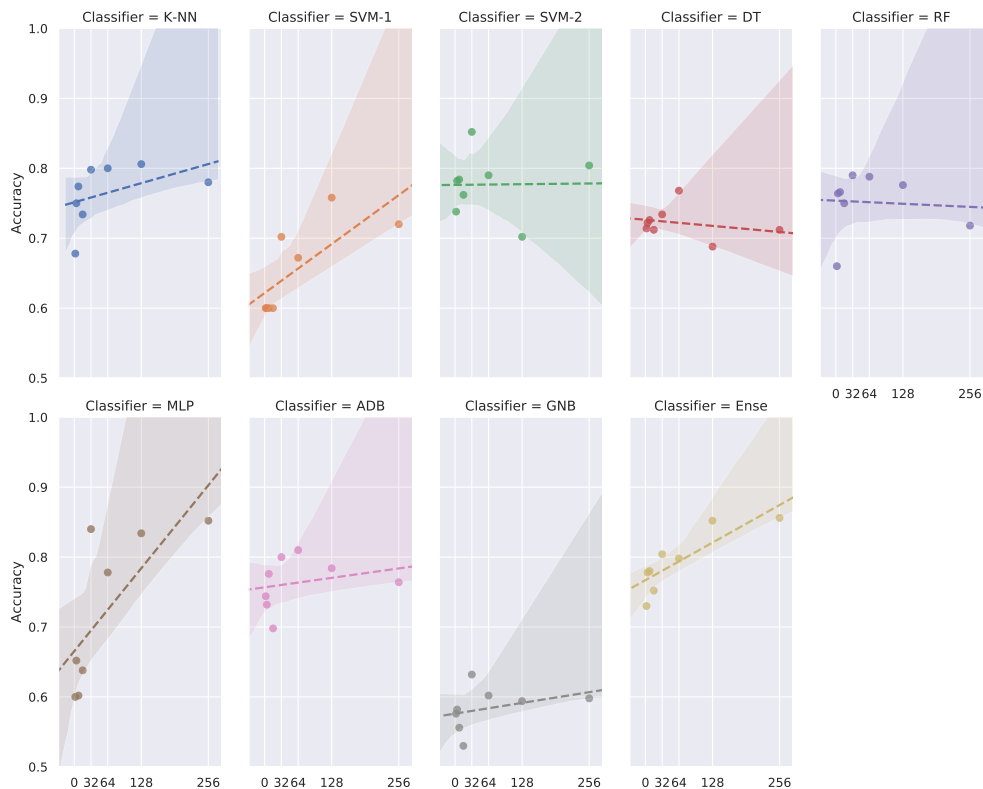


Figure 2. Features Number of autoencoder by accuracy, using linear regression.

The Multilayer Neural Network classifier presents higher angular coefficient than the others, this increase in data volume that leads to the increase of the metric is consistent with the method premise. It is noteworthy that the Support Vector Machine with Linear

Kernel has also high coefficient. The highest Linear coefficient with the lowest associated uncertainty is found in the method proposed by us, the Ensemble method presents an initial result close to the best method for $m = 2$.

Precision Results of AE-CDNN-L1 for Dataset 1										
m	K-NN	SVM-1	SVM-2	DT	RF	MLP	ADB	GNB	AVG	Ense
2	0.75	0.6	0.787	0.746	0.73	0.6	0.794	0.741	0.719	0.774
4	0.841	0.6	0.84	0.741	0.827	0.715	0.787	0.969	0.79	0.837
8	0.851	0.6	0.828	0.788	0.823	0.601	0.843	0.874	0.776	0.833
16	0.815	0.6	0.795	0.765	0.814	0.685	0.762	1.0	0.78	0.813
32	0.899	0.86	0.892	0.858	0.863	0.913	0.845	0.952	0.885	0.932
64	0.865	0.82	0.885	0.914	0.844	0.877	0.853	0.992	0.881	0.947
128	0.87	0.798	0.681	0.779	0.832	0.865	0.823	0.921	0.821	0.922
256	0.809	0.769	0.795	0.797	0.772	0.871	0.799	0.937	0.819	0.932
Specificity Results of AE-CDNN-L1 for Dataset 1										
m	K-NN	SVM-1	SVM-2	DT	RF	MLP	ADB	GNB	AVG	Ense
2	0.65	0.0	0.675	0.595	0.615	0.0	0.68	0.76	0.497	0.65
4	0.795	0.0	0.775	0.565	0.76	0.42	0.68	0.985	0.622	0.77
8	0.8	0.0	0.745	0.665	0.745	0.005	0.78	0.935	0.584	0.76
16	0.75	0.0	0.68	0.63	0.74	0.42	0.66	1.0	0.61	0.735
32	0.88	0.855	0.845	0.835	0.815	0.88	0.77	0.97	0.856	0.92
64	0.815	0.8	0.85	0.9	0.78	0.825	0.78	0.995	0.843	0.94
128	0.825	0.695	0.32	0.71	0.755	0.8	0.73	0.955	0.724	0.895
256	0.695	0.655	0.625	0.73	0.66	0.805	0.69	0.965	0.728	0.91
Sensitivity Results of AE-CDNN-L1 for Dataset 1										
m	K-NN	SVM-1	SVM-2	DT	RF	MLP	ADB	GNB	AVG	Ense
2	0.697	1.0	0.78	0.793	0.69	1.0	0.787	0.453	0.775	0.783
4	0.72	1.0	0.787	0.827	0.767	0.807	0.767	0.313	0.748	0.783
8	0.757	1.0	0.81	0.767	0.78	1.0	0.773	0.303	0.774	0.793
16	0.723	1.0	0.817	0.767	0.757	0.783	0.723	0.217	0.723	0.763
32	0.743	0.6	0.857	0.667	0.773	0.813	0.82	0.407	0.71	0.727
64	0.79	0.587	0.75	0.68	0.793	0.747	0.83	0.34	0.69	0.703
128	0.793	0.8	0.957	0.673	0.79	0.857	0.82	0.353	0.755	0.823
256	0.837	0.763	0.923	0.7	0.757	0.883	0.813	0.353	0.754	0.82

Table 3. Metric results to verify method behavior

By Table 3 we realize that the most precise and specific method was the Gaussian Naive Bayesian, however, when analyzing the behavior in the Sensitivity metric, we do not have a satisfactory result. This indicates that the method pinpoints true negatives rather than true positives. If treated from a medical field, this result is worrying. The cases that the method indicates are true positives, however, this method misses many cases.

We also analyze that we cannot consider SVM-1 or MLP results with the lowest m . The result in specificity indicates that the method behaves unwanted, possibly indicating all values as true positives. This rule burdens the medical system because further detection of the seizure requires further investigation for a complete diagnosis of the disease.

By our method we note that the three metrics indicate that a progression in the number of features generates an improvement in seizure detection. Similar behavior is observed in K-NN, only for accuracy and sensitivity, in this case a beneficial behavior

for the application. No trend was observed in the other methods. Attention is drawn to the 0 values presented in the MLP, SVM1 methods. The average of the methods does not exceed our Ensemble method in any scenario. With the m variation tested, we have that K-NN, RF, ADB worsen in specificity. Our method does not excel in these analyzed metrics.

We analyze the behavior of the F-measure in Table 4. The relationship between sensitivity and precision is captured by this measure, as the MLP, SVM-1, GNB methods did not obtain good results, which contradicts the accuracy result (Table 2). The measures generally show close results with each other. Therefore, the factor to analyze here may be the complexity of the methods.

m	K-NN	SVM-1	SVM-2	DT	RF	MLP	ADB	GNB	AVG	Ense
2	0.721	0.75	0.78	0.769	0.708	0.75	0.785	0.56	0.728	0.776
4	0.775	0.75	0.812	0.781	0.795	0.729	0.775	0.473	0.736	0.809
8	0.801	0.75	0.818	0.765	0.8	0.751	0.806	0.45	0.743	0.812
16	0.766	0.75	0.805	0.762	0.784	0.72	0.742	0.355	0.71	0.787
32	0.811	0.705	0.874	0.75	0.816	0.859	0.831	0.568	0.777	0.816
64	0.826	0.68	0.811	0.778	0.817	0.799	0.84	0.504	0.757	0.806
128	0.829	0.798	0.794	0.72	0.808	0.86	0.819	0.51	0.767	0.869
256	0.822	0.765	0.85	0.744	0.763	0.876	0.805	0.512	0.767	0.871

Table 4. Classification F-Measure Results of AE-CDNN-L1 for Dataset 1

According to [Lorena et al. 2018], we have that the complexity of the task and the algorithms are decisive factors in deciding which methods will be used. Thus, as training is already costly, the complexity of the classification algorithm must be minimized. K-NN and ADB have a lower complexity than the others, and a result close to the champion method. The difference in the performance of the SVM2 method in F-measure with our Ensemble method is close, the difference in averages is only 0.0002. Also, looking at the behavior of the classifiers by the Bayesian Test [Benavoli et al. 2017], we have that the Ensemble method is 52.4% better than SVM2 on Accuracy. We consider a more conservative analysis with $\text{rope} = 0$ and $\text{runs} = 5$, shown in Figure 3.

We see in Table 5 the performance of the AUC-ROC metric that the Ensemble method has the highest performance average, with 79.85%, followed by K-NN with 76.98%. The methods that stand out in this metric showed no deficiency in the Sensitivity and Specificity metrics, given the construction of the measure.

6. Conclusion

In this article, we re-implemented the approach proposed by [Wen and Zhang 2018] an investigation of the accurately tested feature sizes and proposed a new classifier. This classification approach, based on deep learning for detecting epileptic seizures using EGG,

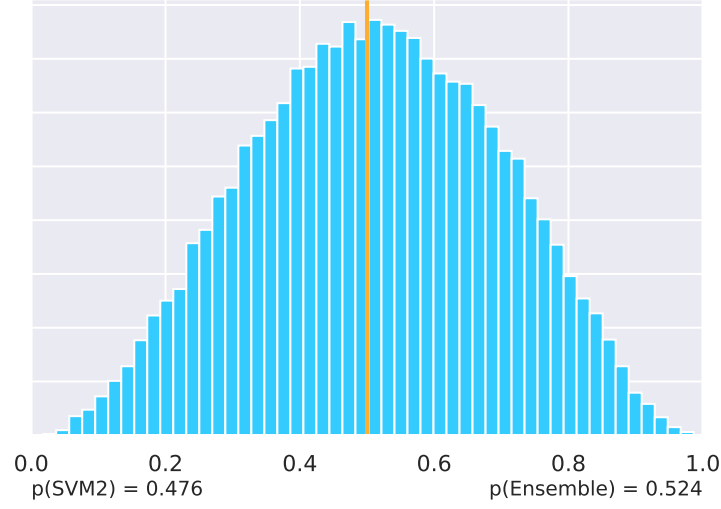


Figure 3. Probabilities using a Bayesian signed-ranks test on Accuracy

m	K-NN	SVM-1	SVM-2	DT	RF	MLP	ADB	GNB	AVG	Ense
2	0.673	0.5	0.728	0.694	0.653	0.5	0.733	0.607	0.636	0.717
4	0.758	0.5	0.781	0.696	0.763	0.613	0.723	0.649	0.685	0.777
8	0.778	0.5	0.778	0.716	0.762	0.503	0.777	0.619	0.679	0.777
16	0.737	0.5	0.748	0.698	0.748	0.602	0.692	0.608	0.667	0.749
32	0.812	0.727	0.851	0.751	0.794	0.847	0.795	0.688	0.783	0.823
64	0.802	0.693	0.8	0.79	0.787	0.786	0.805	0.668	0.766	0.822
128	0.809	0.747	0.638	0.692	0.773	0.828	0.775	0.654	0.74	0.859
256	0.766	0.709	0.774	0.715	0.708	0.844	0.752	0.659	0.741	0.865

Table 5. Classification AUC-ROC Results of AE-CDNN-L1 for Dataset 1

had not been explored previously. We adopted a self-concealment that allows us to construct a smaller representation space. Among the variety of metrics, our method stands out according to the ROC curve. The proposed approach still needs further investigation into the details of the parameters tested by the authors. The method is easily portable to other tasks and codes are a byproduct of this work. As future work we can employ data augmentation that would allow the method to better learn the parameters adopted.

References

- Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. (2016). Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pages 265–283.
- Acharya, U. R., Oh, S. L., Hagiwara, Y., Tan, J. H., and Adeli, H. (2018). Deep convolutional neural network for the automated detection and diagnosis of seizure using eeg signals. *Computers in biology and medicine*, 100:270–278.
- Benavoli, A., Corani, G., Demšar, J., and Zaffalon, M. (2017). Time for a change: a tutorial for comparing multiple classifiers through bayesian analysis. *The Journal of Machine Learning Research*, 18(1):2653–2688.
- Chollet, F. et al. (2018). Keras: The python deep learning library. *Astrophysics Source Code Library*.

- Chua, K. C., Chandran, V., Acharya, U. R., and Lim, C. M. (2011). Application of higher order spectra to identify epileptic eeg. *Journal of medical systems*, 35(6):1563–1571.
- Emami, A., Kunii, N., Matsuo, T., Shinozaki, T., Kawai, K., and Takahashi, H. (2019). Autoencoding of long-term scalp electroencephalogram to detect epileptic seizure for diagnosis support system. *Computers in biology and medicine*.
- G. Andrzejak, R., Lehnertz, K., Mormann, F., Rieke, C., David, P., and Elger, C. (2002). Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 64:061907.
- Hussein, R., Palangi, H., Ward, R., and Wang, Z. J. (2018). Epileptic seizure detection: A deep learning approach. *arXiv preprint arXiv:1803.09848*.
- Kamath, C. (2015). Analysis of eeg dynamics in epileptic patients and healthy subjects using hilbert transform scatter plots. *Open Access Library Journal*, 2(1):1.
- Kuhlmann, L., Burkitt, A. N., Cook, M. J., Fuller, K., Grayden, D. B., Seiderer, L., and Mareels, I. M. (2009). Seizure detection using seizure probability estimation: Comparison of features used to detect seizures. *Annals of biomedical engineering*, 37(10):2129–2145.
- la Fuente, A. D. and Aduviri, R. (2019). [Re] Variational Sparse Coding. *ReScience C*, 5(2):#2.
- Lorena, A. C., Garcia, L. P., Lehmann, J., Souto, M. C., and Ho, T. K. (2018). How complex is your classification problem? a survey on measuring classification complexity. *arXiv preprint arXiv:1808.03591*.
- Nicolaou, N. and Georgiou, J. (2012). Detection of epileptic electroencephalogram based on permutation entropy and support vector machines. *Expert Systems with Applications*, 39(1):202–209.
- Roy, Y., Banville, H., Albuquerque, I., Gramfort, A., Falk, T. H., and Faubert, J. (2019). Deep learning-based electroencephalography analysis: a systematic review. *Journal of neural engineering*.
- Saab, M. and Gotman, J. (2005). A system to detect the onset of epileptic seizures in scalp eeg. *Clinical Neurophysiology*, 116(2):427–442.
- Shoeb, A., Edwards, H., Connolly, J., Bourgeois, B., Treves, S. T., and Guttag, J. (2004). Patient-specific seizure onset detection. *Epilepsy & Behavior*, 5(4):483–498.
- Shoeb, A. and Guttag, J. (2010). Application of machine learning to epileptic seizure detection. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML’10, pages 975–982, USA. Omnipress.
- Shoeb, A., Kharbouch, A., Soegaard, J., Schachter, S., and Guttag, J. (2011). A machine-learning algorithm for detecting seizure termination in scalp eeg. *Epilepsy & Behavior*, 22:S36–S43.
- Ullah, I., Hussain, M., Aboalsamh, H., et al. (2018). An automated system for epilepsy detection using eeg brain signals based on deep learning approach. *Expert Systems with Applications*, 107:61–71.
- Wen, T. and Zhang, Z. (2018). Deep convolution neural network and autoencoders-based unsupervised feature learning of eeg signals. *IEEE Access*, PP:1–1.
- Xun, G., Jia, X., and Zhang, A. (2016). Detecting epileptic seizures with electroencephalogram via a context-learning model. *BMC medical informatics and decision making*, 16(2):70.