Université Paris Cité

EEG Epileptic Spike Detection

Students:

Professor:

UFR Mathematics and Computer Science

April 24, 2022

Abstract: The electroencephalogram (EEG) is widely used in modern medicine, not only to help diagnose diseases such as brain tumors, brain dysfunction, encephalitis, stroke, sleep disorders, dementia. intelligence, ... but this is also a specific test to diagnose epilepsy. In this report, we make the first comparison studies different learning models based on epileptic data for the purpose of automatically detecting whether a case (such as an EEG data series) records a seizure.

Keyword: EEG; Epileptic Spike

1 Introduction

With nearly 40 million patients globally, epilepsy is the most frequent neurological illness, second only to stroke (2015). Many research on the diagnosis of neurological illnesses have used the electroencephalogram in recent years (EEG). The EEG measures voltage changes in electrical currents in the brain by recording active electrical impulses using several electrodes. The EEG's measurement of voltage fluctuations (signals) is assumed to be similar to nervous system activity. It is feasible to discover and characterize brain dysfunctions by comparing the EEGs of multiple patients. EEG also provides information about the location of aberrant brain areas, thus it's utilized to figure out what kind of seizure you're having.

2 Related Work

There have been previous epileptic identification experiments employing a pure EEG dataset in the past. Support vector machine (SVM) was the most commonly used classifier, and the dataset was the CHB-MIT database. Tweenty years later, in 1990s, ANN classification system SEN of 76% and FPR of 1 event/h [1]. In recent years, many research have taken many new approaches about the EEG spike detection: Wavelet transform followed by SVM classification EEG SEN of 99.1% and PPV of 94.8% [2], Used fuzzy algorithms for feature extraction for classification SEN of 95.8% [3].

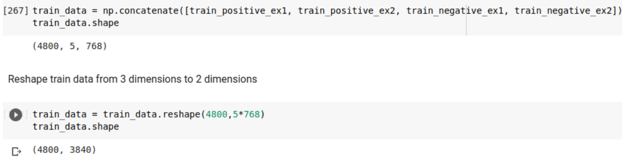
We develope two methodS in two parts of project that can accurately recognize if a sequence represents a sequence of epileptic spikes or a normal EEG sequence.

3 Solution

The project's key contribution is an EEG-based algorithm for determining whether a sample is an epileptic spike or not with EEG.

In phase 1, we code on jupyter notebook and get data from file directly. With phase 2, Pyspark is used in model training and we code on Google Colab.

With four files from two experiments 1 and 2, we concatenate to training dataset.



```
[267] train_data = np.concatenate([train_positive_ex1, train_positive_ex2, train_negative_ex1, train_negative_ex2])
      train_data.shape

      (4800, 5, 768)
```

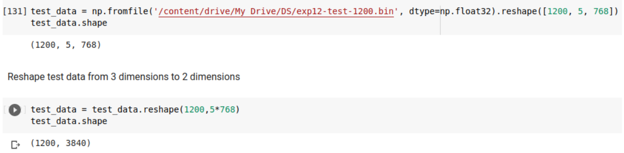
Reshape train data from 3 dimensions to 2 dimensions

```
      train_data = train_data.reshape(4800,5*768)
      train_data.shape

      (4800, 3840)
```

Figure 1: Concatenate dataset.

Training dataset and test dataset are reshaped from 3d dimensions to 2d dimensions.



```
[131] test_data = np.fromfile('/content/drive/My Drive/DS/exp12-test-1200.bin', dtype=np.float32).reshape([1200, 5, 768])
      test_data.shape

      (1200, 5, 768)
```

Reshape test data from 3 dimensions to 2 dimensions

```
      test_data = test_data.reshape(1200,5*768)
      test_data.shape

      (1200, 3840)
```

Figure 2: Reshape dataset.

In step of data processing, we use RandomOverSampler, SMOTE for resampling dataset. And the classifiers will be used for training dataset are Random Forest Classifier, ExtraTreesClassifier and SVC.

Random Forest Classifier

Random Forest Classifier is a well-known and successful machine learning technique for ensemble learning. It's a popular tool for solving classification and regression predictive modeling issues with structured (tabular) data sets, such as data from a spreadsheet or database table. Random Forest can also be used to forecast time series.

ExtraTreesClassifier

This class implements a meta estimator that employs averaging to increase predicted accuracy and control over-fitting by fitting a number of randomized decision trees (a.k.a. extra-trees) on various sub-samples of the dataset.

SVC (Support Vector Classification)

By increasing the distance between sample points and the hyperplane, this classifier seeks to discover the optimal hyperplane for separating the different classes.

4 Experiments

4.1 Hardware

In the first step, we use laptop with processor - AMD Ryzen 7 4800H with Radeon Graphics 2.90 GHz and RAM - 16.0 GB.

In the second step, hardware has informations; Intel(R) Core(TM) i7-1065G7 CPU @ 1.30GHz 1.50 GHz and RAM - 16.0 GB.

4.2 Dataset

The data used in this project varies by step. In the first step, we have access to 400 epileptic spike data series (positive instances) and 2000 normal data series (negative instances) from experiment 1. There are 5x768 sample points in each data series.

In the second step, we get more 400 epileptic spike data series (positive instances) and 2000 normal data series (negative instances) from experiment 2 in addition to the 400 positives and 2000 negatives which we already have from step 1. In this step we have so 800 epileptic spike data series (positive instances) and 4000 normal data series (negative instances) from both of experiments.

A test set of 100 epileptic spike data series and 500 normal data series from both experiments 1 and 2 is provided, for a total of 200 positive and 1000 negative series for both step 1 and 2.

We can see examples of positive dans negative series from the first experiment with Fig.3 and Fig.4

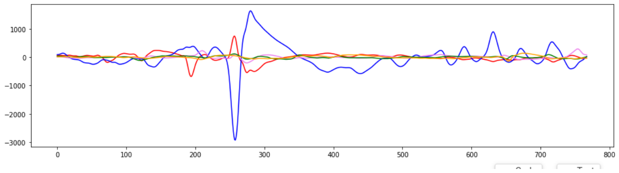


Figure 3: Example of positive serie.

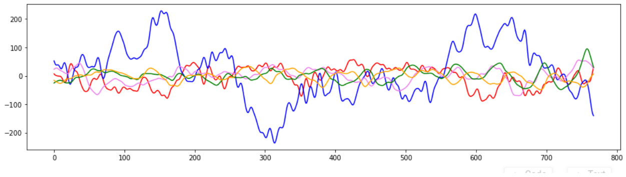


Figure 4: Example of negative serie.

The dataset in this project is time series data which known as a set of observations gathered through repeated measurements over time.

4.3 Step 1

4.3.1 Data Preprocessing

In the first step of project, after read data from two file of 400 positive and 2000 negative series, we concatenate two array to get the train dataset with 2400 series.

The test dataset is provided with 1200 series to test with model..

The function numpy.vstack in Python is used to transform train dataset and test dataset from arrays with 3 dimensions to arrays with 2 dimensions.

4.3.2 Training and Evaluation

For training model, Logistic Regression, SVM, Decision Tree and Random Forest Classifier are tested but we use the last one for the final submission with a score: 0.95083 in Kaggle.

4.4 Step 2

4.4.1 Imbalanced Data

With the second step of project, compared to step 1, we have more data from experiment 2 and the method in step 1 is no longer relevant. In this step we also focus on solving the problem of imbalanced data.

According the Fig.5, we can clearly see that there is a difference between the data set 400 non-epileptic spike and 800 epileptic spike series. There is clearly a class imbalace problem.

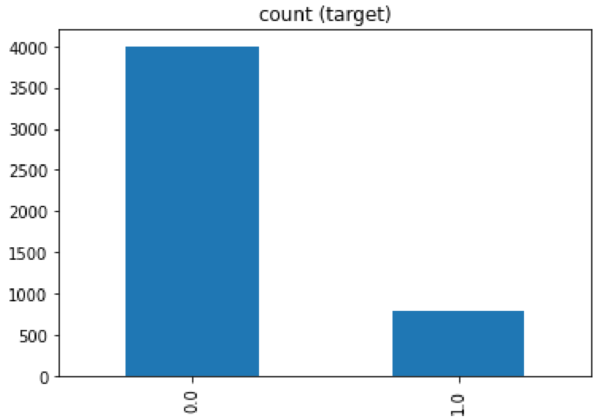


Figure 5: An imbalanced dataset.

4.4.2 Data Preprocessing

Random Over-Sampling

For resampling the dataset, we test some Resampling Techniques using sklearn like Random Over-Sampling like the Fig.6

Oversampling is defined as increasing the number of copies in the minority class.

Table 1 shows the count for both non-epileptic spike and 800 epileptic spike samples when oversampling is performed on train dataset.

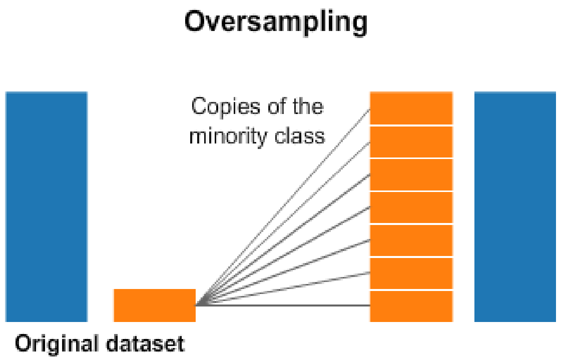


Figure 6: Oversampling.

Category	Original	Resampling
positive(1)	800	4000
negative(0)	4000	4000

Table 1: Number of positive and negative samples after data resampling.

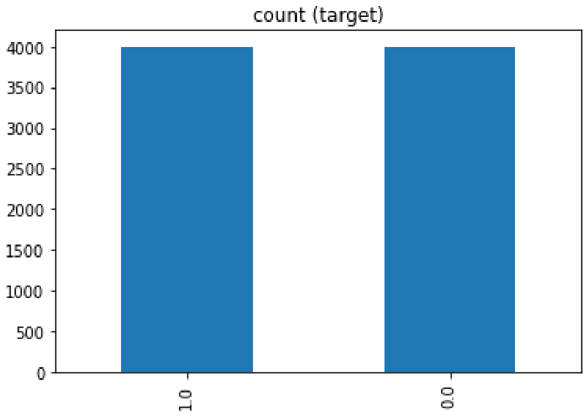


Figure 7: Train dataset after resampling.

Random Under-Sampling

The term ”undersampling” refers to the practice of excluding some observations from the majority class. When you have a lot of data, such as millions of rows, undersampling can be a viable option.

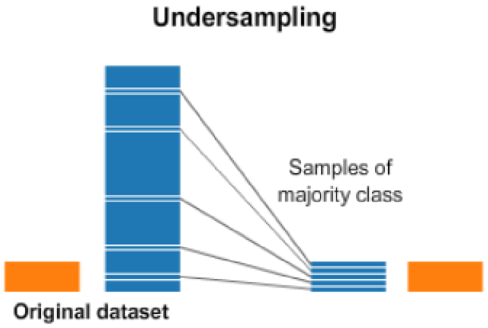


Figure 8: Undersampling.

Table 2 shows the count for both non-epileptic spike and 800 epileptic spike samples when undersampling is performed on train dataset.

Category	Original	Resampling
positive(1)	800	800
negative(0)	4000	800

Table 2: Number of positive and negative samples after data resampling.

SMOTE

Another technique which we use is SMOTE (Synthetic Minority Oversampling TEchnique) is a technique for creating elements for the minority class based on existing elements. It operates by picking a point from the minority class at random and computing its k-nearest neighbors. Between the specified point and its neighbors, synthetic points are added.

Table 2 shows the count for both non-epileptic spike and 800 epileptic spike samples when resampling by SMOTE is performed on train

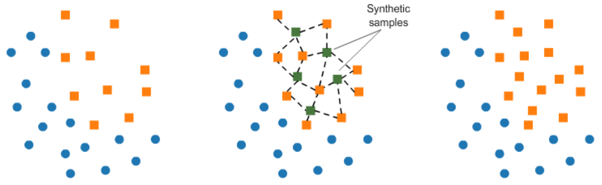


Figure 9: SMOTE.

dataset.

Category	Original	SMOTE
positive(1)	800	4000
negative(0)	4000	4000

Table 3: Number of positive and negative samples after SMOTE.

ClusterCentroids

A method for undersampling the majority class by replacing the cluster centroid of a KMeans algorithm with a cluster of majority samples. By fitting the KMeans algorithm with N clusters to the majority class and using the coordinates of the N cluster centroids as the new majority samples, this algorithm preserves N majority samples.

ADASYN

This method is similar to SMOTE, but it creates a different number of samples based on an estimate of the oversampled class's local distribution.

Principal component analysis

Data dimensionality reduction in machine learning is the process of minimizing the number of data representation features. This can be done in the direction of selecting important features or extracting new features from existing features. Data dimensionality reduction is useful in situations such as visu- alization, storage, and limited computing power. One of popular methods for the data dimensionality reduction is the Principal Component Analysis (PCA) method.The basic idea of PCA is to generate independent new features that are linear combinations of old features. The new features define a projection of the data onto a subspace such that the distance between the projection and the original data is minimized. In other words, PCA searches for the best linear space to approximate the data through its projection.

With the Fig.10, we can chose number of components is 500 for for dimensionality reduction. With PCA, the execution time of model is faster compared to PCA unprocessed data.

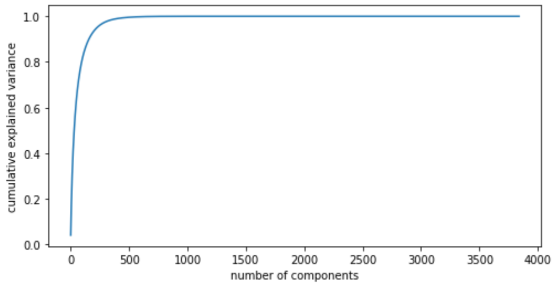


Figure 10: PCA.

4.4.3 Training and Evaluation

We will use Algorithms Based on Trees. The use of tree-based algorithms is the final strategy of the project. Because of their hierarchical structure, decision trees may learn signals from both classes and perform well on unbalanced datasets.

Tree ensembles (Random Forests, Gradient Boosted Trees, and so on) almost always beat individual decision trees in modern applied machine learning.

RF: Random Forest Classifier

SVC: Support Vector Classifier

ETC: ExtraTreesClassifier

RUS: RandomUnderSampler

ROS: RandomOverSampler

CC: ClusterCentroids

Model 1: RF + SMOTE + ADASYN + RUS
Model 2: ETC + SMOTE + ADASYN + RUS
Model 3: RF + SMOTE + ADASYN + CC
Model 4: SVC + ROS

Without PCA, we have the result in table 4.

Model	Score	Execution Time (seconds)
Model 1	0.96166	122.84s
Model 2	0.96000	47.13s
Model 3	0.95666	130.97s
Model 4	0.94750	59.44s

Table 4: Comparaison training models.

With PCA, we can reduce execution time for each model but the score is not good. For example, with Model 1, we have score 0.78083 in 5.3 seconds- 24 times faster than execution time model without PCA.

5 Conclusion

Our approach generalizes several previous techniques that can handle imbalanced data for EEG Spike Detection. The ability to detect and forecast illnesses automatically opens up new possibilities for diagnostic and preventative health treatment.

References

[1] W.R. Webber, R.P. Lesser, R.T. Richardson, K. Wilson - An approach to seizure detection using an artificial neural network (ANN).

[2] E.B. Petersen, J. Duun-Henriksen, A. Mazzaretto, T.W. Kjaer, C.E. Thomsen, H.B. Sorensen - Generic single-channel detection of absence seizur

[3] A.F. Rabbi, R. Fazel-Rezai - A fuzzy logic system for seizure onset detection in intracranial EEG