Seizure Prediction, Kaggle Competition

Capstone Proposal

Kyler Connelly

October 27, 2016

Domain Background

From the Kaggle competition summary which I feel is a very good introduction:

> Epilepsy afflicts nearly 1% of the world's population, and is characterized by the occurrence of spontaneous seizures. For many patients, anticonvulsant medications can be given at sufficiently high doses to prevent seizures, but patients frequently suffer side effects. For 20-40% of patients with epilepsy, medications are not effective. Even after surgical removal of epilepsy, many patients continue to experience spontaneous seizures. Despite the fact that seizures occur infrequently, patients with epilepsy experience persistent anxiety due to the possibility of a seizure occurring.
>
> Seizure forecasting systems have the potential to help patients with epilepsy lead more normal lives. In order for electrical brain activity (EEG) based seizure forecasting systems to work effectively, computational algorithms must reliably identify periods of increased probability of seizure occurrence. If these seizure-permissive brain states can be identified, devices designed to warn patients of impending seizures would be possible. Patients could avoid potentially dangerous activities like driving or swimming, and medications could be administered only when needed to prevent impending seizures, reducing overall side effects.

This is the second Kaggle competition regarding predicting seizures. The first competition used iEEG data from a dog. The results of this competition suggest it is likely possible to predict, with fairly good accuracy, seizures in humans using FFT data from the iEEG. See reference below.

- Brinkmann, B. H., Wagenaar, J., Abbot, D., Adkins, P., Bosshard, S. C., Chen, M., ... & Pardo, J. (2016). Crowdsourcing reproducible seizure forecasting in human and canine epilepsy. Brain, 139(6), 1713-1722. http://brain.oxfordjournals.org/content/139/6/1713

Starting this machine learning program I was and still am interested in machine learning applications in the medical space. I am particularly interested in applications which help reduce the cost of healthcare since for most people on the planet cost determines access. In this case a seizure forecasting device could potentially reduce the amount of medication a user would otherwise require, along with reducing the side effects of taking the medication which could cause the need for other medication or medical procedures. In addition to reducing medication, a device such as this has the potential to reduce the number of hospital visits associated with seizure related accidents or injuries.

This information was found here:
https://www.kaggle.com/c/melbourne-university-seizure-prediction

Problem Statement

The problem to be solved is to be able to use EEG data from patients to be able to as accurately as possible determine if in the near future a seizure to likely to occur or not. To be relevant in this medical environment the model needs to make the prediction between 5 and 65 minutes before the onset of the seizure.

Datasets and Inputs

The dataset is the the same data used for the Kaggle competition. This is a very large dataset, many GB per patient, and there are three patients. There is an additional set of patient data which is referenced on the Kaggle site.

The data is raw voltage readings from the patient electrodes. There are 16 electrode readings at all times. The data is broken up into files which represent 10 minutes of data from the electrodes. For most of the data is it indicated whether or not a seizure occurred.

Since this is such a large set of data, even if I were to take a sample of it, I will likely do a lot of pre-processing to the data to compact it into a form that would actually get fed into the learners. Possible products of the data to use with the learners would be FFT for each segment, calculated for some number of FFT bins, then maybe take the average for each bin. If the data is still too large to practically design with then I may take a random sample of the data.

Solution Statement

One possible solution would be to train a decision tree. I am not sure how well this would work since I have not tried it yet, but it is possible it could produce an acceptable result. The model will be trained with the FFT data plus whether or not these particular instances lead up to a seizure. The model will then be tested on the test dataset. The data set on Kaggle has already been split into training and testing data. However the test set does not indicate whether or not a seizure occurred, that is the point of the competition. Therefore I will randomly split just the training data and withhold the labels for the test portion of the split. The success of the model will be measured by applying the model to the test set and seeing how accurate it is based on the known label.

Benchmark Model

Looking at the leaderboard of the Kaggle competition, the highest score is about 90% and the lowest is about 40%. I will aim to get above 70% accuracy for this project, that seems like a reasonable goal. I do need to consider that the scores on Kaggle are on their test set which contains 50/50 for seizure and non-seizure instances. My test set will be a subset of the training set which is not the same ratio. Because of this I may be more likely to get a higher or lower score, but I won't know until the project is underway.

Evaluation Metrics

The model will be evaluated by applying the same evaluation metric which is used for the Kaggle competition, Receiver Operating Characteristic (ROC). ROC evaluation creates a curve comparing the true positive rate as a function of the false positive rate. The area under the curve is taken where higher is better and the maximum score is 1.0. I will us sklearn to calculate this value.

Project Design

I will break the project into several parts: Initial Exploration, Preprocessing (FFT), Exploration, Outlier Removal, Model Training, Model Tuning, and Final Evaluation. The entire project will be done in python 2.7 and using the standard libraries included in Anaconda.

Initial Exploration: In this section I will explore the data to get an idea of how it should be processed. I will look at things like sampling frequency to determine how to define my FFT bins. I will also look at signal magnitude to get an idea of what is normal so that I can remove outliers (abnormally high signals or no signal at all).

Preprocessing: Here I will preprocess the data by taking the FFT for each 10 (this may change) minute segment. The FFT will be divided into some number of bins. For each bin the average will be taken. The set of averages from this will be the actual input data.

Exploration: With the data in the form which will be used by the models, I will explore the data some more now. From what I see I may device to reprocess the data with different FFT bins, or device to use some other metric all together.

Outlier Removal: Most of the outlier removal would have probably already been performed in the time domain exploration. However, I will consider all the data again here looking for outliers.

Model Training: The pre-processed and filtered data will be fitted to a variety of learning algorithms in sklearn. Each of the learners will be tested on the test set. Some basic adjustments will be done to see a variety of results for each learner.

Model Tuning: Here a more complete and thorough tuning will be performed on the model which got the best score in the previous step. If multiple model performed nearly equally as well in the previous step, I may tune multiple in this step.

Final Evaluation: The fully tuned model(s) will be applied to the test set and a score will be produced.