Predicting seizures in intracraneal EEG recordings

February 12, 2015

Modelling Workshop

Quim Llimona
Enrico Kunz
Josep Mercadal
Samuel Rodriguez
Àngel Farguell

Modelling for Science and Engineering

Universitat Autònoma de Barcelona

Predicting seizures in intracraneal EEG recordings

Q. Llimona, E. Kunz, J. Mercadal, S. Rodriguez, A. Farguell

1 Abstract

The aim of this project is to uncover hidden patterns in human epilepsy that could lead to new insights about the prediction of seizure attacks by analyzing raw EEG recordings. Given the nature of the problem (unknown pattern to look for, huge datasets and a vast solution space to explore) we use machine learning algorithms as first approach towards a more precise end result. Working on the first part of our solution we applied support vector machines (SVM) to our problem, which is the subject of this work. While the use of SVM is not new to the field, our way of fine tuning it does make a difference to previous work. We propose the use of Kriging methods to build a predictive n-dimensional accuracy function from a very low number of real samples in order to find a near-optimal combination of parameters for the SVM in a short period of time.

2 Introduction

Epilepsy affects a lot of people in the world, and this is concerning; having a sudden seizure can be catastrophic, especially if the subject is driving or performing any kind of hazardous activity. That makes it s a very disabling pathology, and those who suffer from it need a lot of care due to the fact that there are no visible symptoms right before a seizure strikes.

Patients have traditionally taken lots of medication, effectively being constantly under drugs in order to minimize the likelihood of a sudden seizure. However, this can be too aggressive for the body in the long run, and the brain will adapt to the medication faster and faster, needing larger doses or new compounds each time. It becomes clear then that having a method for predicting when a seizure is about to take place and thus being able to administer drugs only when needed would be very beneficial for those suffering the pathology.

The American Epilepsy Society is well aware of that, and it called a competition on the Kaggle platform offering a substantial reward [1] for systems that could predict the onset of an epileptic seizure given electroencefalogram (EEG) data from before the seizure. While there is extensive literature about the matter, there is no known method that works accurately enough for any patient [2]. This paper is our attempt at it, although we were out of the competition from the beginning due to time constraints.

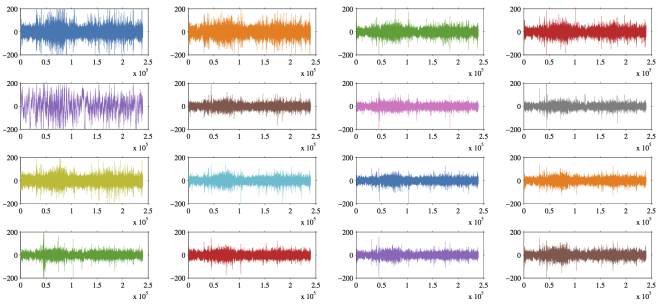

Figure 1: EEG recordings of 16 channels taken from one of our human datasets

The EEG recordings are temporal series that represent electric pulses recorded by electrodes within the brain. Therefore, our goal is to learn through statistic techniques the difference between series corresponding to the moments before an outbreak (preictal recordings) and those corresponding to regular activity (interictal recordings). We can see an example of EEG recordings in Figure 1, displaying 16 channels corresponding to 16 distinct electrodes in the brain.

The American Epilepsy Society provides several datasets for analysis, grouped in recordings from 5 distinct dogs and 2 different humans. We began studying the dogs because of its wider availability and found literature [3], but ended up testing the algorithm with human data because that is the ultimate goal of the project. Each human data set has 16 channels in different parts of the brain for each recording. For each human, around 20 preictal recordings (1 hour before the seizure), 50 interictal recordings (half an hour before), and 150 unlabeled test recordings are provided. We were left, therefore, with very few preictal recordings, which made the development of an algorithm that would do reliable predictions complicated.

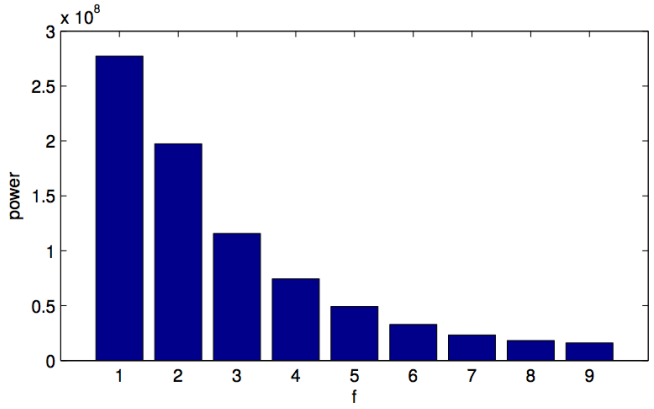

Figure 2: Energy distribution of an EEG signal across different frequency bands.

3 Prediction pipeline

The pipeline of our system can be summarized as follows: a continuous stream of multi-channel EEG data comes in, it is split in short windows a few seconds long, for each window a prediction of whether it anticipates a seizure or not is made, and the result is combined with a fixed number of the previous windows to give a prediction.

We will first discuss how the signal is prepared within each window and how the within-window prediction is made, and finally how to combine multiple windows.

Our system, inspired in what was presented in [4], relies on the Support Vector Machine (SVM) [5] method to classify individual frames of data. This algorithm learns how to classify multi-dimensional vectors into one of two categories. Therefore, we devised a method to convert a chunk of a multi-dimensional signal into a meaningful vector that lies in a space where pre-seizure and inter-seizure signals are in different regions.

3.1 Feature extraction

The goal of the feature extraction step is, as mentioned above, to convert a discrete-time signal into a single feature vector so that the SVM can classify it. Once the signal is cut into chunks of a predefined length (typically between 1 and 10 seconds long), we apply the following steps:

1. A Finite Impulse Response filter is applied to the signal in order to normalize frequency content across the spectrum, of the form: $y[n] = x[n] + x[n-1]$. As seen in Figure 2, the signal has originally more energy in low frequency regions.

2. The Fourier Transform is applied to the signal, using the FFT method.

3. The spectrum is split in equally spaced regions, and the sum of the energy of all bins that lie in the same band is computed.

The reason behind using the frequency space is that neurons communicate and operate through the frequency at which they emit electric spikes, not through their intensity; therefore, the most meaningful information we can get from the data is at which frequencies the neuron groups near each electrode are operating at each time instant.

3.2 Learning model

As stated before, the feature vectors constructed by integrating energy over different frequency bands are fed into an SVM algorithm for classification. Support Vector Machines are very well-known in the Machine Learning field, and have become an industry standard over the years thanks to their simplicity, performance and efficiency, as well as the flexibility they provide for incorporating domain-specific knowledge through feature engineering (what we just described above).

The original SVM proposal finds a hyperplane in the feature space such that points belonging to the same class are on the same side of hyperplane (negative or positive), while maximizing the distance to the hyperplane for those vectors closest to it. However, it is easy to extend the algorithm to learn optimal non-linear separating surfaces through the kernel trick. SVM implementations typically provide linear, polynomial and radial (gaussian) surfaces, which are parametrized with a series of coefficients that the user must choose.

3.3 Post-processing

The presented system so far outputs a binary prediction (imminent seizure or resting state) for each frame or window of data. Since we are interested in detecting events operating at larger time scales than a few seconds, we join the windows in groups in order

to get an average and smooth out occasional noise. For the evaluation, since we were asked whether complete 10-minute recording was pre-seizure or not we merged predictions from all 10 minutes together.

The aggregation method we selected is the following: the output is converted to a numerical representation (0 or 1), the average is taken over all windows, and finally the result is binarized again by comparing against a given threshold. This is equivalent to predicting an imminent seizure if more than the percentage of windows selected by the threshold predict so.

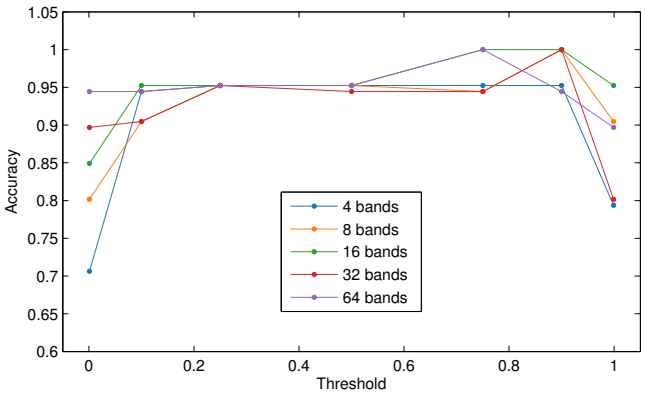

Figure 3: Performance of the system for different number of frequency bands being fed into the classifier, in a selected threshold vs accuracy plot.

4 Preliminary evaluation

In this section, we present a basic pre-evaluation of the system. All reported results have been obtained though cross-validation on our dataset, experimenting independently on different patients, and using the same number of positive and negative samples to avoid having to compensate for the bias.

4.1 Sensitivity and specificity

In every classification system there is a trade-off between sensitivity and specificity, that is, how many positive samples the system detects among those on the data versus how many positive tagged samples are actually positive. In an extreme case, one might output always a positive outcome, and all positive samples would be identified (maximum sensitivity), although lots of samples tagged as positive would be wrong (low specificity). Although it is common to maximize the two of them at the same time by choosing the configuration that yields the maximum number of classified samples, both positive and negative, it can be interesting to lend towards one of them depending on the needs of the problem being addressed.

In our particular case, it may be the case that the patient receives an important drug dose when the system predicts an imminent seizure in order to avoid it. These drugs cannot be too aggressive, because epilepsy patients normally take them regularly in order to minimize the number of seizures. Therefore, it is better to take drugs more times than needed as a response to a false positive than to avoid it and suffer a seizure because of a false negative.

However, establishing the correct balance is beyond the scope of this work, and it should be done with supervision of medical doctors, who will know the acceptable dosages of commonly used drugs - it may even be the case that this must be adjusted per patient, or event dynamically depending on the frequency of seizures. Therefore, we will perform our analysis taking into account the accuracy of the system, that is, the total portion of correctly identified samples - both positive and negative.

As introduced in Section 3, our system provides a threshold parameter for selecting how easy it is to classify a sample as positive. As Figure 3 shows, for extreme values of the threshold the performance is worse; actually, a low threshold corresponds to classifying too many samples as positive, because a single positive frame will already trigger it, and viceversa.

4.2 Meta-parameter space

Besides the effect of thresholds, Figure 3 also shows that the performance of the system varies significantly when the number of bands in which the frequency space is grouped for integration when constructing the feature vectors changes. In that particular case, 16 bands seems to yield the best results.

The number of bands in which the frequency space is divided is not the only parameter of the system that can be manually adjusted: the window length, the overlap between windows (which defines the total amount of data per training example being fed to the system), the SVM kernel function, or the parameters

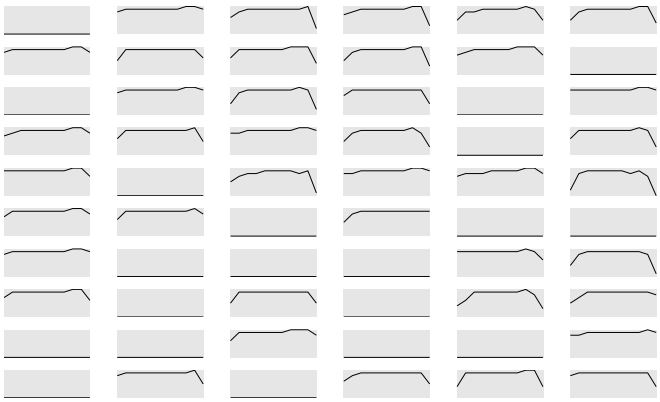

Figure 4: Prediction accuracy for different parameter combinations. The horizontal axis represents the selected threshold, the vertical axis is accuracy.

to that kernel function are just some examples.

Figure 4 presents a grid with the accuracy of the system for randomly sampled combinations of parameters. Indeed, some combinations perform very good (very high accuracy in medium threshold regions), and others perform incredibly bad (no better than chance). Inspection revealed that in most of those bad cases the system was actually degenerate and was always predicting the same outcome, be it positive or negative. Next section presents an approach for choosing the right combination.

5 Parameter optimization

Several approximate optimization methods have been studied to solve the SVM tuning problem such as, the response surface method (RSM) with optimization of coefficients for a base function, the neural network approximation (NN), heuristic algorithms (Genetic algorithms, Simulated annealing, etc). Generally, model building methods are preferred over heuristic and gradient-based ones because of the large number of SVM training and testing iterations needed to test every possible point.

The Kriging model drastically reduces the computational time required for objective function evaluation in the optimization (optimum searching) process. Computational cost of the Kriging method to determine the estimation model is not too high either. However, in order to estimate a function value for each location will be higher than NN or RSM for a larger number of input variables. In our case, the computational cost of the calculation of each sample is about two orders of magnitude greater than the model generation, so we chose the Kriging model as the one with best trade-off of samples - results.

5.1 Kriging model

The Kriging model expresses the unknown function $y(x)$ as:

$$y(V) = \mu + Z(V)$$

where V is a $m-$dimensional vector, μ is a constant global model and $Z(V)$ represents the local deviation from the global model. In the model, the local deviation at an unknown point V is expressed using stochastic processes. The sample points are interpolated with the Gaussian random function like the correlation function to estimate the trend of the stochastic processes. The correlation between $Z(V^i)$ and $Z(V^j)$ is based on the distance between V^i and V^j but, instead of using Euclidean distance, Kriging model uses a special weighted distance, expressed as:

$$d(V^i, V^j) = \sum_{k=1}^{m} \Theta_k \mid V_k^i - V_k^j \mid^2$$

where $\Theta_k (0 \leq \Theta_k \leq \infty)$ is the k_{th} element of the correlation vector parameter Θ. By using the Gaussian random function and the distance function, the correlation (C) between two points is defined as:

$$C\left[Z(V^i), Z(V^j)\right] = exp[-d(V^i, V^j)]$$

The Kriging predictor is:

$$\hat{y}(V) = \hat{\mu} + r' R^{-1}(y - \hat{\mu})$$

where $\hat{\mu}$ is the estimated value of μ, R denotes the nxn matrix whose (i, j) entry is $C[Z(V^i), Z(V^j)]$. r is a vector whose $i_t h$ element is defines as:

$$r_i(V) = C[Z(V), Z(V^i)]$$

and $y = [y(V^1), ..., y(V^n)]$. The detailed derivation of the predictor can be found in [6]. The unknown parameter to be estimated for constructing the Kriging model is Θ. This parameter can be estimated by maximizing the following likelihood function:

$$L_n(\hat{\mu}, \hat{\sigma}^2, \Theta) =$$
$$= -\frac{n}{2} ln(2\pi) - -\frac{n}{2} ln(\hat{\sigma}^2) - \frac{1}{2} ln\left(\mid R \mid\right) - \frac{1}{2\hat{\sigma}^2}(y - \hat{\mu})' R^{-1}(y - \hat{\mu})$$

Maximizing the likelihood function is an $m-$dimensional unconstrained non-linear optimization problem.

The accuracy of the predicted value largely depends on the distance from sample points; the closer point V to sample points, the more accurate is the prediction $\hat{y}(V)$. This is expressed in the equation:

$$s^2(V) = \hat{\sigma}^2 \left[1 - r' R^{-1} r + \frac{(1 - R^{-1} r)^2}{R^{-1}}\right]$$

5.2 SVM tuning

Using a reduced number of V samples homogeneously distributed, we evaluated the accuracy of the SVM on the points and then we constructed the Kriging model. Once the model was constructed we found the maximum and test the SVM again on that point.

Although we hit the 100% accuracy mark, there was a big iso with the surface on the model at 100% expected accuracy value, making the impressive results useless. The problem was that even using resampling or random forest techniques we could not overcome the homogeneity of the input data. The conclusion was that our SVM algorithm was over trained due to the small training data.

6 Final results and remarks

The SVM algorithm has been proposed and applied to predict seizures on intracraneal EEG recordings. Given the degrees of freedom of the model, the authors proposed a methodology based on Kriging sampling methods to optimize the input parameters. Although the Krigin method modeled well the expected behavior of the SVM, the dataset in use was not heterogeneous enough to train the SVM effectively.

Although the use of machine learning algorithms is not new on classification problems, the use of the Kriging method is the novel contribution of the optimization of the SVM kernel.

References

[1] Some Authors. Kaggle competition from american epilepsy society seizure prediction challenge. `https://www.kaggle.com/c/seizure-prediction`, 2014. Last time we have seen 06-02-2015.

[2] Florian Mormann, Ralph G. Andrzejak, Christian E. Elger, and Klaus Lehnertz. Seizure prediction: the long and winding road. *Oxford journals: Brain*, I 30:314–333, 2007.

[3] J Jeffry Howbert, Edward E Patterson, S Matt Stead, Ben Brinkmann, Vincent Vasoli, Daniel Crepeau, Charles H Vite, Beverly Sturges, Vanessa Ruedebusch, Jaideep Mavoori, et al. Forecasting seizures in dogs with naturally occurring epilepsy. *PloS one*, 9(1):e81920, 2014.

[4] Yun Park, Lan Luo, Keshab K Parhi, and Theoden Netoff. Seizure prediction with spectral power of eeg using cost-sensitive support vector machines. *Epilepsia*, 52(10):1761–1770, 2011.

[5] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge University Press, New York, 2003.

[6] J. Koehler and A. Owen. *Handbook of statistics*. Elsevier, Amsterdam, 1996.