**Seizure Detection and Brain Machine Interface Decoding with Neural Networks**

Dan Jaklic, University of Michigan

April 1, 2021

**Introduction**

Epilepsy is a debilitating neural condition, marked by recurrent seizures, that currently affects about 50 million people worldwide. 25% of these patients cannot be treated with current medications or medical technology, with mortality very high in uncontrolled patients. Biomedical technologies such as electrical stimulation can abort seizures and have proven to be safe and effective. However, the ability to abort seizures is not so helpful if seizures can’t be predicted or detected. Seizure detection attempts have been made by placing intracranial electrode matrices (ECoG) on the surface of the brain and processing neural data with signal processing algorithms that decode signal features such as spectral power, spikes, zero crossings, Teager Kaiser Energy, among others. In this study, we attempt to detect seizures from patient ECoG data using “signal amplitude”, “zero crossings” and “line length” ECoG features. Next, we attempt seizure detection using Linear Discriminant Analysis (LDA). Finally, we attempt seizure detection using “high-frequency oscillations” (HFO), a feature seen by extracting 100-600 Hz frequencies from patient ECoG data.

An artificial neural network (ANN) is a machine learning (ML) technique that has captured the imagination of computational scientists since Farley and Clark used computers to simulate a Hebbian network in 1954 [1]. ANN’s were not very useful for any applications until algorithmic advances such as backpropagation [2] and technology advances such as Graphical Processing Units (GPU) appeared in the 1980’s and 1990’s. Still, neural decoding largely depended on support vector machines (SVM), or more recently, linear regression and Kalman filtering. The performance of ANN’s have improved such that in the last ten years, they became viable neural signal decoders. Previously, we used continuous decoding methods such as linear regression, ridge regression, LASSO, and Kalman filtering to predict hand positions and velocities from firing rates of 95 recorded neurons. In this study, we implement a simple, feed-forward neural network to decode the same data set with the goal of comparing results to previous methods.

**Methods**

1. Data Acquisition and Assumptions

From two patients (Patient A, Patient B), three channels of ECoG data sampled at 3 kHz (1800 seconds total) was recorded and saved to MAT files (MATLAB, Natick MA). ECoG arrays are assumed to be 360 channels and spaced at 1 cm2. ECoG arrays across patients are assumed to be placed on corresponding anatomical brain regions (individual differences cannot be completely avoided). In this study, all seizure detection algorithms were trained on patient A and tested on patient B. All detection algorithms were written manually in MATLAB.

(ANN data)

1. Seizure Detection Using Common Features

Three features prominently seen in EEG or ECoG recordings were used to attempt seizure detection: signal amplitude (SA), zero crossings (ZC) and line length (LL). For all algorithms, EcoG signals were smoothed and analyzed over 10 second, non-overlapping windows.

* 1. Signal Amplitude Detection

For each channel, the root mean square (RMS) was taken over the whole sample time course to establish a patient and channel specific baseline. Loosely speaking, this gives an average power of the signal. Then the RMS was taken for each smoothed, 10 second window. Values were normalized using the average power value described, giving each window (180 windows over the sample) a normalized power value. Using patient A for training, a threshold was determined to classify ‘seizure’ or ‘no seizure.’ Then, testing was carried out on patient B.

2.2 Zero Crossing Detection

For each channel, the total number of ZCs (crossing the x-axis at voltage = zero) over the whole sample time course was determined using MATLAB’s Zero Crossing Detector from the Digital Signal Processing library. Using this, an average number of ZCs per window was calculated. Next, the actual number of ZCs for each smoothed, 10 second window was calculated with the same method. A normalized value was calculated. Using patient A for training, a threshold was determined to classify ‘seizure’ or ‘no seizure.’ Then, testing was carried out on patient B.

* 1. Line Length Detection

Calculating ECoG signal line lengths can be useful because a longer line length within a time window tends to correspond with higher frequencies and larger amplitudes, markers of seizure. For each channel, the total LL was calculated in MATLAB [sum(abs(diff(signal)))]. Then, average LL per window was calculated. Next, the actual LL each smoothed, 10 second window was calculated with the same method. Values were normalized. Using patient A for training, a threshold was determined to classify ‘seizure’ or ‘no seizure.’ Then, testing was carried out on patient B.

All features were combined into a single seizure detection algorithm. Binary results from each corresponding window in time were compared using ‘AND’ logic – Seizure detected only if indicated all three features, otherwise no seizure detected.

1. Seizure Detection Using Linear Discriminant Analysis

LDA is a well known and broadly used classifier, ideal for two class differentiation (seizure or no seizure). We implemented LDA using ‘classify()’ in MATLAB. To review, patient A was used for training and patient B used for testing.

1. Seizure Detection Using High Frequency Oscillation Detection

()

1. Brain Machine Interface Decoding with an Artificial Neural Network

An ANN was constructed using the open source machine learning library PyTorch in Python. The ANN consisted of an input layer, one hidden layer, and an output layer (all linear layers). Batch normalization [5] was done on the input, hidden, and output layers to provide stable traversal of data through layers. Dropout regularization [6] was done on the input and hidden layers to reduce overfitting during ANN training. The ANN was initialized using a normal kaiming distribution with a rectified linear unit (RELU) nonlinearity. For comparison, the ANN was run a second time using uniform kaiming initialization with all other parameters remaining the same.

**Results**

1. Seizure Detection Using Common Features
   1. Signal Amplitude Detection

The normalized power threshold was manually determined to be 1.35 (> 1.35 = seizure, < 1.35 = no seizure).

1.2 Zero Crossing Detection

The normalized ZC thresholds were manually determined to be: low = 0.76, high = 0.84. Anything outside of this range was marked ‘seizure.’

* 1. Line Length Detection

The normalized LL threshold was manually determined to be 1.1 (> 1.1 = seizure, < 1.1 = no seizure) .

1. Seizure Detection Using Linear Discriminant Analysis

()

1. Seizure Detection Using High Frequency Oscillation Detection

()

1. Brain Machine Interface Decoding with an Artificial Neural Network

()

**Discussion**

Thresholds for each channel?

Less supervision?

**References**

[1] B. Farley and W. Clark, "Simulation of self-organizing systems by digital computer," in *Transactions of the IRE Professional Group on Information Theory*, vol. 4, no. 4, pp. 76-84, September 1954, doi: 10.1109/TIT.1954.1057468.

[2] Rumelhart, D., Hinton, G. & Williams, R. Learning representations by back-propagating errors. *Nature* **323,**533–536 (1986). <https://doi.org/10.1038/323533a0>

[3] Chestek, C and Stacey, W. Lab 10 handout: Seizure Detection. *BIOMEDE 517* 2021

[4] Chestek, C. Lab 11 handout: Artificial Neural Networks. *BIOMEDE 517* 2021

[5] Sergey Ioffe, Christian Szegedy Proceedings of the 32nd International Conference on Machine Learning, PMLR 37:448-456, 2015.

[6] <https://www.connectedpapers.com/main/1366de5bb112746a555e9c0cd00de3ad8628aea8/Improving-neural-networks-by-preventing-coadaptation-of-feature-detectors/graph>

**Appendix A – Figures**

**Timeline

Description automatically generated**

Figure : Common Feature seizure detection results. Any non-zero detection value indicates seizure detection. Channel 1-3 (top to bottom) of patient B ECog.

**A picture containing chart

Description automatically generated**

Figure : Common Feature seizure detection feature breakdown. Any non-zero value indicates intermediate seizure detection for that time bin. Ultimate seizure detection depends on seizure indication of all three features. Channel 1-3 (top to bottom) of patient B.

**Timeline

Description automatically generated**

Figure : LDA seizure detection results. Any non-zero detection value indicates seizure detection. Channel 1-3 (top to bottom) of patient B ECog. (editors note – Had a lot of trouble with this figure in MATLAB for unknown reasons. It practice, the format should be the same as the others)

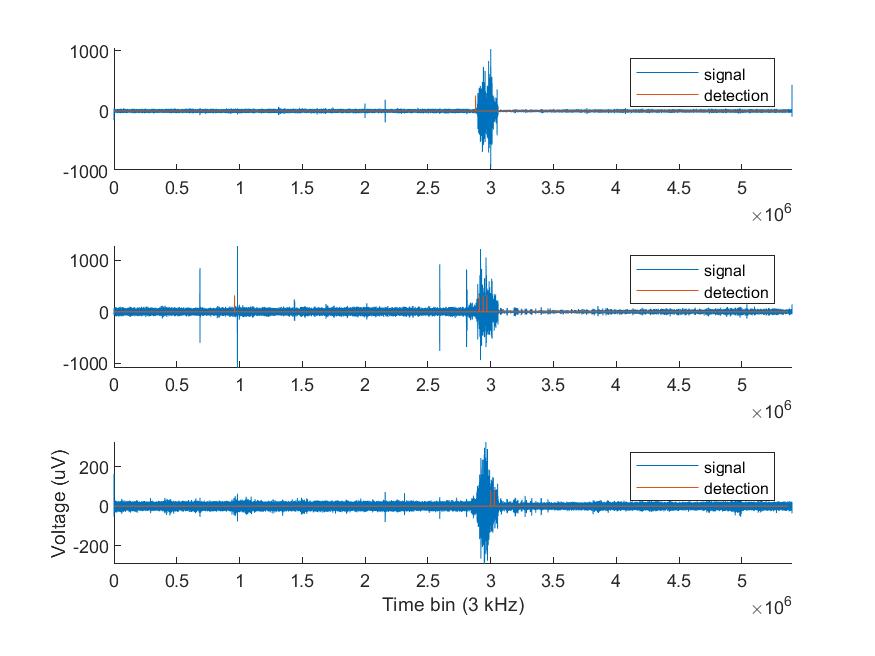
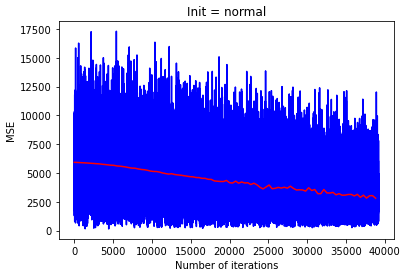
****

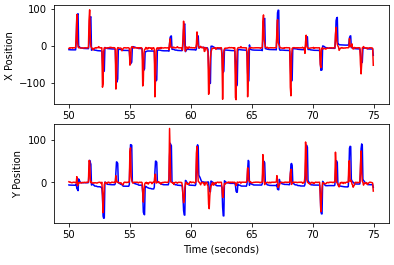
Figure : HFO seizure detection results. Any non-zero detection value indicates seizure detection. Channel 1-3 (top to bottom) of patient B ECog.

**Chart

Description automatically generated**

Figure : HFO seizure detection feature breakdown. Any non-zero value indicates intermediate seizure detection for that time bin. Ultimate seizure detection depends on seizure indication of all three features. Channel 1-3 (top to bottom) of patient B.

****

****

**Appendix B – Tables**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Features | Linear Discriminant Analysis | High Frequency Oscillation |
| **Overall Accuracy** | **97.95%** | **95.16%** | **97.02%** |
| False Positive | 0% | 2.23% | 0.19% |
| True Negative | 2.05% | 2.61% | 2.79% |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Linear Regression** | **Ridge Regression** | **LASSO** | **Kalman** | **ANN**  **‘Normal’** | **ANN**  **‘Uniform’** |
| X position | 0.9141 | 0.9144 | 0.9151 | 0.8623 | 0.6699 | 0.6932 |
| Y position | 0.8859 | 0.8860 | 0.8869 | 0.8175 | 0.6680 | 0.6684 |
| X velocity | 0.8696 | 0.8700 | 0.8701 | 0.8262 | 0.9385 | 0.9473 |
| Y velocity | 0.8388 | 0.8391 | 0.8393 | 0.7899 | 0.9012 | 0.9217 |
| **Total** | **0.8593** | **0.8596** | **0.8493** | **0.8117** | **0.7944** | **0.8077** |