# Predicting Seizure Onset in Epileptic Patients Using Intracranial EEG Recordings

Janet An
Stanford University
janeta94@stanford.edu

Amy Bearman
Stanford University
abearman@stanford.edu

Catherine Dong
Stanford University
cdong@stanford.edu

***Abstract*—Intracranial EEG-based monitoring systems have the potential ability to predict seizure onset, but preictal (pre-seizure) states are difficult to identify and are often confused with normal variations in brain activity. Our goal was to distinguish between EEG recordings covering preictal brain activity an hour prior to a seizure, and those recording purely interictal (non-seizure) activity. We modeled this as a binary classification problem, computing several linear and nonlinear bivariate features (maximal cross-correlation and Euclidean distance) over pairs of EEG channels. We trained support vector machines, logistic regression, and neighbors-based classification to discriminate interictal from preictal patterns of features. Among the evaluated methods, L2-regularized logistic regression was most successful, predicting with 85.7% accuracy and an F1 score of 86.7%.**

***Index Terms*—Seizure prediction, logistic regression, EEG, machine learning.**

## I. Introduction

About 1% of the world's population is afflicted with epilepsy. Although seizures occur infrequently (99% of a patient's life is spent in the interictal (non-seizure state), the possibility of a spontaneous seizure causes constant anxiety for patients and inhibits their ability to lead normal lives. Sufficiently high doses of anticonvulsant medications can be effective in 60-80% of patients, but because they often come with significant side effects, these medications should only be administered when necessary, i.e., before a seizure. Therefore, some method of predicting seizure onset is needed to alert epileptic patients to administer medication and avoid risky activities, such as swimming or driving.

Intracranial electrocorticography (EEG) monitoring systems have the potential ability to predict seizure onset by measuring electrical activity in the brain, as recorded from multiple electrodes implanted directly on the surface of the brain. However, even with this data, preictal (pre-seizure) states are difficult to identify and often confused with normal variations in brain activity.

Because epilepsy is such a prevalent neurological disorder, with unexpected seizures having such dangerous consequences, attempts to develop seizure forecasting systems have been recorded in a number of previous papers. One aspect of the prediction process that all of the researchers have had to handle is selecting relevant features from the enormous datasets of EEG recordings. Different approaches to feature selection that have been undertaken include choosing a small subset of the electrode channels using recursive feature elimination, as well as spectral analysis using wavelet transform to compress EEG recordings into a small number of features. After running their algorithms, mostly using various types of SVMs, the previous papers report varying degrees of success, with accuracies ranging from just barely over 50% to around 85%. Many of these papers report that future work to be done to improve the accuracy of these prediction algorithms include dimensionality reduction of the data and better feature extraction to obtain potentially hidden information.

Our goal is to apply a variety of machine learning classification methods to this EEG data in order to more accurately model the probability that a patient is in a preictal state.

## II. Data

Annotated intracranial EEG (iEEG) data is freely available at the International Epilepsy Electrophysiology Portal. There are iEEG data clips for both human and canine subjects. The data for each subject is organized into ten minute clips labeled "Preictal" for pre-seizure data segments, or "Interictal" for non-seizure data segments. Preictal training and testing data segments are provided covering one hour prior to seizure with a five minute seizure horizon (i.e. 1:05 to 0:05 before seizure onset). Interictal data is chosen randomly from the full data record, with the restriction that the interictal segments be as far away as possible from any seizure to avoid contamination with preictal or postictal signals. The pre-seizure horizon ensures that seizures can be predicted with enough time for the subject to take appropriate action (such as administer medication).

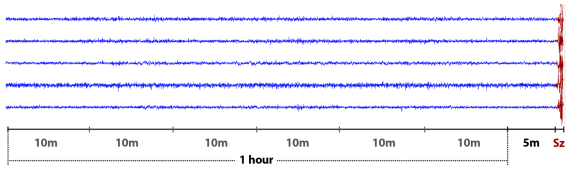



Fig. 1: EEG data over 1 hour pre-seizure window

Each data clip contains a matrix of EEG sample values arranged row $\times$ column as electrode channel $\times$ time. In our subject's data set, there are 15 electrode channels and 240,000 iEEG readings sampled at 400 Hz. The recordings span multiple months up to a year, and record 30 seizures in our subject (up to 100 in other subjects). We used 5 hours of interictal data and 5 hours of preictal data for training. There are 60 data clips (30 preictal, 30 interictal), for 36,000 total examples in both the training and testing data sets.

## III. Preprocessing

### A. Normalization

Since the range of voltage readings in the EEG dataset vary widely, we normalized the data to avoid heavily weighting readings that are very positive or very negative. For each segment, we calculated the minimum voltage reading over all 15 channels, as well as the range of the data for that segment. We normalized each EEG reading, $x$, by computing:

$$x := \frac{x = \min(x)}{|\max(x) - \min(x)|} \tag{1}$$

### B. Independent Components Analysis

We attempted to use ICA to separate the EEG data into maximally independent components, in order to potentially remove noisy artifacts such as eye movement, heartbeat, or muscle activity on the scalp. We extracted the unmixing matrix, $W$, in order to recover the "sources" (i.e., the independent components, including the heartbeat waveform) by computing $s^{(i)} = Wx^{(i)}$. However, we did not recover any obvious artifacts, and running our models on the independent sources did not give us higher performance metrics, so we decided to use the original data. It is possible that ICA was not successful because the EEG sources were not temporally independent, or that there were non-negligible propagation delays from the sources to the electrodes [6].

## IV. Feature extraction from EEG

The challenge was to distinguish between ten minute long data clips covering an hour prior to a seizure, and data clips of the same length recording purely interictal (non-seizure) activity. Predicting seizure onset can be modeled as a binary classification problem, where data clips are labeled as $\pm 1$ (negative for interictal and positive for preictal). Our goal was to construct a function that maps from a feature vector $x$ taken from a data segment to a correct labeling of the segment.

As a baseline, we created a feature matrix where each EEG channel reading was one feature. For more advanced feature selection, we also computed bivariate features over 1 second windows that measure a relationship between two EEG channels, rather than univariate features which are computed on each channel separately. In past studies, bivariate features have been more successful than univariate ones in seizure prediction tasks. Though there is little understanding of the preictal brain state, neuroscience researchers hypothesize that there is correlation between the synchronization of brain activity and seizure onset. This synchronization hypothesis motivated our choice to use bivariate features.

### A. Maximal Cross-Correlation

Cross-correlation is a measure of similarity of two waveforms as a function of a time-lag applied to one of them. Cross-correlation (C) values between two pairs ($x_i$, $x_j$) of EEG channels were computed at delays $\tau$ ranging from -0.5 to 0.5 seconds in order to account for the propagation and processing time of brainwaves. We retained the maximal value of cross-correlation:

$$C_{a,b} = \max_{\tau} \left\{ \frac{C_{a,b}(\tau)}{\sqrt{C_i(0) \cdot C_b(0)}} \right\} \tag{2}$$

where

$$C_{a,b}(\tau) = \begin{cases} \frac{1}{N-\tau} \sum_{t=1}^{N-\tau} x_a(t+\tau) x_b(\tau) & \tau \geq 0 \\ C_{b,a}(-\tau) & \tau < 0 \end{cases} \tag{3}$$

and $N$ is the number of readings within the analysis window ($N = 400$ in our case). We also retained the non-delayed cross-correlation between the two signals.

### B. Euclidean Distance

Euclidean distance measures the distance in state-space between the trajectories of two EEG channels. First, each channel was time delay embedded into a trajectory with a time delay $\tau = 6$ readings and an embedding dimension $d = 10$, as seen in [3]:

$$\mathbf{x}(t) = \{x(t - (d-1)\tau), \ldots, x(t-\tau), x(t)\} \tag{4}$$

Then, we computed the distance of each time-delay embedded vector to its K nearest neighbors in state space:

$$\frac{1}{N} \sum_{t=1}^{n} \frac{\frac{1}{K} \sum_{k=1}^{K} \|\mathbf{x}_a(t) - \mathbf{x}_a(t_k^a)\|_2^2}{\frac{1}{K} \sum_{k=1}^{K} \|\mathbf{x}_a(t) - \mathbf{x}_a(t_k^b)\|_2^2} \tag{5}$$

where:

$\{t_1^a, t_2^a, \ldots, t_K^a\}$ are the time indices of the $K$ nearest neighbors of $\mathbf{x}_a(t)$, and
$\{t_1^b, t_2^b, \ldots, t_K^b\}$ are the time indices of the $K$ nearest neighbors of $\mathbf{x}_b(t)$.

### C. Pearson Correlation

The Pearson correlation coefficient measures the linear relationship between two time series $(x, y)$, giving a value between +1 and -1, where 1 is total positive correlation, 0 is no correlation, and -1 is total negative correlation. The formula for Pearson correlation coefficient when applied to a sample is:

$$r = \frac{1}{n-1}\sum_{i=1}^{n}\Big(\frac{x_i - \bar{x}}{s_x}\Big)\Big(\frac{y_i - \bar{y}}{s_y}\Big) \tag{6}$$

where

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i, \text{ and } s_x = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{7}$$

are the sample mean and sample standard deviation, respectively. This expression gives the correlation coefficient as the mean of the products of the standard scores.

### D. Spearman Correlation

The Spearman correlation coefficient is defined as the Pearson correlation coefficient between the ranked variables:

$$\rho = 1 - \frac{6\sum d_i^2}{n(n^2-1)} \tag{8}$$

where $d_i = x_i - y_i$ is the difference between ranks.

## V. Models

After selecting our features, we proceeded to run multiple learning algorithms using our dataset in order to select the model with the highest training accuracy and cross validation accuracy. In this section, we present the models we tried and how we found the optimal parameters for these models.

### A. Model Selection

We trained our data on various types of solvers in the LIBLINEAR package, as well as with varying kernel options in the LIBSVM package (see [1] and [4]).

*1) L2-Regularized Logistic Regression:* Solves the following unconstrained optimization problem:

$$\min_{\mathbf{w}} \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{l}\log(1 + e^{-y_i\mathbf{w}^T\mathbf{x}_i}) \tag{9}$$

where $C$ is the cost parameter which determines the degree of misclassification of the objective function.

*2) L2-regularized, L2-loss Linear Kernel SVM (primal):* Solves the following primal problem:

$$\min_{\mathbf{w}} \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{l}(\max(0, 1 - y_i\mathbf{w}^T\mathbf{x}_i))^2. \tag{10}$$

*3) L1-regularized, L2-loss Linear Kernel SVM:* Solves the following primal problem:

$$\min_{\mathbf{w}} \|\mathbf{w}\|_1 + C\sum_{i=1}^{l}(\max(0, 1 - y_i\mathbf{w}^T\mathbf{x}_i))^2 \tag{11}$$

*4) RBF Kernel Support Vector Classification:* Solves the following optimization problem:

$$\begin{aligned} &\min_{\mathbf{w},b,\xi} \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{l}\xi_i \\ &\text{subject to: } y_i(\mathbf{w}^T\phi(x_i) + b) \geq 1 - \xi_i \\ &\qquad \xi_i \geq 0 \\ &\text{using the kernel: } K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma\left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2\right) \\ &\qquad \gamma > 0 \end{aligned} \tag{12}$$

*5) Sigmoid Kernel Support Vector Classification:* This SVM solves the same optimization problem as in RBF Kernel Support Vector Classification, but with the kernel function:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\gamma\mathbf{x}_i^T\mathbf{x}_j + r) \tag{13}$$

*6) K-Nearest Neighbors Classification:* Neighbors-based classification labels each example, $x^{(k)}$, according to:

$$\hat{f}(x^{(q)}) = \arg\max_{v\in V}\sum_{k=1}^{K} d(v, f(x^{(k)})) \tag{14}$$

where each training example $v$ is sampled from the training set $V$ and:

$$d(x^{(i)}, x^{(j)}) = \sqrt{\sum_{r=1}^{n}(x_r^i - x_r^j)^2} \tag{15}$$

The results of our testing are shown in TABLE I. Because we have significantly more training instances than features, we tried non-linear kernels to increase the model complexity for higher accuracy. However, we found the computational time of the LIBSVM software on our dataset to be much too slow to run in real time, even after scaling our data to a [0, 1] range. After comparing all of our models, we found that L2-regularized Logistic Regression in the LIBLINEAR package was the optimal model in terms of accuracy and computational time.

### B. Parameter Selection

For L2-regularized logistic regression, the only parameter to optimize is the value of the cost parameter $C$ in the formulation for the L2-regularized logistic regression optimization problem (9). To find the optimal value of $C$, we first performed coarse grid search by running the algorithm using exponentially increasing values for $C$, from $2^{-2}$ to $2^{13}$. A portion of the resulting plot is shown in Fig. 2. The value in this range that resulted in the lowest cross validation error was $C = 16$. We then centered in on

TABLE I: Performance of different models

| Model | Training Accuracy | 5-Fold CV accuracy | Num Iterations |
|---|---|---|---|
| L2-reg LR (primal) | 86.2472% | 85.7194% | 16 |
| L2-reg, L2-loss, linar kernel SVM (dual) | – | – | max |
| L2-reg, L2-loss, linar kernel SVM (primal) | 85.9361% | 85.6694% | 16 |
| L2-reg, L1-loss, linar kernel SVM (dual) | – | – | max |
| L1-reg, L2-loss, linar kernel SVM | 86.2639% | 85.8444% | 52 |
| RBF kernel SVM | 100% | 50% | 1800 |
| Sigmoid kernel SVM | 50% | 50% | 180 |
| KNN | 100% | 54.2000% | – |

this value and performed fine grid search with a smaller range, and finally deduced that $C = 16$ was the optimal value.

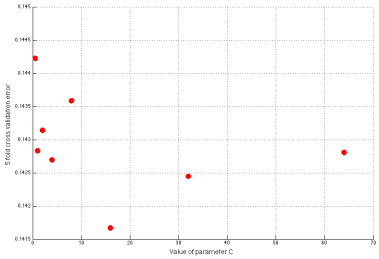


Fig. 2: Cross-validation error for different values of $C$

## VI. Results

### A. Single Feature Performance

TABLE II shows the accuracies from running L2-regularized logistic regression on each of the feature types individually. It includes the baseline features (univariate), linear features (MCC), and nonlinear features (Euclidean distance, similarity, Spearman correlation, Pearson correlation). The best cross-validation accuracy achieved with a single feature type was 78.4% (using maximum cross-correlation). However, using a combination of feature types increased this accuracy by about 7%, as discussed in the next subsection.

TABLE II: Individual Feature Accuracies

| Feature Type | Training Accuracy | 5-Fold CV accuracy |
|---|---|---|
| Univariate | 52.0222% | 51.3694% |
| Euclidean distance | 70.4472% | 70.1194% |
| Max cross-correlation | 78.6972% | 78.4000% |
| MCC with time delay | 54.9806% | 54.2750% |
| Similarity coefficient | 52.6000% | 52.2167% |
| Spearman correlation | 76.7472% | 76.4944% |
| Pearson correlation | 77.1833% | 76.9444% |

### B. Ablative Analysis

To choose an optimal feature set, we performed ablative analysis, starting with all features and removing them one at a time to see how each feature impacts performance. As shown in TABLE III, removing the Pearson and Spearman correlation features actually increased the cross-validation accuracy by a small amount. From there, removing the time-delayed features decreased accuracy by about 2.5%, removing the maximum cross-correlation features decreased accuracy by about 13%, and removing Euclidean distance features decreased accuracy by about 19%, leaving us with the baseline accuracy of about 51%. Using these results, we determined that the optimal feature subset consists of univariate, Euclidean distance, maximum cross-correlation, and time delay features, which gives us an cross-validation accuracy of 85.725%.

TABLE III: Ablative analysis on feature combinations

| Feature Combination | Training Accuracy | 5-Fold CV accuracy |
|---|---|---|
| Univariate + Euclidean + MCC + MCC w/ time delay + Pearson + Spearman | 85.9917% | 85.4944% |
| Univariate + Euclidean + MCC + MCC w/ time delay + Pearson | 86.0111% | 85.6583% |
| Univariate + Euclidean + MCC + MCC w/ time delay | 85.9500% | 85.7250% |
| Univariate + Euclidean + MCC | 83.7639% | 83.2917% |
| Univariate + Euclidean | 70.4472% | 70.1194% |
| Univariate | 52.0222% | 51.3694% |

### C. Precision and Recall

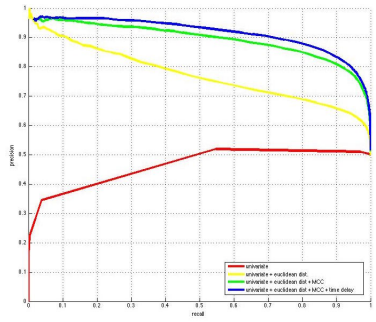
For seizure prediction, high recall is especially important so that seizure warnings are not missed. However, a relatively high precision is also important to avoid unnecessary stress caused by false positives. By adjusting the threshold probability at which to output a positive prediction (Fig. 3), we determined that we acheived a maximum F1 score of 86.68% at a threshold of 0.4502 (TABLE IV).

TABLE IV: Confusion matrix and relevant values at the threshold probability that gives the maximum F1 score.

| | | Prediction | | |
|---|---|---|---|---|
| | | Preictal | Interictal | Total |
| Actual | Preictal | 14,094 | 3,906 | 18,000 |
| | Interictal | 1,243 | 16,757 | 18,000 |
| | Total | 15,337 | 20,663 | 36,000 |

| | |
|---|---|
| Precision | 0.8110 |
| Recall | 0.9309 |
| Specificity | 0.7830 |
| F1 Score | 0.8668 |

(a) Precision-recall curves



(b) ROC curves

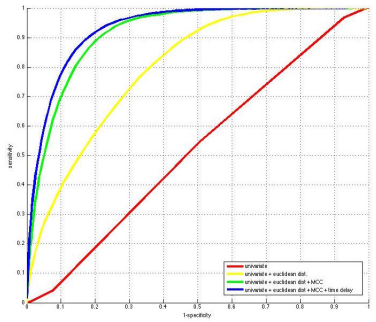


Fig. 3: Precision-recall and ROC curves for different feature combinations

### D. Error Diagnostics

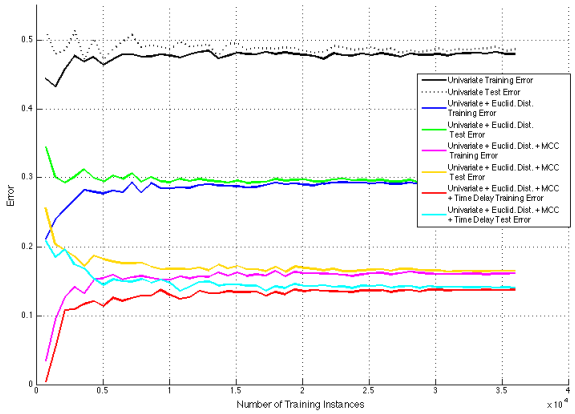



Fig. 4: Training and cross-validation error for different feature combinations

To diagnose the bias vs. variance of our learning algorithm, we plotted the training error and the test error for different training set sizes (Fig. 4), starting with just the 15 baseline univariate features (1 feature for each electrode channel). Because the training error and test error both plateaued to about the same values, we confirmed that we did not have high variance, and a larger training set would not increase accuracy. However, since the value at which the error converged was higher than our desired error, we determined that our classifier was suffering from high bias, so adding more features could help. We then concatenated the 105 Euclidean distance features onto the original features, and noticed that the training and test errors decreased. Next, we concatenated on the maximal cross-correlation features and noticed another decrease in error. Finally, after adding on the maximal cross-correlation with time delay features, we were able to reach a more desirable level of accuracy.

## VII. Conclusion

In this project we implemented a supervised binary classifier for EEG recordings. Throughout the course of our work, we progressed from a basic feature set with just one feature for each electrode channel, to a more complex feature space encapsulating non-linear and linear correlations between pairs of electrode channels. Through this process, we were able to increase our cross validation accuracy from just over 50% to 85.7%, corresponding to an F1 score of 86.68%.

## VIII. Future Work

We would like to try different models and feature combinations on multiple subjects, since we did not get a chance to experiment with more than one subject and the International Epilepsy Electrophysiology Portal has many more intracranial EEG datasets. We might try computing more complicated nonlinear features, such as dynamical entrainment or wavelet-based measures of synchrony. In the future, we could also compute relationships on more than two EEG channels, and eliminate pairs of channels that do not contribute much to the final prediction, in order to prevent the dimension of our feature vector from increasing linearly with the number of types of features added.

## References


[1] C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines," in *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[2] F. S. Bao, *et al.*, PyEEG: An Open Source Python Module for EEG/MEG Feature Extraction, in *Computational Intelligence and Neuroscience*, vol. 2011, Article ID 406391, 7 pages, 2011. doi:10.1155/2011/406391.

[3] P. Mirowski, *et al.*, "Classification of Patterns of EEG Synchronization for Seizure Prediction" Electroencephalography and Clinical Neurophysiology, 120 (11):19271940, Nov 2009.

[4] R.-E. Fan, *et al.*, LIBLINEAR: A Library for Large Linear Classification, in *Journal of Machine Learning Research* 9(2008), 1871-1874. Software available at http://www.csie.ntu.edu.tw/ cjlin/liblinear.

[5] A. Shoeb and J. Guttag, "Application of Machine Learning to Epileptic Seizure Detection," in *Proceedings of the International Conference on Machine Learning*, Haifa, Israel, 2010.

[6] T. P. Jung and S. Makeig. "Removing Artifacts from EEG." *ICA EEG Tutorial*. Web. 11 Dec. 2014.