Seizure Prediction Challenge (Group #3)

Ryan Marshall, Bassel El Mabsout, Aditya Chechani, William Chapman
{ryanmars, bmabsout, adityac, wchapman}@bu.edu

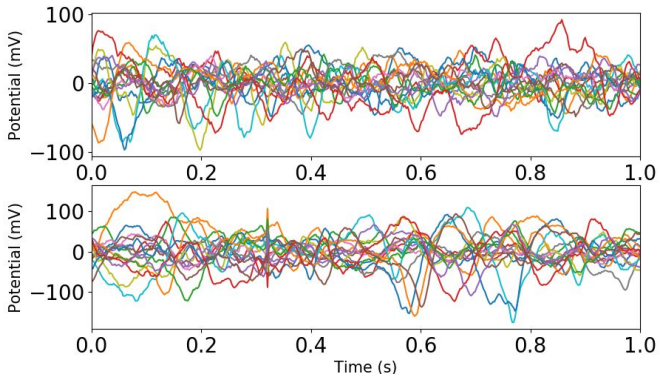

Figure 1: Showing 16 channel traces for one second of interictal (top) and preictal (bottom) data. Note the lack of any perceptible difference in the two.

1. Project Task

The goal of this project is to accurately predict whether a subject is going to experience a seizure in the near future. This is done by analyzing data from an intracranial EEG (iEEG) device implanted on the cortical surface of a subject. However, the current clinical standard for predicting seizure activity only has a sensitivity of 0.41 [1, 2]. The task is difficult because of two major issues. First, due to clinical constraints, the placement of electrodes varies between patients, and therefore the model must be trained separately for each patient, decreasing the size of training data for each model. Second, the spatiotemporal nature of the data requires a recurrent or convolutional network, as well as features indicating coherence between channels.

2. Related Work

Previous classification of iEEG into preictal and interictal epochs has tended to rely on hand designed features. These features include power in certain spectral frequencies, coherence between electrodes, and statistical summaries such as mean and kurtosis [4]. These features are calculated in various temporal bins, and fed into support vector machines [4] or random forests [4]. Notably, these earlier approaches specifically avoid the use of neural networks, due to the small size of previously available datasets and issues in overfitting.

3. Approach

Preprocessing
While the original dataset is sampled at 400Hz, there is little biological significance of signals above 100Hz, and the majority of signal comes in lower frequency bands, capped at the gamma frequency, with an upper bound of 40Hz. Meanwhile, the high sampling rate slows training and increases overfitting in machine learning approaches. For this reason, we downsample our data to 40Hz before further processing.

Baseline: Traditional Feature Classification
To provide a reasonable baseline comparison, we will replicate recent high performing algorithms, such as those that won the previous round of the Kaggle Competition [2]. All of these algorithms relied on hand-selected features, passed into a small variety of classifiers. Here, we implement the most common features, all calculated on a per-channel basis: mean, standard deviation, hurst-exponents, skew, kurtosis, power, channel-pair coherence, and power within specific spectral bands (2, 6, 10, 21, and 45 Hz). These statistics were calculated for three durations: 10 minutes, 1 minute, 30 seconds, with each time period being non-overlapping.

Autoregressive Coefficients
An alternative approach to the predefined features used previously, is to use an autoregressive model, in which the predictive features on each channel predict data at some latency. Specifically:

$$X_t = c + \sum_{i=1}^{p} \rho_i X_{t-i} + \varepsilon_t$$

Where X is a vector representing all signals at a given time point, X_{t-i} is a time lagged version of the same matrix, and ρ_i is a coefficient vector of coefficients. P is the model order, as determined by compared the Bayes Information Criterion (BIC) for orders ranging from zero to 15. As with the traditional features, these coefficients are calculated for non-overlapping blocks of either 10 minutes, 1 minute, or 30 seconds in length.

Feature-Based Classification:
600 second period features (both traditional and autoregressive) are used in four different classification algorithms. We separately evaluated traditional and autoregressive features, as well as a combination of the two. The first algorithm was a logistic regression classifier. The second was a support vector machine (SVM) with linear kernel. The third was a random-forest classifier. Finally, a shallow neural

network (dense layers) was used. The first three of these classifiers are chosen to align with previous approaches, while the neural network was chosen to allow non-linear interactions of provided features. The shape and other hyper parameters of the neural network were chosen by a gridsearch, as summarized in table 1. In all cases, hyperparameters were chosen based on optimal mean *accuracy* across a stratified K-Fold (10 folds) approach across all subjects.

Parameter	Evaluation Points
L1	1e-5, 1e-3, **0**
L2	1e-5, 1e-3, **0**
Batch_size	4, **32**, 64
Dropout	**0**:0.05:0.3
N_units	[32, 1], [64, 1], [32,64, 1], [64,32,1], **[32,64,32,1]**
Block Length (seconds)	1, 30, 60, **600**

Table 1: Parameters used in grid search for classification using AR-coefficients and a shallow network. N_units indicates the number of units in the dense layers. In all cases, the network was trained for 1000 epochs. Bold indicates optimal parameters.

CNN / LSTM
Long short-term memory (LSTM) units are a specialized recurrent neural network which are useful for predicting sequential data. These units have a seperate gate for input and forgetting which allows them to have variable sized inputs. Here, we will connect the output values to a single-unit dense layer and train the network by the binary entropy loss function. Convolutional neural networks can also be used classify our time series data as it is of fixed length and actually performs equally and sometimes better.

Unified Model:
Based on the above, we have models operating at two different timescales. The autoregressive features extract information on the order of .125 seconds while CNN capture information on the order of multiple seconds. Thus, we attempted to use the autoregressive coefficients from small temporal bins (5 second bins) as features in the same convolutional network described above. Such a network may improve on the CNN by incorporating long and short term information.

4. Dataset and Metric

Our project is based on a 2016 Kaggle competition [1]. The data are recorded from 16 electrodes sampled at 400 Hz, from three different patients. Each file consists of 10 minutes worth of recording, and is classified as either interictal or preictal. The number of training and testing points varies by subject (Table 2). All performances reported are based on 10-Fold cross validation metrics.

Metric: Consistent with the original competition, we will evaluate performance by the Area Under the Curve (AUC) on the cross-validation datasets, and aim for a measure of at least 0.745, placing us within the top 10%.

Subject	Training+CV (pos / neg / %)	Test
1	279 / 619 / 31%	144
2	240 / 2123 / 10%	697
3	273 / 2097 / 11%	483

Table 2: Dataset Summary

5. Results

Autoregressive Model Order:
An autoregressive model was fit for varying history-orders, ranging 0 to 15. Optimal model order was found to be 5 samples, corresponding to ⅛ of a second (figure 2). Notably, this aligns with the upperband of the human alpha rhythm.

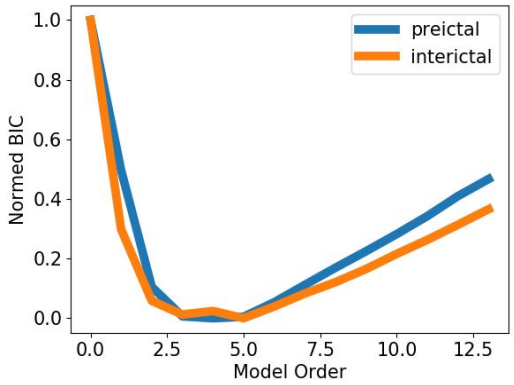

Figure 2: Showing BIC as a function of autoregressive model order Note that model order was not different for preictal and interictal trials.

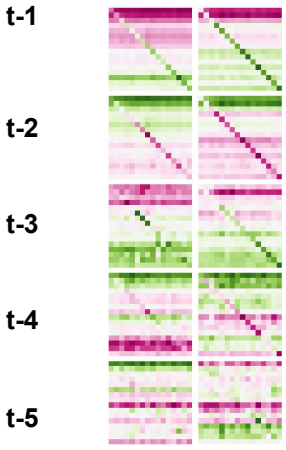

Figure 3: Showing AR coefficients for preictal (left) and interictal (right) segments. Each pixel indicates the AR coefficient from a given electrode (row) to another (column) at a given latency. Pink values indicate negative coefficients, while green indicate positive.

Traditional Features
Four separate classifiers were trained on the traditional features outlined in the introduction. AUC was calculated for each patient separately, and the mean AUC for that approach is the mean across all subjects. We found that all four provided AUC in the 0.75-0.80 range, consistent with previous reports (Figure 4).

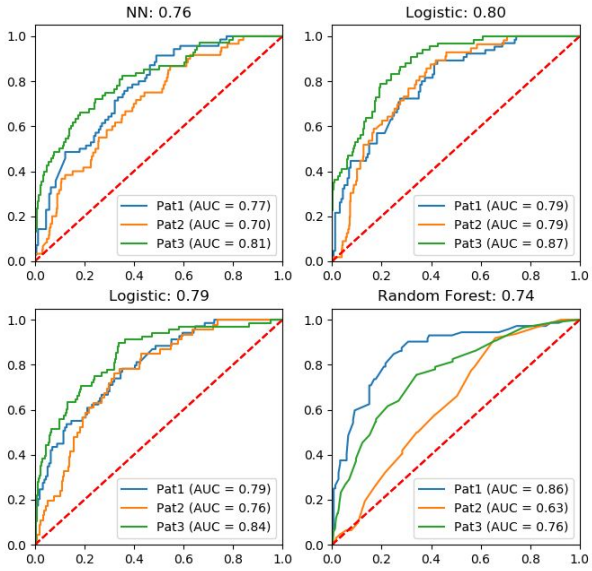

Figure 4: Showing the ROC curve for each approach using the traditional channel-based features.

Autoregressive Classification:
Similar to the traditional feature approach, separate classifiers were trained on the autoregressive features. We found an increased AUC regardless of the classifier used (Figure 5), and found that our optimal result was with the neural network classifier, providing performance higher than the previous best.

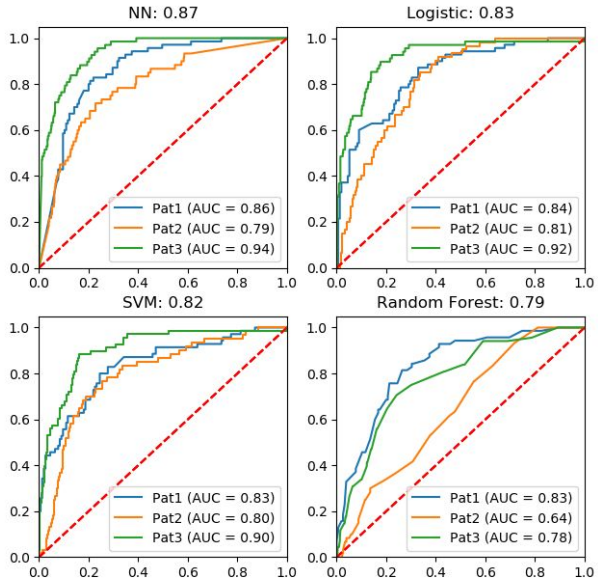

Figure 5: Showing ROC curve using autoregressive coefficients as features. This provided an increased AUC on all classification algorithms.

Combining these two sets of features (classic and autoregressive) was not found to increase performance at all, and decreased performance with all but the neural network approach.

LSTM
An initial LSTM model was constructed with a first many-to-many layer of 80 units, a second many-to-one 40 unit layer, connected to a dense layer with a logistic activation. The model seemed to overfit very easily getting an AUC of ROC that is 0.99 for training and 0.7 for testing even after grid searching dropout values and different sizes for the network and regularization amount. Also due to the sequential nature of LSTMs, training was generally much slower and GPUs made little to no difference.

Regularization
Firstly, due to the large number of features per data point available, the networks were overfitting by memorizing the data easily. Secondly, the neural networks usually used for time-series predictions are fairly large so that they can learn complex transformations of the input. Which allowed for easy overfitting in our case. To handle this we decreased the feature count. Initially convolutional autoencoders were employed for this task but the method was not more fruitful than simply downsampling the data. Downsampling by 100x preserved enough information to get a high accuracy while allowing for very fast iteration times. The data was also standard scaled. Drop-off and tuning the regularization parameters also helped but the biggest gain in terms of difference between the training and test accuracy was using a smaller network.

CNN

The highest performing CNN has 3 convolutional layers with 2x max pooling and drop-off in between. With the layers filters of 10,20,30, and kernel sizes of 20,10,5. Then, global average pooling and fully connected network to a single sigmoidal output is added. CNNs were much faster to train than LSTMs due to the fact that they are easily parallelizable. This allowed for a large number of iterations over the architecture directly leading to the accuracy that was achieved.

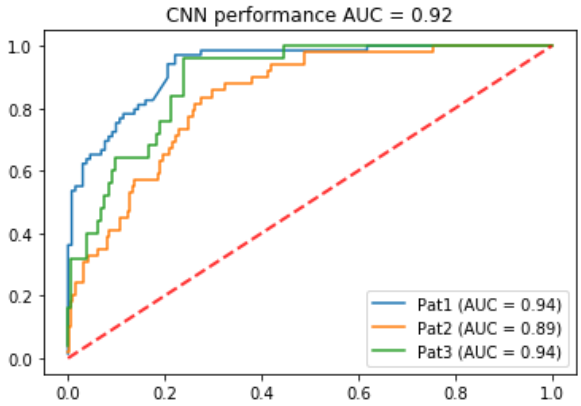

Figure 7: Showing ROC for the CNN model.

Unified Models:
Using the time-blocked AR-coefficients, we attempted a unified model which used autoregressive features calculated in 1-second temporal bins as inputs to the CNN described above. However, this model showed poor generalization on the cross validation data (Figure 8), despite strong performance on training data (0.98). This suggests that additional regularization may be necessary to incorporate these additional features into the temporal convolution network.

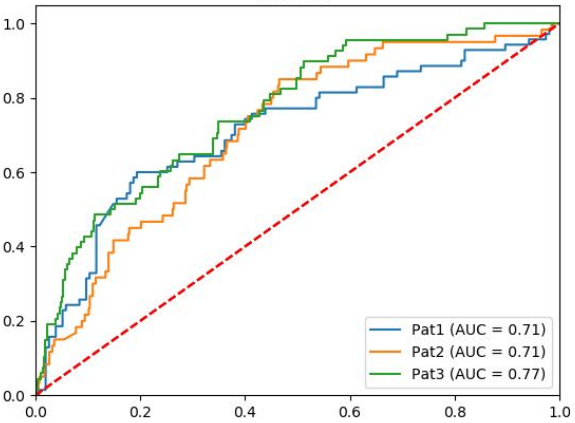

Figure 8: Showing the poor generalization of the unified AR+CNN network.

#	change	Team	Score	Entries
0	-	CNN **(ours)**	0.92	-
0	-	AR NN **(ours)**	0.87	-
1	+1	Not-so-random-anymore	0.807	260
(2)	-	Coef LogReg **(ours)**	0.80	0
2	+35	Arete Associates	0.798	56
3	+12	ZhengYi	0.796	74
...	-			
42 (10%)	+19	CDS_Grp5	0.745	22

Table 3: Kaggle Leaderboard. Note that this is the public leaderboard based on cross validation metrics, similar to our methods reported above. Performance on true test data could not be evaluate as the Kaggle competition was closed to new entries.

7. Discussion

We attempted two broad classes of machine learning approaches in an attempt to classify iEEG recordings as interictal or preictal in a clinically relevant dataset. Feature-based approaches were able to replicate previous approaches, and autoregressive coefficients combined with a shallow neural network achieved better performance than previous bests. A relatively simple 1D convolutional neural network was able to achieve even better performance on held-out data, with a performance of 0.92. While these approaches indicate a significant improvement over previous attempts, further improvement may be possible by integrating a convolutional neural network with preprocessed signals which contain information about smaller temporal sequences.

8. Timeline and Roles.

Task	Deadline	Lead
LSTM & CNN	11/22/18	Bassel
Classic Features	11/22/18	Ryan & William
AR Features + classifiers	11/22/18	William
Prepare proposal	11/01/18	William
Prepare Update 1	11/15/18	William
Prepare Update 2	11/29/18	Bassel & William
Prepare Report	12/11/18	Bassel & William
Prepare Poster	12/09/18	Bassel & William

-1. References

1) https://www.kaggle.com/c/melbourne-university-seizure-prediction
2) Kuhlmann, L. Crowd-Sourcing Reproducible Seizure Prediction with Long-Term Human Intracranial EEG, Brain, awy210, https://doi.org/10.1093/brain/awy210
3) Wang, Y., Yao, H., & Zhao, S. (2016). Auto-encoder based dimensionality reduction. Neurocomputing, 184, 232–242. https://doi.org/10.1016/J.NEUCOM.2015.08.104
4) M. Mursalin, Y. Zhang, Y. Chen, and N. V Chawla, "Automated epileptic seizure detection using improved correlation-based feature selection with random forest classifier," *Neurocomputing*, vol. 241, pp. 204–214, Jun. 2017.
5) Scikit-learn: Machine Learning in Python, Pedregosa *et al.*, JMLR 12, pp. 2825-2830, 2011.