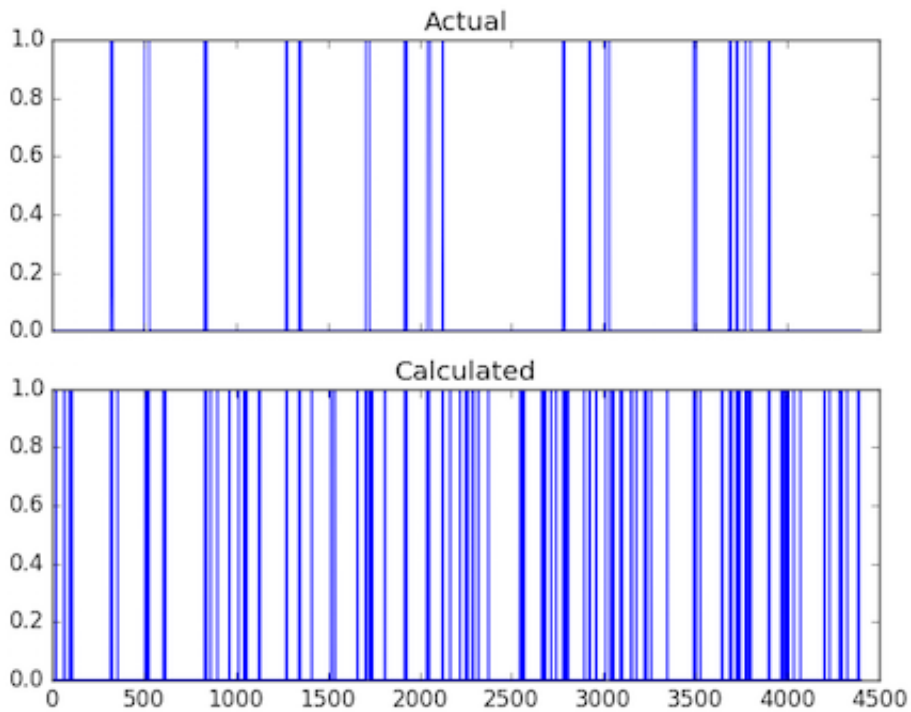
6/29/2016

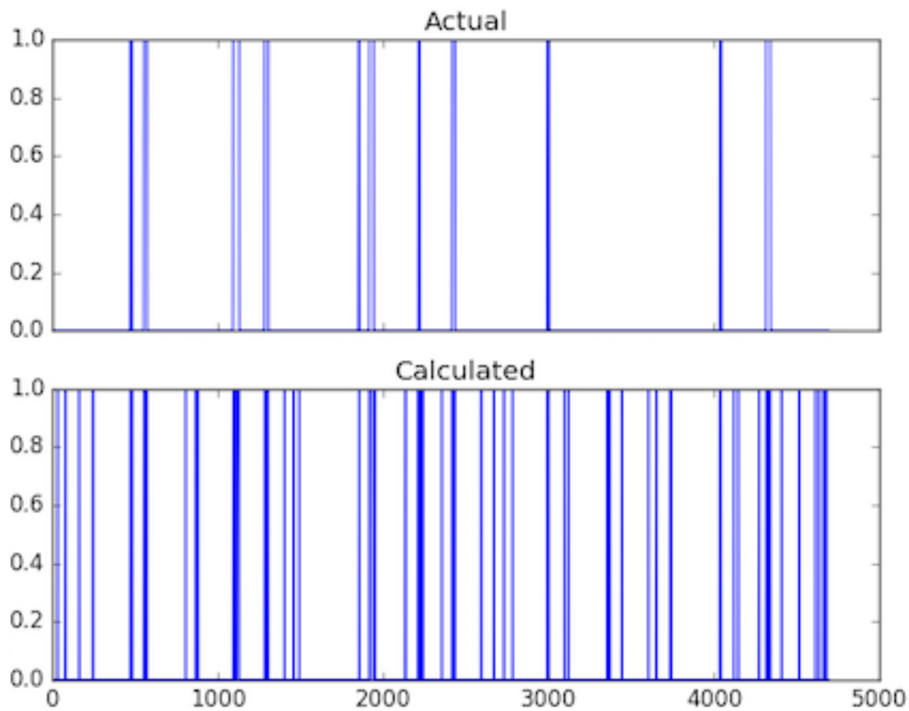
Today's Goal: Keep experimenting with different features and clustering methods. Today, I will try to calculate more features and do true plots.

Below are some generated hypnograms for the explicit decision tree threshold classification method:

Subject 1032:



Subject 1039:



There are many false positive (which was intended). To reduce this number as well as keep the true positives the same (or maybe even greater), more features and classification methods will be examined for this week.

From *Efficient unsupervised algorithms for the detection of seizures in continuous EEG recordings from rats after brain injury:*

The characteristics that made the seizure identifiable at this resolution included: (1) the increased height over baseline of these lines, (2) the increased correlation of the minima and maxima from one pixel to the next.

The mechanics of this method are as follows. A set of 3000 points (12 s) was considered. The maximum and minimum values for each 30-point group were computed. These were termed max(S_i) and min(S_i) where S_i represents a group of 30 points. One could simply sum the difference between the max and min over the 100 values of the set, but this would not take into account the correlation of the different time periods, and therefore would still be subject to noise. Instead, only the portion of the signal that was within the boundaries of the next two groups was summed, yielding the expressions:

$HV_i=\min[\max(S_i),\max(\max(S_{i+1}),\max(S_{i+2}))]$

and

$LV_i=\max[\min(S_i),\min(\min(S_{i+1}),\min(S_{i+2}))]$

where HV_i was the ith high value and LV_i was the ith low value. The metric used was the sum of the difference, i.e.

$$\text{metric}_3 = \sum_{i=1}^{100}(\text{HV}_i - \text{LV}_i).$$

This may be used for detecting REM, because REM sleep will show regular theta amplitudes while NREM/wake will show variable theta amplitudes.

From S*ignal Regularity-Based Automated Seizure Detection System For Scalp EEG Monitoring:*

This paper uses a metric PMRS that was motivated by the Approximate Entropy statistic, which quantifies the unpredictability of fluctuations in a time series. The calculation of Approximate Entropy is based on a conditional probability of the next corresponding points being value-matched given that the previous *m* corresponding points are all value-matched, for a fixed integer *m*. A value match is defined as the maximum difference between the corresponding points of two subsequences being less than a critical value *r*. For different selections of parameters *m* and *r*, Approximate Entropy could yield very inconsistent results, even when the choices are within a reasonable range.

I will try to implement these features and evaluate their effectiveness. To evaluate effectiveness, I'll run a One-Way ANOVA to see how different the means within the sleep states are for each feature.

For reference, these are the ANOVA results for the original feature (0=delta power, 1=delta/theta ratio, 2=emg power)

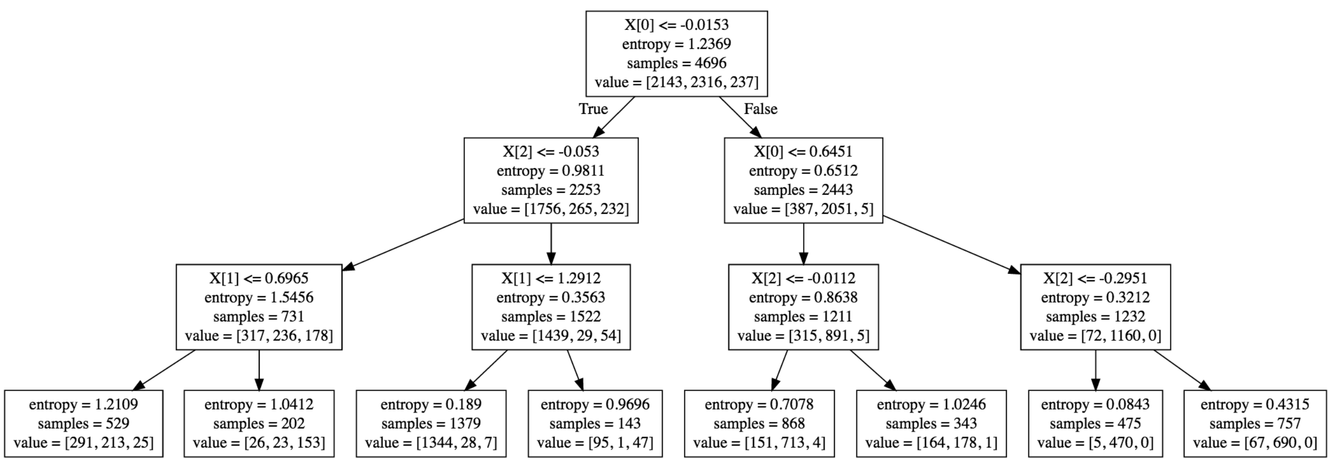
```
Feature 0 has F=1268.6069069592656 and p=0.0
Feature 1 has F=622.4617349085395 and p=1.679368768149662e-240
Feature 2 has F=232.53867568162838 and p=5.081407497736326e-97
```

Seems like these three features are extremely strong.

Implemented Autocorrelation feature and tested effectiveness with ANOVA. It returned the following results:

```
F=165.2396472052387 and p=4.481183698728 22e-70
```

I ran the decision tree with all features and autocorrelation. The outputted graph (entropy with depth=3) was this:



And the statistics were like this (subject 1032):

```
Score of Decision Tree: 0.8377342419080068
Wake. Calculated 2051 with actual 2143
NREM. Calculated 2443 with actual 2316
REM. Calculated 202 with actual 237
```

This is marginally better than the entropy depth 3 decision tree with just three features, but strangely, the auto correlation feature is not even used in this decision tree, so it may be safe to say that it is not adding any useful information.