# Predicting Epilepsy using Intracranial EEG Data


**Minor Project Report**

Siddharth Lalwani

B.Tech CSE 7th Sem

02816403213

**Mentor-** Prof C S Rai

# CERTIFICATE

This is to certify that the project entitled **"Predicting Epilepsy using Intracranial EEG Data"** was successfully completed by **Mr. Siddharth Lalwani**, under my supervision for the partial fulfilment of requirement for the award of B. Tech Computer Science from GGS Indraprastha University, Delhi during 7$^{th}$ semester.

Dated: 21$^{st}$ November, 2016

Prof C.S Rai

# DECLARATION

I Siddharth Lalwani, a student of B.Tech (CSE) 7th Semester hereby declare that this project report submitted is in the partial fulfillment of requirements for the award of the degree of Bachelors of Technology in Computer Science and Engineering.

My Minor Project Program work term was under mentorship of **Prof C S Rai.** This Project develops a Seizure forecasting System to help patients with epilepsy lead more normal lives.

Siddharth Lalwani

# ACKNOWLEDGEMENT

I am sincerely thankful to all who have provided me with invaluable assistance during this project and have helped me to develop this project report. I take this opportunity to express my gratitude to Prof C.S Rai for the guidance and providing me with a high conceptual understanding of the project, which helped make my project extremely interesting to me and gave me the impetus to work hard. With his vast experience and knowledge, he has been able to guide me ably and successfully towards the successful completion of the project.

I also express my sincere thanks to University School of Information and Communication Technology (USICT), Guru Gobind Singh Indraprastha University for providing me this opportunity and allowing me to work on this project.

Last but not the least, I am grateful to my parents whose blessings and encouragement has provided me the strength to complete the project of this complexity.

Siddharth Lalwani

# CONTENTS

# *Introduction*

The apparently random nature of seizures is a significant factor affecting the quality of life for patients with epilepsy. Despite taking daily medications many patients with epilepsy continue to have seizures. Accurate seizure forecasting could transform epilepsy care, allowing patients to modify activities to avoid risk and take antiepileptic drugs only when needed to stop seizures before they develop. However, to achieve clinically relevant seizure forecasting, better methods are needed for identifying periods when seizures are likely to occur. Significant evidence has emerged supporting the idea that seizures arise from an identifiable preictal brain state. Clinical studies describe patients self-reporting seizure-prone states prior to seizure at a rate greater than chance, and changes in cerebral blood flow, oxygenation, and cortical excitability have been reported prior to seizures

While many early seizure forecasting studies using EEG features suffered from inadequate statistical analysis, particularly with regards to adequate sampling of the interictal period, recent studies have demonstrated in a rigorous statistical framework that human and canine seizure forecasting is possible. A major challenge for seizure forecasting research has been the lack of long duration recordings with adequate interictal data and number of seizures for rigorous statistical testing. The majority of early studies were limited to relatively short human intracranial EEG (iEEG) recordings obtained as part of epilepsy surgery evaluations. These clinical iEEG studies from the epilepsy monitoring units rarely extend beyond 10 days and are enriched with seizures because the antiepileptic drugs are tapered to expedite the evaluation. These clinical records rarely yield an adequate number of seizures separated by clear interictal periods for rigorous statistical testing, and thus are limited in their usefulness to develop predictors of patients' habitual seizures. Longer-duration iEEG recordings have been analysed from epileptic animal models where an artificial epileptic focus is created, but the usefulness of these models to develop algorithms for forecasting naturally occurring focal epilepsy remains unclear.

# *Impact of the project*

Seizure forecasting systems have the potential to help patients with epilepsy lead more normal lives. In order for EEG-based seizure forecasting systems to work effectively, computational algorithms must reliably identify periods of increased probability of seizure occurrence. If these seizure-permissive brain states can be identified, devices designed to warn patients of impeding seizures would be possible. Patients could avoid potentially dangerous activities like driving or swimming, and medications could be administered only when needed to prevent impending seizures, reducing overall side effects.

There is emerging evidence that the temporal dynamics of brain activity can be classified into 4 states: Interictal (between seizures, or baseline), Preictal (prior to seizure), Ictal (seizure), and Post-ictal (after seizures). Seizure forecasting requires the ability to reliably identify a preictal state that can be differentiated from the interictal, ictal, and postictal state. The primary challenge in seizure forecasting is differentiating between the preictal and interictal states. The goal of the project is to demonstrate the existence and accurate classification of the preictal brain state in dogs and humans with naturally occurring epilepsy.

For many years, experts in neurology, computer science, and engineering have worked toward developing algorithms to predict a seizure before it occurs. If an algorithm could detect subtle changes in the electrical activity of a person’s brain (measured by electroencephalography (EEG)) before a seizure occurs, people with epilepsy could take medications only when needed, and possibly reclaim some of those daily activities many of us take for granted. But algorithm development and testing requires substantial quantities of suitable data, and progress has been slow. Many early research reports developed and tested algorithms on relatively short intracranial EEG data segments from patients with epilepsy undergoing intracranial EEG before surgery. There are a number of problems with this. First, patients undergoing pre-surgical monitoring for epilepsy typically have their medications reduced to encourage seizures to occur, which causes a progressive decrease in the blood levels of medications which have been shown to affect the normal baseline pattern in a patient’s EEG. Second, hospital stays for pre-surgical monitoring by necessity rarely last more than two weeks, providing a very limited amount of data for any single patient.

# *Subjects and data*

Intracranial EEG data were recorded chronically from eight canines with naturally occurring epilepsy using the NeuroVista seizure advisory system implanted device described previously. The dogs were housed at the veterinary hospitals at the University of Minnesota and University of Pennsylvania. Sixteen subdural electrodes were implanted intracranially in each canine in a bilaterally symmetrical arrangement, with paired four-contact strips oriented from anterior to posterior on each hemisphere.

The electrode wires were tunnelled caudally through openings in the cranium, anchored, looped and passed under the skin to the implanted telemetry unit medial to the dog's shoulder. Wires were connected to a recording device, which was implanted under the latissimus dorsi muscle and iEEG data were wirelessly telemetered to a receiver and storage unit in a vest worn by the dog. Recorded data were stored on removable flash media, which were periodically removed and copied via the internet to a cloud storage platform for subsequent analysis. The implanted recording device was powered by a rechargeable battery unit, which was charged daily by monitoring personnel.

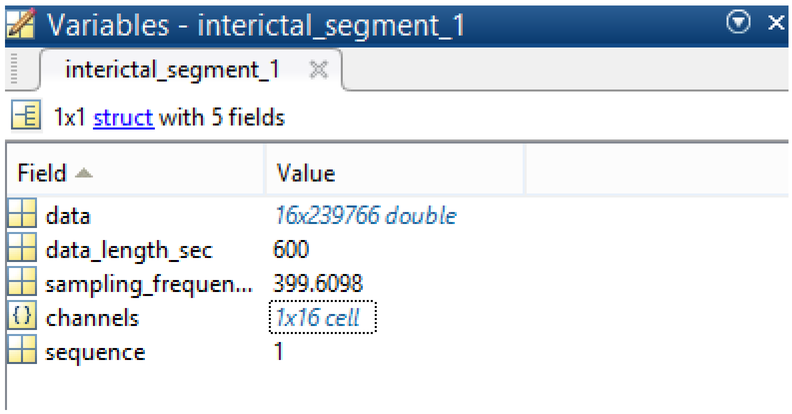
Data characteristics for the Kaggle.com seizure forecasting contest and held-out data experiment

| **Subject** | **Sampling rate (Hz)** | **Recorded data (h)** | **Seizures** | **Lead seizures** | **Training clips (% interictal)** | **Testing clips (% interictal)** | **Held-out clips (% interictal)** |
|---|---|---|---|---|---|---|---|
| Dog 1 | 400 | 1920 | 22 | 8 | 504 (95.2) | 502 (95.2) | 2000 (99.7) |

| Subject | Sampling rate (Hz) | Recorded data (h) | Seizures | Lead seizures | Training clips (% interictal) | Testing clips (% interictal) | Held-out clips (% interictal) |
|---|---|---|---|---|---|---|---|
| Dog 2 | 400 | 8208 | 47 | 40 | 542 (92.3) | 1000 (91.0) | 1000 (100) |
| Dog 3 | 400 | 5112 | 104 | 18 | 1512 (95.2) | 907 (95.4) | 1000 (100) |
| Dog 4 | 400 | 7152 | 29 | 27 | 901 (89.2) | 990 (94.2) | 1000 (95.8) |
| Dog 5 | 400 | 5616 | 19 | 8 | 480 (93.8) | 191 (93.7) | 0 |
| Patient 1 | 5000 | 71.3 | 5 | 4 | 68 (73.5) | 195 (93.9) | 0 |
| Patient 2 | 5000 | 158.5 | 41 | 6 | 60 (70.0) | 150 (90.7) | 0 |

# *Getting started*

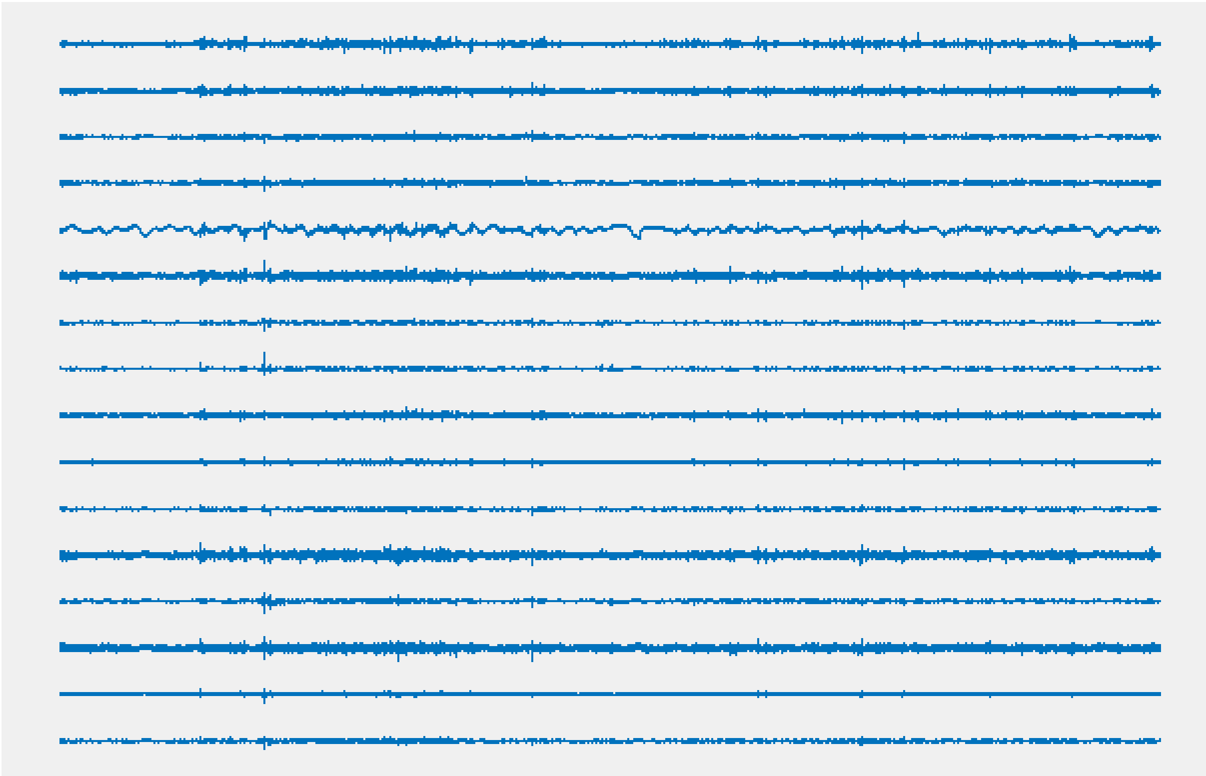
Firstly, I analyzed the data inside the .mat file for a few interictal and preictal segment. The aim of the project is to distinguish between preictal and interictal segments.




The .mat segment files consist of 1 X 1 struct with variables data, data_length_sec, sampling_frequency,channels and sequence . The channels refers to the various electrodes placed on different parts on the head.

**Matlab code to plot an interictal segment ->**

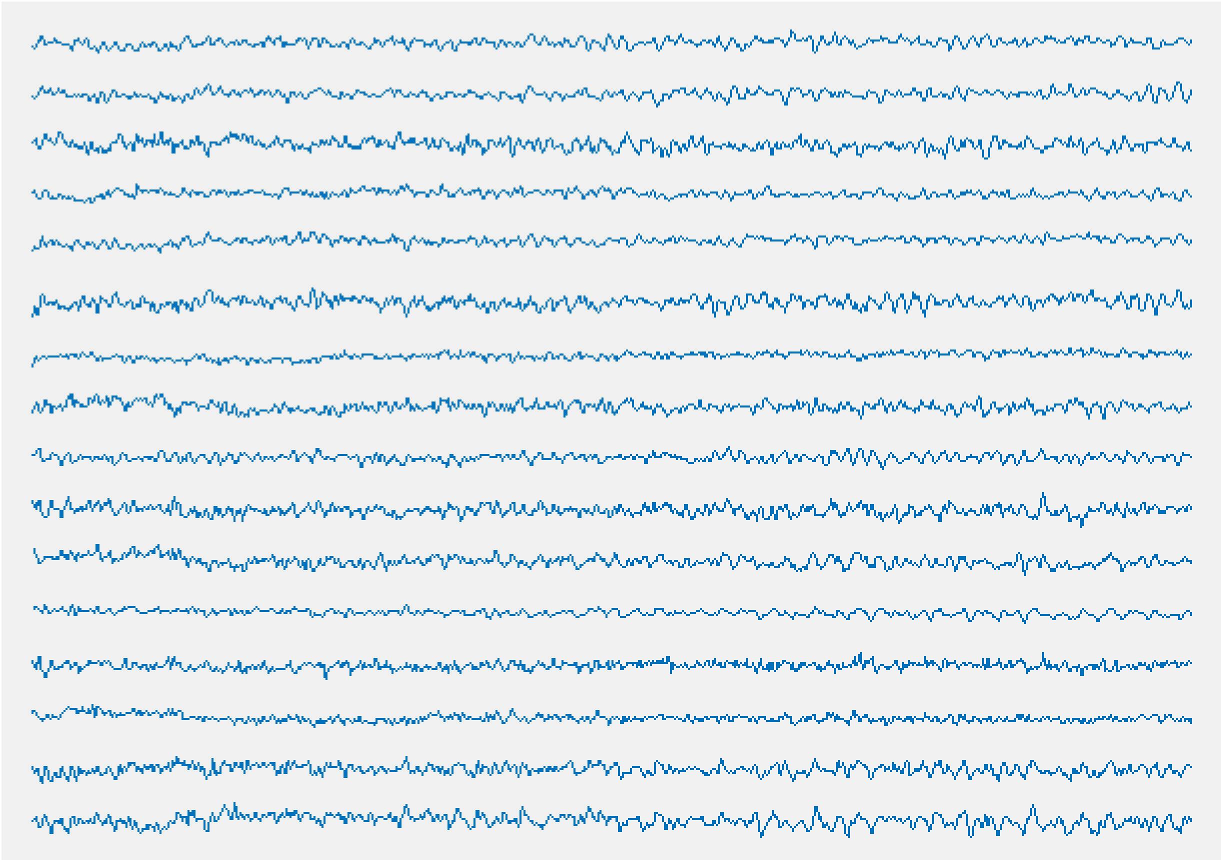
```
for j=1:16

          subplot(16,1,j);

          plot(1:p,interictal_segment_1.data(j:j,:))

          axis off

end
```



*Interictal Segment*

**Matlab code to plot a preictal segment ->**

```
for j=1:16
    subplot(16,1,j);
    plot(k,preictal_segment_3.data(j:j,1:1000))
    axis off
end
```



*Preictal Segment*

# *Using Scipy Library functions->*

SciPy is a collection of mathematical algorithms and convenience functions built on the Numpy extension of Python. It adds significant power to the interactive Python session by providing the user with high-level commands and classes for manipulating and visualizing data. With SciPy an interactive Python session becomes a data-processing and system-prototyping environment rivaling systems such as MATLAB, IDL, Octave, R-Lab, and SciLab.

The additional benefit of basing SciPy on Python is that this also makes a powerful programming language available for use in developing sophisticated programs and specialized applications. Scientific applications using SciPy benefit from the development of additional modules in numerous niches of the software landscape by developers across the world. Everything from parallel programming to web and data-base subroutines and classes have been made available to the Python programmer.

To use the interictal or preictal data to train any model using scipy, firstly I have to convert the data 2d matrix into 1d sequence.This is done using

eegDict[seg][0][0][0]**.ravel()**

The first model I have used for classification of preictal segments is **Decision Tree Classifier**.

**Decision Trees (DTs)** are a non-parametric supervised learning method used for classification and regression. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features.

## Simple Implementation for a particular subject (Dog1) ->

Editor - Canopy

File Edit View Search Run Tools Window Help

*tree_1.py

```python
import scipy.io
from sklearn import tree
import cPickle
import numpy as np

#eegDict=[]
Data=[]
A=[]
for i in range(1,400):
    addr='D:\\Epilepsy\\Dog_1\\Dog_1_interictal_segment_0'+str(str(i).zfill(3))+'.mat'
    eegDict= scipy.io.loadmat(addr)
    seg='interictal_segment_'+str(i)
    Data.append(eegDict[seg][0][0][0].ravel())
    A.append(0)
    print(addr)
for i in range(1,25):
    addr='D:\\Epilepsy\\Dog_1\\Dog_1_preictal_segment_0'+str(str(i).zfill(3))+'.mat'
    eegDict= scipy.io.loadmat(addr)
    seg='preictal_segment_'+str(i)
    Data.append(eegDict[seg][0][0][0].ravel())
    A.append(1)
    print(addr)

cPickle.dump(Data, open('D:\\Epilepsy\\save.p', 'wb'))


res = tree.DecisionTreeClassifier()
res= res.fit(Data, A) #Training the model

S=[Data[0],Data[1],Data[401],Data[402]]
print(res.predict(S))

print("Yes")
```

Cursor pos 30 : 40 Python

*Implementation using Python in Enthought Canopy*

Another approach I tried for classification was using **Support Vector Machines**.

A Support Vector Machine (SVM) is a discriminative classifier formally defined by a separating hyperplane. In other words, given labeled training data (*supervised learning*), the algorithm outputs an optimal hyperplane which categorizes new examples.

**Simple SVC implementation for a particular subject(Dog1)->**

```
#Made by Siddharth Lalwani,Minor Project
import scipy.io
from sklearn import svm
import cPickle

Data=[]
A=[]
for i in range(200,400):
    addr='D:\\Epilepsy\\Dog_1\\Dog_1_interictal_segment_0'+str(str(i).zfill(3))+'.mat'
    eegDict= scipy.io.loadmat(addr)
    seg='interictal_segment_'+str(i)
    Data.append(eegDict[seg][0][0][0].ravel())
    A.append(0)
    print(addr)
for i in range(1,25):
    addr='D:\\Epilepsy\\Dog_1\\Dog_1_preictal_segment_0'+str(str(i).zfill(3))+'.mat'
    eegDict= scipy.io.loadmat(addr)
    seg='preictal_segment_'+str(i)
    Data.append(eegDict[seg][0][0][0].ravel())
    A.append(1)
    print(addr)

#cPickle.dump(Data, open('D:\\Epilepsy\\save.p', 'wb'))
res = svm.SVC()
res= res.fit(Data, A)

Test=[]
for i in range(1,25):
    addr='D:\\Epilepsy\\Dog_1\\Dog_1_preictal_segment_0'+str(str(i).zfill(3))+'.mat'
    eegDict= scipy.io.loadmat(addr)
    seg='preictal_segment_'+str(i)
    Test.append(eegDict[seg][0][0][0].ravel())
    A.append(1)
    print(addr)
#cPickle.dump(res, open('D:\\Epilepsy\\model.p', 'wb'))

print(res.predict(Test))
```

# Sample Results

*Using Decision Tree Classification for 20 Interictal Segments*

```
Test=[]
for i in range(400,420):
    addr='D:\\Epilepsy\\Dog_1\\Dog_1_interictal_segment_0'+str(str(i).zfill(3))+'.mat'
    eegDict= scipy.io.loadmat(addr)
    seg='interictal_segment_'+str(i)
    Test.append(eegDict[seg][0][0][0].ravel())

#cPickle.dump(res, open('D:\\Epilepsy\\model.p', 'wb'))


print(res.predict(Test))

#with open('D:\\Epilepsy\\Data.txt', 'w') as f:
    #for s in Data:
        #f.write((str(s) + u'\n').encode('unicode-escape'))
```

Python

```
D:\Epilepsy\Dog_1\Dog_1_preictal_segment_0019.mat
D:\Epilepsy\Dog_1\Dog_1_preictal_segment_0020.mat
D:\Epilepsy\Dog_1\Dog_1_preictal_segment_0021.mat
D:\Epilepsy\Dog_1\Dog_1_preictal_segment_0022.mat
D:\Epilepsy\Dog_1\Dog_1_preictal_segment_0023.mat
D:\Epilepsy\Dog_1\Dog_1_preictal_segment_0024.mat
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
```

For the following interictal segments, Model correctly predicts all of them to be 0 ,that is , not preictal.

*Using Decision Tree Classification for 24 Preictal Segments*

```

Test=[]
for i in range(1,25):
    addr='D:\\Epilepsy\\Dog_1\\Dog_1_preictal_segment_0'+str(str(i).zfill(3))+'.mat'
    eegDict= scipy.io.loadmat(addr)
    seg='preictal_segment_'+str(i)
    Test.append(eegDict[seg][0][0][0].ravel())
    A.append(1)
    print(addr)
#cPickle.dump(res, open('D:\\Epilepsy\\model.p', 'wb'))


print(res.predict(Test))

#with open('D:\\Epilepsy\\Data.txt', 'w') as f:
    #for s in Data:
        #f.write((str(s) + u'\n').encode('unicode-escape'))
```

Python

```
D:\Epilepsy\Dog_1\Dog_1_preictal_segment_0019.mat
D:\Epilepsy\Dog_1\Dog_1_preictal_segment_0020.mat
D:\Epilepsy\Dog_1\Dog_1_preictal_segment_0021.mat
D:\Epilepsy\Dog_1\Dog_1_preictal_segment_0022.mat
D:\Epilepsy\Dog_1\Dog_1_preictal_segment_0023.mat
D:\Epilepsy\Dog_1\Dog_1_preictal_segment_0024.mat
[1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1]
```

Due to limitation of preictal data, we are using 20 preictal segments redundantly for both training and testing. The model correctly classifies the segments to be preictal.

# *Using multiprocessing*

Because the dataset is very large, we will use multiprocessing while extracting features from the data. Multiprocessing is a package in python that supports spawning processes using an API similar to the threading module. The multiprocessing package offers both local and remote concurrency, effectively side-stepping the Global Interpreter Lock by using subprocesses instead of threads. Due to this, the multiprocessing module allows the programmer to fully leverage multiple processors on a given machine.

I have used 3 major functions of multiprocessing library->

**n_cpus = mp.cpu_count()**

It finds the number of CPUs in the system.

**pool = mp.Pool(n_cpus)**

A Pool object controls a pool of worker processes. Jobs can be submitted to the Pool, which then sends the jobs to the individual workers.

**result = pool.map(worker, args)**

The Pool.map() achieves the same functionality as Matlab's **parfor** construct. This method essentially applies a function to each element in an iterable and returns the results. For example, if I wanted to square each number in a list of integers between 0 and 9 and perform the square operation on multiple processors, I would write a function for squaring an argument, and supply this function and the list of integers to Pool.map().

# *Feature Extraction*

Feature extraction involves reducing the amount of resources required to describe a large set of data. When performing analysis of complex data one of the major problems stems from the number of variables involved. Analysis with a large number of variables generally requires a large amount of memory and computation power, also it may cause a classification algorithm to overfit to training samples and generalize poorly to new samples. Feature extraction is a general term for methods of constructing combinations of the variables to get around these problems while still describing the data with sufficient accuracy.

The EEG is typically described in terms of

(1) rhythmic activity

(2) transients

The rhythmic activity is divided into bands by frequency. To some degree, these frequency bands are a matter of nomenclature (i.e., any rhythmic activity between 8–12 Hz can be described as "alpha"), but these designations arose because rhythmic activity within a certain frequency range was noted to have a certain distribution over the scalp or a certain biological significance.

Some features of the EEG are transient rather than rhythmic. Spikes and sharp waves may represent seizure activity or interictal activity in individuals with epilepsy or a predisposition toward epilepsy. Other transient features are normal: vertex waves and sleep spindles are seen in normal sleep.

Most of the cerebral signal observed in the scalp EEG falls in the range of 1–20 Hz (activity below or above this range is likely to be artifactual, under standard clinical recording techniques). Waveforms are subdivided into bandwidths known as alpha, beta, theta, and delta to signify the majority of the EEG used in clinical practice.

Various bands of eeg data are->

- delta band (0.1-4hz)
- theta band (4-8hz)
- alpha band (8-12hz)

- beta band (12-30hz)
- low-gamma band (30-70hz)
- high-gamma band (70-180hz)

Features are sum of power in power spectrum of time series. Summed power in 6 bands ==> delta(0.1-4hz), theta(4-8hz), alpha(8-12hz),beta(12-30hz), low-gamma(30-70hz), high-gamma(70-180hz) . This is done for 1 minute segment of the time series.

```
# delta band (0.1-4hz)
ipos = (freq >= 0.1) & (freq < 4.0)
pib[0] = np.trapz(Pxx[ipos], freq[ipos])

# theta band (4-8hz)
ipos = (freq >= 4.0) & (freq < 8.0)
pib[1] = np.trapz(Pxx[ipos], freq[ipos])

# alpha band (8-12hz)
ipos = (freq >= 8.0) & (freq < 12.0)
pib[2] = np.trapz(Pxx[ipos], freq[ipos])

# beta band (12-30hz)
ipos = (freq >= 12.0) & (freq < 30.0)
pib[3] = np.trapz(Pxx[ipos], freq[ipos])

# low-gamma band (30-70hz)
ipos = (freq >= 30.0) & (freq < 70.0)
pib[4] = np.trapz(Pxx[ipos], freq[ipos])

# high-gamma band (70-180hz)
ipos = (freq >= 70.0) & (freq < 180.0)
pib[5] = np.trapz(Pxx[ipos], freq[ipos])

return pib
```

All the segments are 10 minute clippings of EEG data. They are sampled at sampling frequency of almost 400 Hz. We split the EEG data into 600 equal parts ( 10 mins * 60 sec = 600 sec ) . We iterate through the chunks and calculate summed spectral power of the band. This will give 6 features/1 minute for 1 channel or 96/minute for 16 channels. This will give us 960 features for 1 dataset.

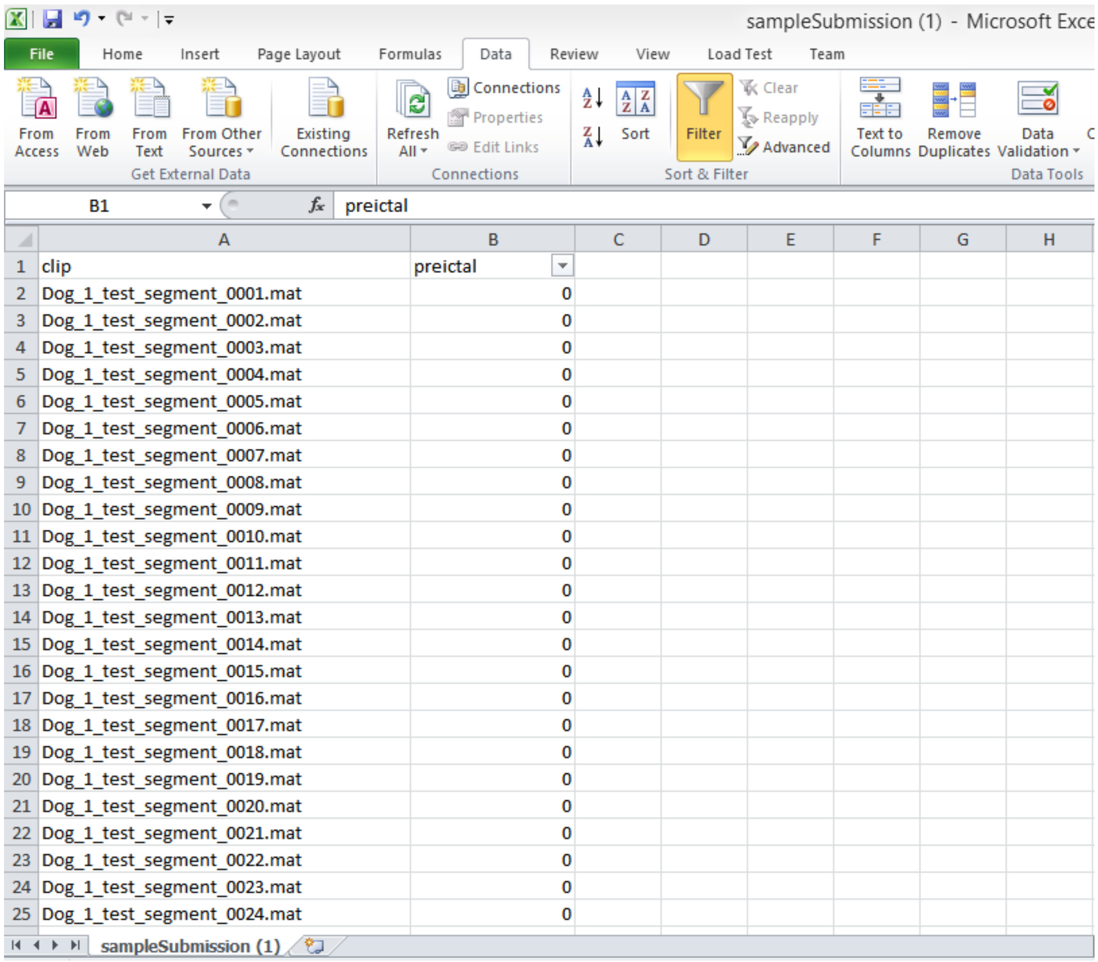
We have used glob function to list together the names of all the preictal, interictal and test segments. It uses a sort of pattern matching.

```
if __name__ == "__main__":
    preictal_files = glob("D:/Epilepsy/Dog_4/*preictal_segment*.mat")
    preictal_files=sorted(preictal_files)
    print(preictal_files[0])
    feature_arr_1 = main(preictal_files)

    pre_rows=np.shape(feature_arr_1)[0]
    a1=[1]*pre_rows

    interictal_files = glob("D:/Epilepsy/Dog_4/*interictal_segment*.mat")
    feature_arr_2 = main(interictal_files)

    inter_rows=np.shape(feature_arr_2)[0]
    a2=[0]*inter_rows

    X = np.concatenate((feature_arr_1, feature_arr_2),axis=0)
    Q = np.append(a1, a2)
```

We want to load the result of all the test segments for all the patients into a csv file which will be exported for finding the accuracy. So the results coming out from the working function have to be sorted according to the name of test segment. For this,we also return the name of eegfile and sort the list of features according to that factor.

```
# Create a pool of worker functions
pool = mp.Pool(n_cpus)
result = pool.map(worker, args)
#print(result.shape)
result=sorted(result,key=lambda x: x[1]) #sorting by the name of eeg file
# Feature array
for i in range(len(result)):
    nrecs = result[i][0].shape[1]
    print(result[i][1])
    feature_arr[i,:nrecs] = result[i][0]
```

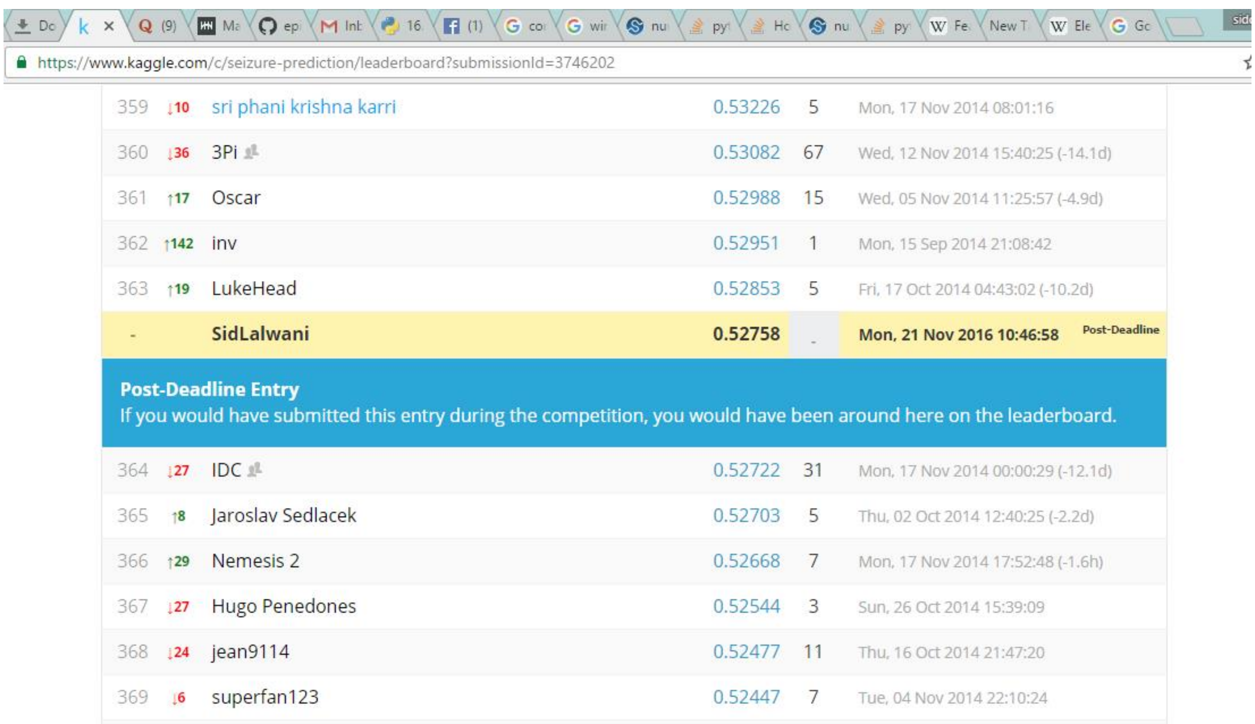
All the results are test segments are then saved in a csv file.



| clip | preictal |
|---|---|
| Dog_1_test_segment_0001.mat | 0 |
| Dog_1_test_segment_0002.mat | 0 |
| Dog_1_test_segment_0003.mat | 0 |
| Dog_1_test_segment_0004.mat | 0 |
| Dog_1_test_segment_0005.mat | 0 |
| Dog_1_test_segment_0006.mat | 0 |
| Dog_1_test_segment_0007.mat | 0 |
| Dog_1_test_segment_0008.mat | 0 |
| Dog_1_test_segment_0009.mat | 0 |
| Dog_1_test_segment_0010.mat | 0 |
| Dog_1_test_segment_0011.mat | 0 |
| Dog_1_test_segment_0012.mat | 0 |
| Dog_1_test_segment_0013.mat | 0 |
| Dog_1_test_segment_0014.mat | 0 |
| Dog_1_test_segment_0015.mat | 0 |
| Dog_1_test_segment_0016.mat | 0 |
| Dog_1_test_segment_0017.mat | 0 |
| Dog_1_test_segment_0018.mat | 0 |
| Dog_1_test_segment_0019.mat | 0 |
| Dog_1_test_segment_0020.mat | 0 |
| Dog_1_test_segment_0021.mat | 0 |
| Dog_1_test_segment_0022.mat | 0 |
| Dog_1_test_segment_0023.mat | 0 |
| Dog_1_test_segment_0024.mat | 0 |

# *Result*

Using feature extraction and decision tree classifier, algorithm got a score of 0.52758 on kaggle ranked (Post Deadline) 363 among 17000 entries.


https://www.kaggle.com/c/seizure-prediction/leaderboard?submissionId=3746202

| | | | | | |
|---|---|---|---|---|---|
| 359 | ↓10 | sri phani krishna karri | 0.53226 | 5 | Mon, 17 Nov 2014 08:01:16 |
| 360 | ↓36 | 3Pi | 0.53082 | 67 | Wed, 12 Nov 2014 15:40:25 (-14.1d) |
| 361 | ↑17 | Oscar | 0.52988 | 15 | Wed, 05 Nov 2014 11:25:57 (-4.9d) |
| 362 | ↑142 | inv | 0.52951 | 1 | Mon, 15 Sep 2014 21:08:42 |
| 363 | ↑19 | LukeHead | 0.52853 | 5 | Fri, 17 Oct 2014 04:43:02 (-10.2d) |
| - | | **SidLalwani** | **0.52758** | - | **Mon, 21 Nov 2016 10:46:58** Post-Deadline |

**Post-Deadline Entry**
If you would have submitted this entry during the competition, you would have been around here on the leaderboard.

| | | | | | |
|---|---|---|---|---|---|
| 364 | ↓27 | IDC | 0.52722 | 31 | Mon, 17 Nov 2014 00:00:29 (-12.1d) |
| 365 | ↑8 | Jaroslav Sedlacek | 0.52703 | 5 | Thu, 02 Oct 2014 12:40:25 (-2.2d) |
| 366 | ↑29 | Nemesis 2 | 0.52668 | 7 | Mon, 17 Nov 2014 17:52:48 (-1.6h) |
| 367 | ↓27 | Hugo Penedones | 0.52544 | 3 | Sun, 26 Oct 2014 15:39:09 |
| 368 | ↓24 | jean9114 | 0.52477 | 11 | Thu, 16 Oct 2014 21:47:20 |
| 369 | ↓6 | superfan123 | 0.52447 | 7 | Tue, 04 Nov 2014 22:10:24 |

# *The Way Forward*

The model so far is trained for a particular subject and has not been tested against other subjects. One of the main aim of the project is to see if we can develop a generic model for everyone suffering with epilepsy and and classify preictal state among interictal segments.
Following the work done so far, we can try to develop a model using the interictal and preictal data for a few subjects and try it against new subjects to see if respectable accuracy can be achieved in developing a generic model.
Moreover, various algorithms and preprocessing operations apart from feature extraction like sum of power in spectrum, can be done on the data to improve the performance and quality of model generation. For performance basis, we can reduce the sampling frequency of data and convert .mat EEG data to HDF5 format before importing in Python.
For improving the accuracy of model, we can use various new methods for feature extraction like-

- **Spectral Entropy**

Spectral entropy measures irregularity, complexity or amount of EEG disorders and has been proposed as indicator of anesthetic depth.

- **Hjorth parameters**

Hjorth Parameters are indicators of statistical properties used in signal processing in the time domain introduced by Bo Hjorth in 1970.[1] The parameters are Activity, Mobility, and Complexity. They are commonly used in the analysis of electroencephalography signals for feature extraction.

# *Bibliography*


1. Howbert JJ, Patterson EE, Stead SM, Brinkmann B, Vasoli V, Crepeau D, Vite CH, Sturges B, Ruedebusch V, Mavoori J, Leyde K, Sheffield WD, Litt B, Worrell GA (2014) Forecasting seizures in dogs with naturally occurring epilepsy. PLoS One 9(1):e81920.
2. Cook MJ, O'Brien TJ, Berkovic SF, Murphy M, Morokoff A, Fabinyi G, D'Souza W, Yerra R, Archer J, Litewka L, Hosking S, Lightfoot P, Ruedebusch V, Sheffield WD, Snyder D, Leyde K, Himes D (2013) Prediction of seizure likelihood with a long-term, implanted seizure advisory system in patients with drug-resistant epilepsy: a first-in-man study. LANCET NEUROL 12:563-571.
3. http://www.uc.pt/en/fctuc/dei/ensino/doctoral_program/colloquiumbrunodireit
4. https://www.ieeg.org/
5. Park Y, Luo L, Parhi KK, Netoff T (2011) Seizure prediction with spectral power of EEG using cost-sensitive support vector machines. Epilepsia 52:1761-1770.
6. http://brain.oxfordjournals.org/content/139/6/1713.full
7. https://www.kaggle.com/c/seizure-prediction