PREDICTION OF DISORDER OF BRAIN USING AI

Digital Signal Processing

Project Report

Gautham Krishna K 18BEC0207

Suryaa A M 18BEC0211

Nikhil Subramanian 18BEC0711

Varshini A 18BEC0839

June 2020

Acknowledgment

Our team would like to thank our Prof. Prakasam P for his continuous motivation and firm belief that he trusted in us to complete this project. This project report entitled "Prediction of disorder of Brain using AI" is submitted to Vellore Institute of Technology as apart of the J-component for the DSP subject. The contents of this project work have not have been submitted for any other CAL course. We take pride in the fact that our idea was thought of all by our own with little modification suggestions from our sir.

Abstract

Electroencephalography (EEG) records electrical activity of the brain. In this paper, a user-friendly platform using MATLAB with ML is employed for analysing EEG data and spotting patterns in it for fast diagnosis purposes. Firstly, the noise is excluded by using a Butterworth low pass and FIR bandpass filter and then evaluated to determine power spectral density. Several parameters (mean, variance, skewness and kurtosis) are extracted from the diagrams of power spectral density and contour of the peaks. We train our model using the algorithm called logistic regression with training data to develop a relationship between mean, kurtosis with the affected brain. Our model learns the pattern and can be used for diagnosis for any person in future.

TABLE OF CONTENTS

Introduction

The project we are doing is based on a published paper. It is the same paper with extra work put into it.

The Base paper used is Prediction of Disorder of Brain using EEG Signal Processing in MATLAB GUI Platform by Ratri Mukherjee, Shemul Sutra Dhar and Kusum Tara of Rajshahi University of Engineering & Technology published in International Conference on Electrical & Electronic Engineering (ICEEE).

In this paper, by using the Electroencephalography (EEG) record of a human brain and using power spectral density and statistical data like Mean, Variance, Skewness and kurtosis the disorder of the brain is identified. The EEG signal is filtered through Butterworth and Bandpass Filter. Then its power spectral density is calculated and also it's Mean, Variance, skewness and kurtosis is also calculated.

Our Project is to create a machine learning algorithm to take in EEG signal and predict the presence of any disorder in brain.

EEG

What is an EEG Signal?

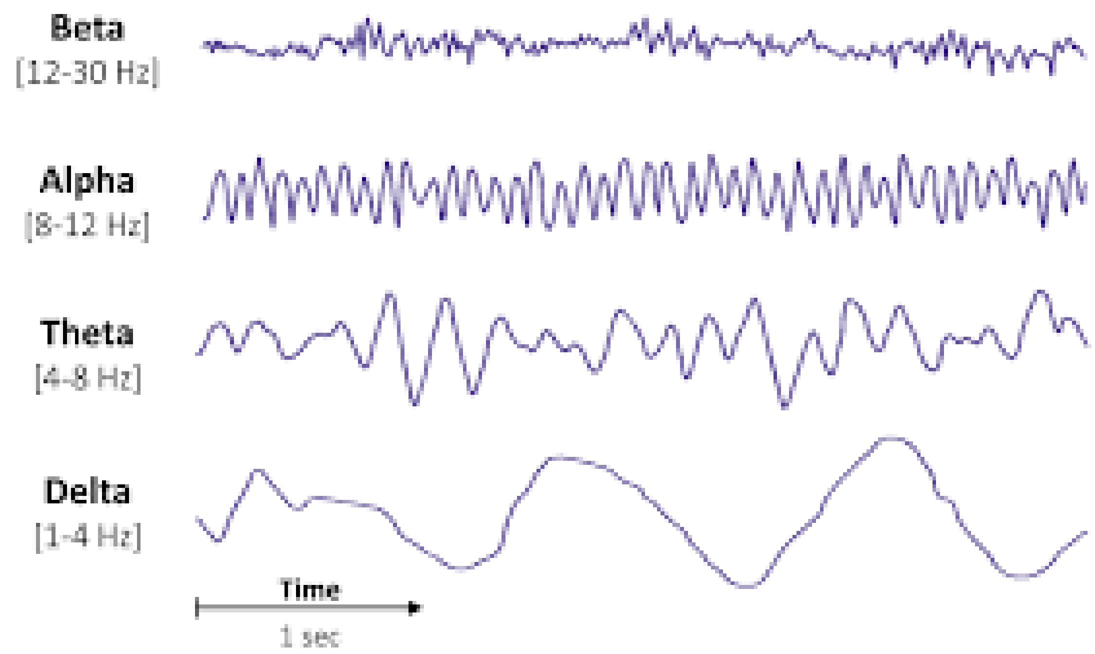
The Electroencephalogram (EEG) is a recording of the electrical activity of the brain from the scalp. The recorded waveforms reflect the cortical electrical activity. EEG activity is quite small and is measured in microvolts. The frequency is divided into four parts Delta, Theta, Alpha and Beta.

Delta: Frequency of 4 Hz and below. Tends to be the highest in amplitude.

Theta: Frequency of 4 to 8 Hz. It is classified as Slow activity.

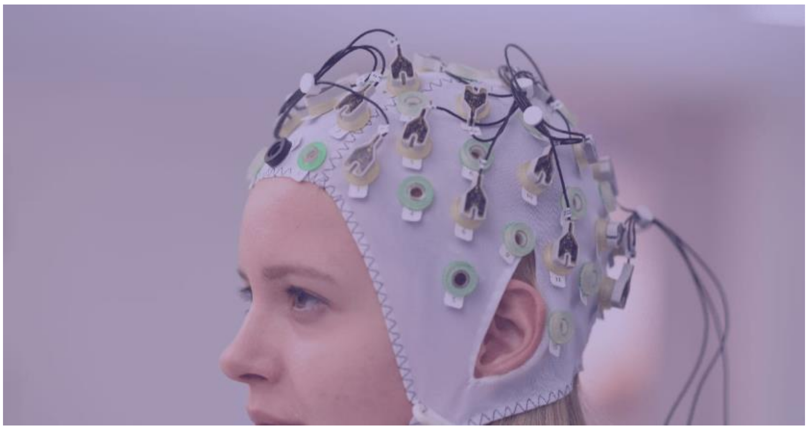
Alpha: Frequency of 8 to 12 Hz. Alerted when thinking or calculating.

Beta: Frequency of 12 Hz and above. Termed as Fast activity.

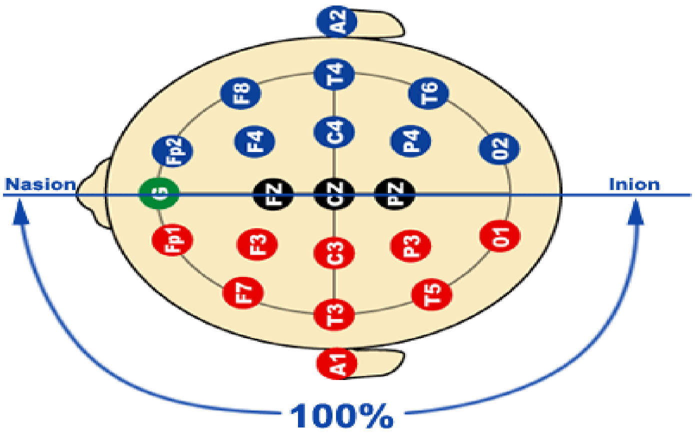


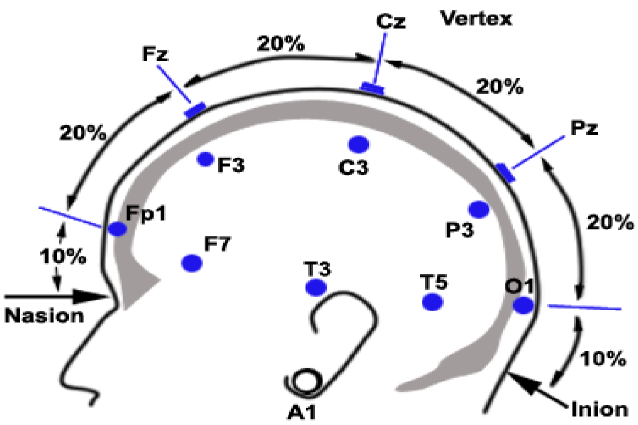
How to Record the EEG?

Small metal discs usually made of stainless steel, tin, gold or silver covered with a silver chloride coating. They are placed on the scalp in special positions. Each electrode site is labelled with a letter and a number. The letter refers to the area of brain underlying the electrode. Eg: F-Frontal lobe and T-Temporal lobe. Even numbers denote right side, odd numbers denote left side.



These electrodes are fitted into a cap and they are attached to the patients' scalp. This cap will record the electric signals in the brain and will be converted as EEG signal. An electrode gel is used to maximize the skin contact and allows for a low-resistance recording through the skin.The standardized placement of scalp electrodes for a classical EEG recording has become common since the adoption of the 10/20 system. The essence of this system is the distance in percentages of the 10/20 range between Nasion-Inion and fixed points. The midline electrodes are marked with a subscript z, which stands for zero.



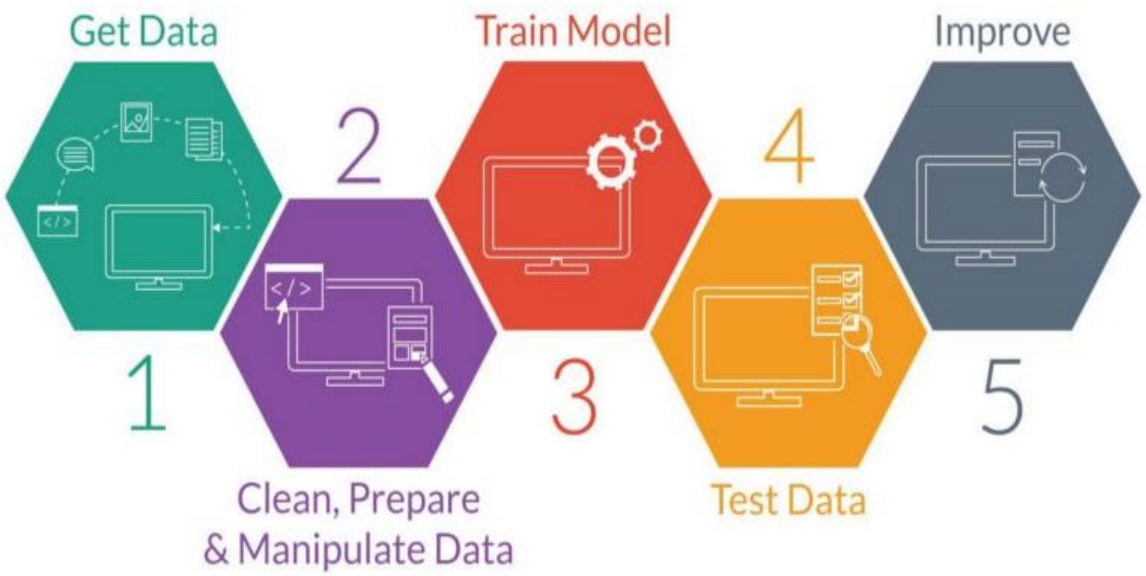


Machine Language

Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. Machine learning focuses on the development of computer programs that can access data and use it learn for themselves.

The process of learning begins with observations or data, such as examples, direct experience, or instruction, in order to look for patterns in data and make better decisions in the future based on the examples that we provide. The primary aim is to allow the computers learn automatically without human intervention or assistance and adjust actions accordingly.

But, using the classic algorithms of machine learning, text is considered as a sequence of keywords; instead, an approach based on semantic analysis mimics the human ability to understand the meaning of a text.



Methodology

To do our project we need a MATLAB code to get the statistical values from the EEG values. With the resultant values from the code we train an AI to predict if there is presence of disorder in brain.

First, the EEG data is taken in by the MATLAB code. It is filtered through butterworth and bandpass filters and then it is graphed. Then the Power Spectral density of the function is found. Then using statistical formulas, the mean, Variance, Skewness and kurtosis is found.

Mean:- $\mu_n = \frac{1}{N}\sum_{n=1}^{N-1} x_n$

Variance:- $V = \sum_{i=1}^{N} \frac{(x_i - \bar{x})^2}{N-1}$

Kurtosis:- $Kurt[X] = \frac{E\left[(X-\mu)^3\right]}{E\left[(X-\mu)^2\right]^2}$

Skewness:- $Skew[X] = E\left[\left(\frac{X-\mu}{\sigma}\right)^3\right] = \frac{E\left[X^3\right] - 3\mu\sigma^2 - \mu^3}{\sigma^3}$

The values obtained from this will be used to train the AI. The Sample of data used:

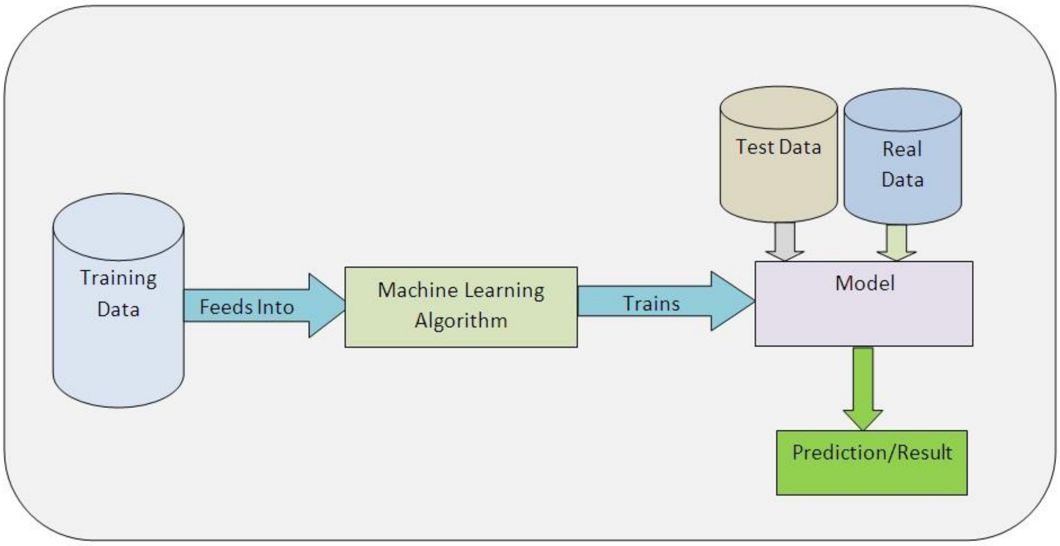
TABLE I. ANALYZED RESULTS OF EEG SIGNALS

Subjects	Statistical Parameters				
	Mean	*Variance*	*Kurtosis*	*Skewness*	*Status*
S1	732.25109	958423.86	6.28277	1.61845	Sound
S2	1382.548	4832538.9	6.8146	1.7681	Sound
S3	760.55799	1230025	6.10753	1.62294	Sound
S4	55639.7055	38210218	4.60547	1.03812	Affected
S5	164416.766	65245479	5.18188	1.77365	Affected
S6	1399.5519	4531829.9	6.83904	1.83543	Sound
S7	637.11056	1116823.5	6.14460	1.74440	Sound
S8	53541.8292	50840025	4.76716	1.27448	Affected
S9	118857.514	18337943	5.08950	1.23617	Affected
S10	17169.990	84669042	5.05027	1.56911	Affected

In order to predict if a patient has brain disorder or not, we will use the concepts of machine learning. We train the computer to find a certain logic based on the inputted values. The inputted values are called training sets and have both inputs and the expected outputs.

Since our output is binary, That is, it has the output 0 for not affected and 1 for affected, we will use the logistic regression algorithm. What this does is create a linear equation with the number of variables equal to number of input values. It then calculates what is called the cost function which is the error between the expected output and calculated output.

It then loops till it gets the right value of coefficients such that the cost function or error is minimized. Now when a new set is given as input, the system uses the coefficients calculated and predicts the output.

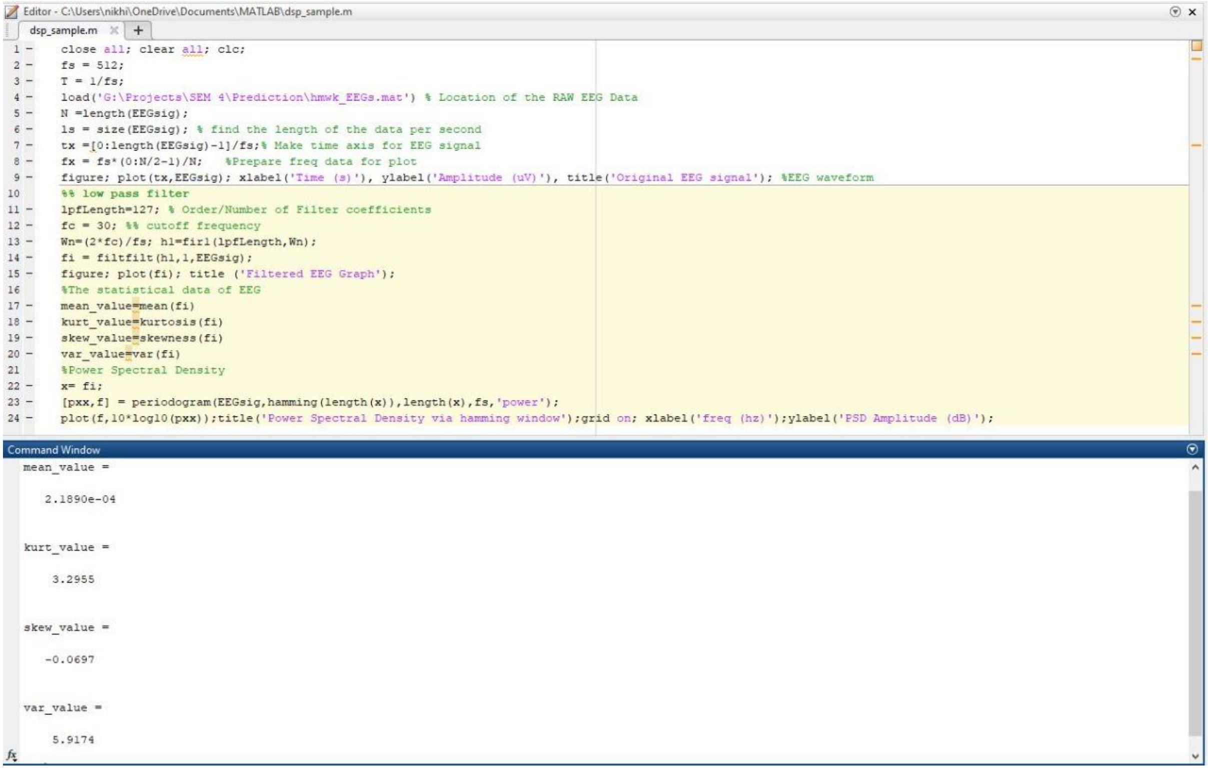


A Simple Machine Learning Pipeline Explanation

A Flowchart to illustrate the same

RESULT

The obtained EEG data is plotted as a signal, then the signal is filtered using a low pass filter. The filtered signal is further used to compute the mean, variance, kurtosis and skewness using the below code.



Editor - C:\Users\nikhi\OneDrive\Documents\MATLAB\dsp_sample.m

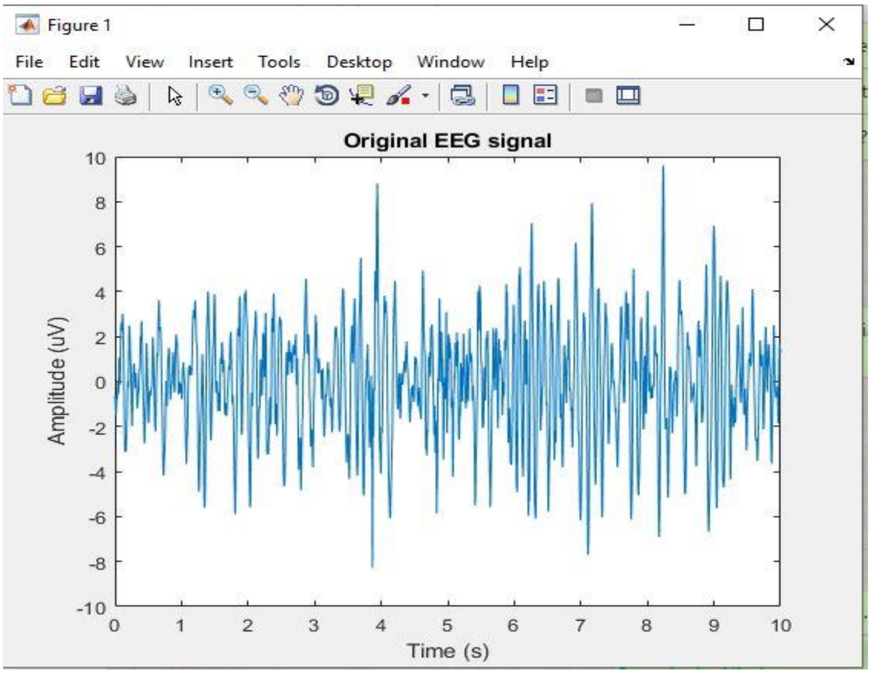
dsp_sample.m

```
close all; clear all; clc;
fs = 512;
T = 1/fs;
load('G:\Projects\SEM 4\Prediction\hmwk_EEGs.mat') % Location of the RAW EEG Data
N =length(EEGsig);
ls = size(EEGsig); % find the length of the data per second
tx =[0:length(EEGsig)-1]/fs;% Make time axis for EEG signal
fx = fs*(0:N/2-1)/N;   %Prepare freq data for plot
figure; plot(tx,EEGsig); xlabel('Time (s)'), ylabel('Amplitude (uV)'), title('Original EEG signal'); %EEG waveform
%% low pass filter
lpfLength=127; % Order/Number of Filter coefficients
fc = 30; %% cutoff frequency
Wn=(2*fc)/fs; h1=fir1(lpfLength,Wn);
fi = filtfilt(h1,1,EEGsig);
figure; plot(fi); title ('Filtered EEG Graph');
%The statistical data of EEG
mean_value=mean(fi)
kurt_value=kurtosis(fi)
skew_value=skewness(fi)
var_value=var(fi)
%Power Spectral Density
x= fi;
[pxx,f] = periodogram(EEGsig,hamming(length(x)),length(x),fs,'power');
plot(f,10*log10(pxx));title('Power Spectral Density via hamming window');grid on; xlabel('freq (hz)');ylabel('PSD Amplitude (dB)');
```

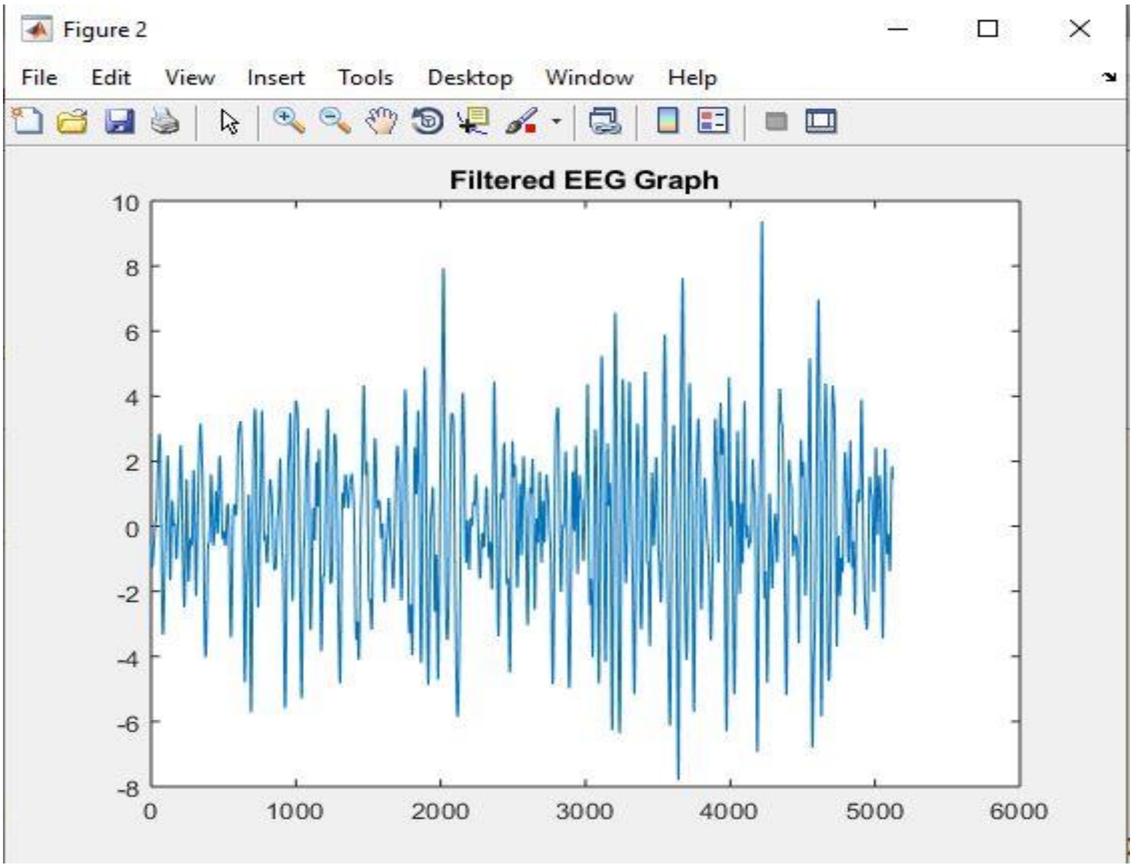
Command Window

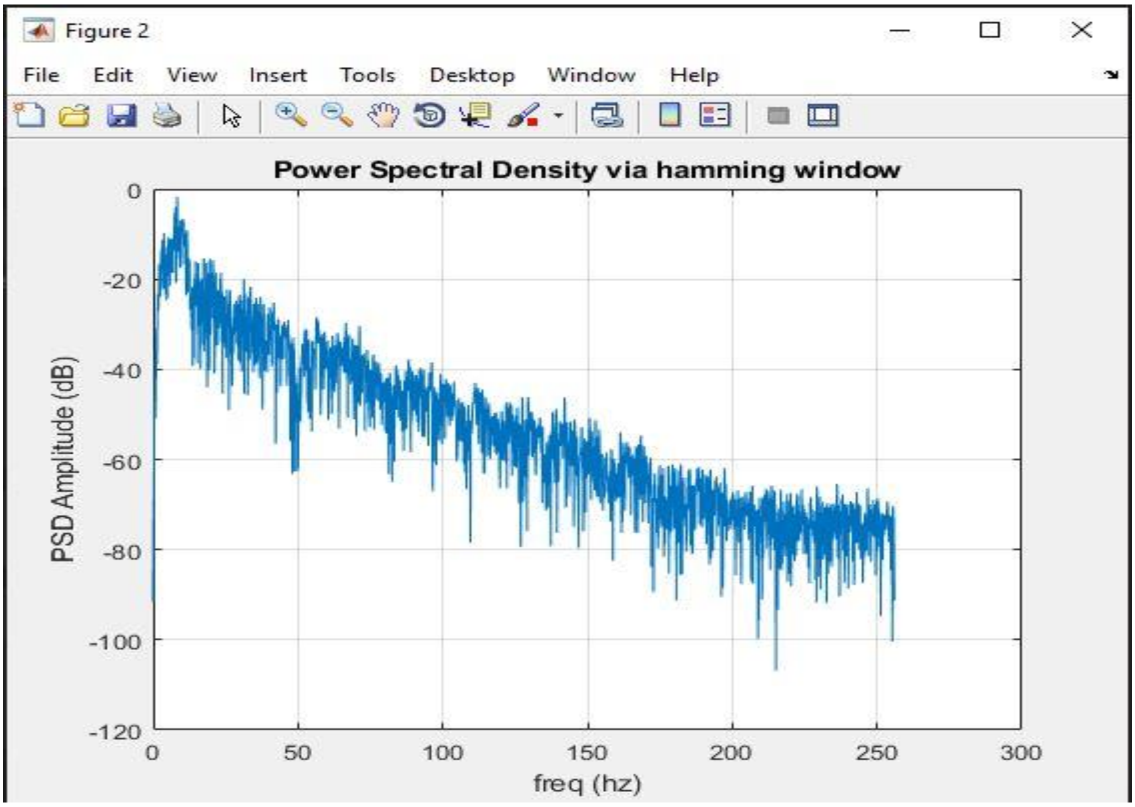
```
mean_value =

   2.1890e-04


kurt_value =

    3.2955


skew_value =

   -0.0697


var_value =

    5.9174
```

The signal is plotted from the EEG data of one sample.



The signal after passing through Low Pass Filter.



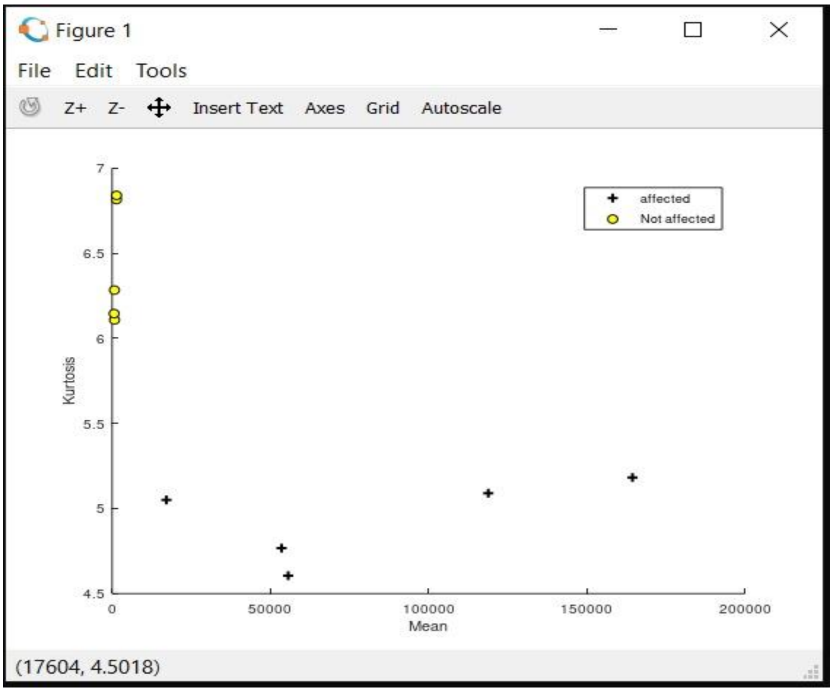


The following is the 10 sample data used in the research paper. We used the mean and kurtosis along with the status of the below table as training data to our algorithm.

TABLE I. ANALYZED RESULTS OF EEG SIGNALS

Subjects	Statistical Parameters				
	Mean	*Variance*	*Kurtosis*	*Skewness*	*Status*
S1	732.25109	958423.86	6.28277	1.61845	Sound
S2	1382.548	4832538.9	6.8146	1.7681	Sound
S3	760.55799	1230025	6.10753	1.62294	Sound
S4	55639.7055	38210218	4.60547	1.03812	Affected
S5	164416.766	65245479	5.18188	1.77365	Affected
S6	1399.5519	4531829.9	6.83904	1.83543	Sound
S7	637.11056	1116823.5	6.14460	1.74440	Sound
S8	53541.8292	50840025	4.76716	1.27448	Affected
S9	118857.514	18337943	5.08950	1.23617	Affected
S10	17169.990	84669042	5.05027	1.56911	Affected

We first visualised the training set data.



Cost function and parameter values obtained and the trained algorithm predicts with 0 as not affected and 1 as affected

```
Plotting data with + indicating (y = 1) examples and o indicating (y = 0) examples.
Cost at theta found by fminunc: 0.124554
theta:
 -0.031945
 0.000231
 -0.235424
p =

   0
   0
   0
   1
   1
   0
   0
   1
   1
   1

octave:5>
```

The theta shown above is the optimised by “fminunc” function which is used for hypothesis function. With this hypothesis function we can predict for any person if they are affected or not. Sice we did not get any extra data, we used the sample input without the output to our modal. It predicted either affected or not with 1 and 0 respectively. Hence, we can work on any unknown inputs.

Discussion:-

We had originally planned to do a project on brain disorder by processing the EEG datas obtained to output the EEG graph using Butterworth filters for removing noise, and obtaining the mean variance skewness and kurtosis. The teacher had suggested us to make the project a little more challenging by trying to predict if a patient in the future will have brain disorder or not. Although this could be done using the statistical values obtained, we decided to employ machine learning algorithms to make our project more diverse. We wrote a logistic regression algorithm code based on the online course we learnt and it was working well in predicting data. We encountered problems in predicting the current problem due to lack of data. Machine learning algorithms need sufficient amount of data to be able to make safe predictions of reasonable accuracy. We were able to gather data of only about 10 patients which resulted in underfitting and bad prediction. We wish to work on that problem in the future and obtain sufficient amount of actual patient data to create an accurate working model.

References

EEG DATA-

http://epileptologie-bonn.de/cms/upload/workgroup/lehnertz/eegdata.html

Theory of EEG Signal-
https://www.medicine.mcgill.ca/physio/vlab/biomed_signals/eeg_n.htm#:~:text=The%20electroencephalogram%20(EEG)%20is%20a,measured%20in%20microvolts%20(mV).

Theory of ML-

https://expertsystem.com/machine-learning-definition/#:~:text=Machine%20learning%20is%20an%20application,use%20it%20learn%20for%20themselves.

Research Paper:-

https://ieeexplore.ieee.org/document/8412847

Appendix – Source Code

MATLAB code for obtaining the statistical data of the EEG

```
close all; clear all; clc;
fs = 512;
T = 1/fs;
load('G:\Projects\SEM 4\Prediction\hmwk_EEGs.mat') % Location of the RAW
EEG Data
N =length(EEGsig);
ls = size(EEGsig); % find the length of the data per second
tx =[0:length(EEGsig)-1]/fs;% Make time axis for EEG signal
fx = fs*(0:N/2-1)/N;    %Prepare freq data for plot
figure; plot(tx,EEGsig); xlabel('Time (s)'), ylabel('Amplitude (uV)'),
title('Original EEG signal'); %EEG waveform
%% low pass filter
lpfLength=127; % Order/Number of Filter coefficients
fc = 30; %% cutoff frequency
Wn=(2*fc)/fs; h1=fir1(lpfLength,Wn);
fi = filtfilt(h1,1,EEGsig);
figure; plot(fi); title ('Filtered EEG Graph');
%The statistical data of EEG
mean_value=mean(fi)
kurt_value=kurtosis(fi)
skew_value=skewness(fi)
var_value=var(fi)
%Power Spectral Density
x= fi;
[pxx,f] = periodogram(EEGsig,hamming(length(x)),length(x),fs,'power');
plot(f,10*log10(pxx));title('Power Spectral Density via hamming
window');grid on; xlabel('freq (hz)');ylabel('PSD Amplitude (dB)');
```

Machine Learning Code

```
clear ; close all; clc
data = load('ex2data1.txt');
X = data(:, [1, 2]); y = data(:, 3);

% Visualizing the training data sets

fprintf(['Plotting data with + indicating (y = 1) examples and o ' ...
         'indicating (y = 0) examples.\n']);

plotData(X, y);
hold on;
xlabel('Mean')
ylabel('Kurtosis')
legend('affected', 'Not affected')
hold off;

% Using built-in function (fminunc) to find the optimal parameter theta for
the hypothesis function and its cost function

% options for fminunc

options = optimset('GradObj', 'on', 'MaxIter', 400);
[theta, cost] = ...
      fminunc(@(t)(costFunction(t, X, y)), initial_theta, options);
```

```
fprintf('Cost at theta found by fminunc: %f\n', cost);
fprintf('theta: \n');
fprintf(' %f \n', theta);
```

% Plotting Data

```
function plotData(X, y)

      figure; hold on;
      pos=find(y==1);
      neg=find(y==0);
      plot(X(pos,1),X(pos,2),'k+','LineWidth',2,'MarkerSize',7)
      plot(X(neg,1),X(neg,2),'ko','MarkerFaceColor','y','MarkerSize',7)
      hold off;
end
```

% Predicting for unknown samples

```
function p = predict(theta, X)
%PREDICT Predict whether the label is 0 or 1 using learned logistic
%regression parameters theta
%   p = PREDICT(theta, X) computes the predictions for X using a
%   threshold at 0.5 (i.e., if sigmoid(theta'*x) >= 0.5, predict 1)

m = size(X, 1); % Number of training examples
p = zeros(m, 1);
x=sigmoid(X*theta)

for i=1:m
  if x(i,1)>=0.5
     p(i,1)=1
  else
     p(i,1)=0
  end
end

end
```

% Sigmoid function

```
function g = sigmoid(z)
g = zeros(size(z));
g=1./(1+exp(-z))
end
```