University of Moratuwa

Faculty of Engineering

BM4151 Biosignal Processing

Project

Epileptic Seizure Detection in Continuous EEG

K. C. M. Madarasingha 130338V

W. N. D. Perera 130441F

Department of Electronic and Telecommunication Engineering

Date of Submission: 29/05/2017

Introduction

Seizure is a condition which occurs due to abnormal electrical activity in brain. Seizure results in uncontrolled convulsions of the body and sometimes unconsciousness. Epilepsy is the condition which is characterized by the recurrence of seizure. The severity of a seizure depends on the duration of the seizure and the physical environment the patient is in which could cause physical injuries. The causes of seizure are currently not known entirely, but there are several things which trigger he onset of a seizure including stroke, cancer, brain tumor, trauma, flashing lights, low blood sugar etc. In under-developed countries, seizure is quite common. Epileptic seizure detection and prediction has been a highly active research area for the last 10-12 years. The objective of this research is to increase the quality of life of the patients who are suffering from this condition by developing detection systems and early warning systems.

The abnormal activity of the brain is not similar in all the patients. The focal point of the seizure and the parameters of the abnormal activity is different from patient to patient. Hence, most recent researches have been conducted to develop patient specific epileptic seizure detection and prediction systems instead of developing a universal system. Epileptic seizure is analyzed using continuous EEG or ECoG (Electrocorticography). The non-stationarity of EEG signals makes it even more challenging in signal analysis. Epileptic seizure detection and prediction uses a variety of signal processing techniques such as spectral analysis, principle component analysis, wavelet decomposition and recently wavelet packet decomposition coupled with classifiers such as LDA classifiers, ANNs, k-NNs, random forest classifiers and SVM classifiers. The goal is to develop a classifier which detects and predicts seizure with a high accuracy and specificity. Research related to epileptic seizure detection are carried out by monitoring epilepsy patients or using datasets generated by previous studies.

In this project, our goal is to detect epileptic seizure using spectral and wavelet analysis and develop a patient specific classifier to detect seizure and non-seizure events. The EEG data required for the analysis is acquired by CHB-MIT Scalp EEG Database in physionet.org [1]. We will use signal processing techniques to preprocess the signals, use appropriate analysis tools to extract features and use statistical procedures for dimensionality reduction of data. Finally, we will evaluate the accuracy of the classifier and suggest whether the approach is appropriate for epileptic seizure detection.

Objectives of the Project

1. Detecting epileptic seizure events in continuous EEG
    - Use signal processing and analyzing techniques to process signals and extract features and identify seizure and non-seizure windows
    - Develop a classification algorithm to classify seizure and non-seizure time windows with high accuracy
2. Use dimensionality reduction while maintaining accuracy for efficient and robust computation
3. Use manual and automatic feature extraction methods
4. Minimize the number of false detections of seizure events and detect seizure onset with minimum latency

Acknowledgement of Prior Research

Several researches have been conducted throughout the past decade in seizure detection and prediction. Some of most commonly used methods are spectral analysis, wavelet analysis and principle component analysis. Significant accuracies (typically around 90%) have been achieved in these researches with low false detection rates ($0.1hr^{-1}$) with low latency onset detection (3 seconds). Most of the researches use feature spaces of higher dimensions which include features such as mean power, maximum amplitude, entropies etc.

Some research has focused on developing patient specific seizure detection systems while the others have developed patient specific detection systems. Furthermore, going beyond detecting seizure and non-seizure events, several researches have been conducted to identify stages of seizure such as ictal, interictal etc.

Some of the signal analysis techniques used here are based on prior research and minor improvements of them so that higher classification accuracies could be achieved. We will compare two signal analysis techniques in this project along with feature extraction strategies and discuss about their performance and accuracy.

Seizure Detection by Spectral Analysis

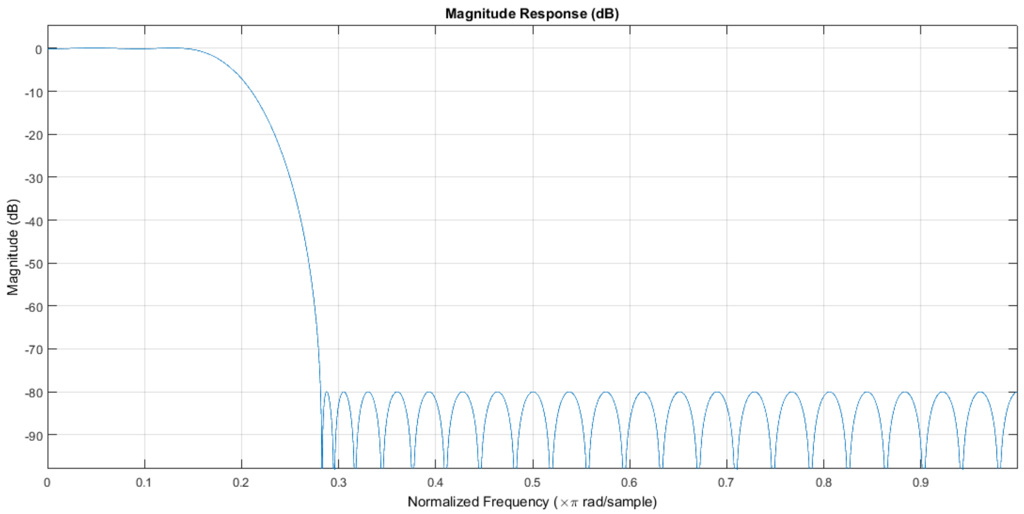
The first method used in this research is spectral analysis technique. Typically, the bandwidth of interest in EEG is 0-30Hz which comprises of alpha, beta, theta, mu and delta bands. Hence the EEG recordings should be bandlimited to this band. Prior research has suggested that the band powers of 0-16Hz and 16-25Hz can be used to distinguish between seizure and non-seizure events [2]. The complete algorithm implemented on MATLAB is given in Appendix 1.

Preprocessing

Since the EEG recordings should be bandlimited to 0-25Hz, a low pass filter with the following parameters were used.

```
N      = 50;      % Order
Fc     = 25;      % Cutoff Frequency
DpassU = 0.01;    % Upper Passband Ripple
DpassL = 0.01;    % Lower Passband Ripple
DstopU = 0.0001;  % Upper Stopband Attenuation
DstopL = 0.0001;  % Lower Stopband Attenuation
```

The frequency response of the filter is shown below.



Since the 50Hz frequency component has already been suppressed by the low pass filter, there was no requirement for a separate notch filter of 50Hz.

Feature Extraction

1. Spatial Feature Extraction

Since the EEGs obtained were comprised of 23 channels, in order to reduce the dimensionality of the feature space, only several channels were selected. These were selected by visualizing EEG data on EEGlab and singling out channels which show a greater variability in the time domain. Even though it may not be statistically accurate, the results which will be presented later suggests that it could be accepted. 16 channels were selected in this manner for the first patient.

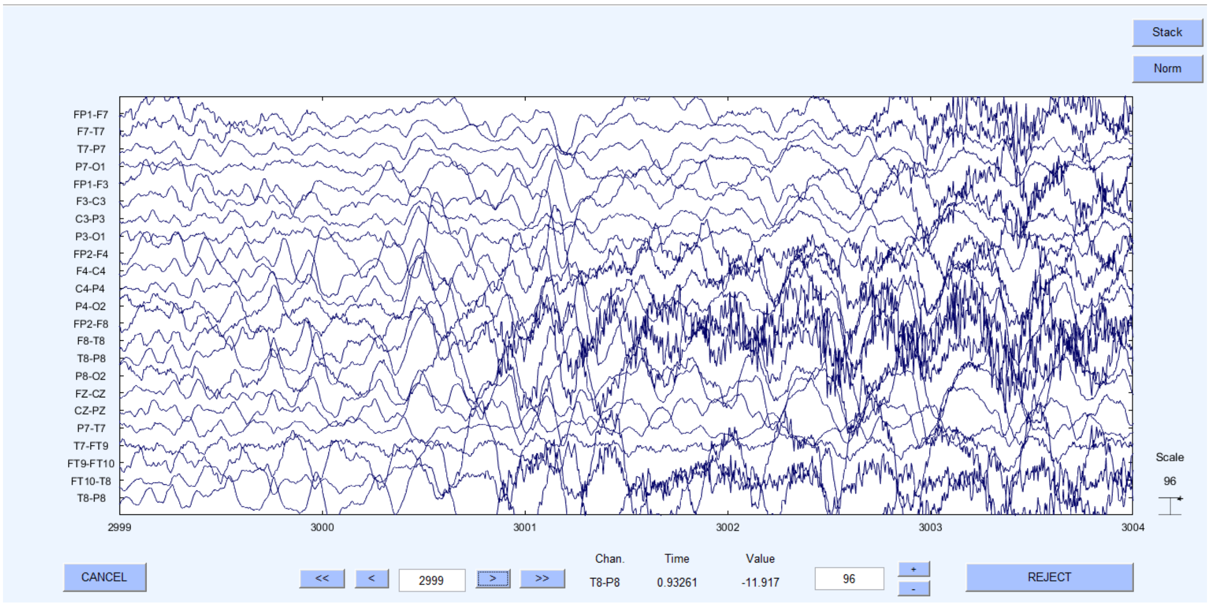


Figure 1: EEG recording at the onset of seizure in the patient

2. Frequency Feature Extraction

The features of interest were the powers of the frequency bands, 0-16Hz and 16-25Hz. This was performed on the 16 channels selected earlier which resulted in 32 features.

3. Temporal Feature Extraction

Temporal features of the EEG recordings are extracted using a sliding, non-overlapping window of 4 seconds (1024 samples). Spatial and frequency features of each window are extracted and saved in a feature matrix.

Classification and Results

An SVM classifier is trained with Radial Basis Function as the kernel under 3 conditions with varying number of training samples. During all these instances, 1000 non-seizure events were incorporated for training the classifier. The performance parameters and results obtained are as follows.

Patient 02 data was not used as there were only 2 seizure events throughout all the data.

Training accuracy – Accuracy of prediction of seizures when testing on data which have been trained on

False detections – Event of 2 consecutive windows being detected as seizures

Average latency – Average time elapsed for the identification of seizure

Statistics for Patient 01

- Number of seizure windows – 65
- Training data – 20 hours
- Number of prominent channels – 16

1. Training using all seizure events
   - 58 out of 65 classified as seizures – 90% of accuracy

     Windows corresponding to the onset of the seizure have been misclassified and a window or two within the seizure event have been misclassified.

   - Average of 8 false detections of onset – $0.4h^{-1}$
   - Average latency of 5 seconds
2. Cross validation with 46% of training data (30 windows) and rest for validation
   - 22 out of 35 classified as seizures – 62% of accuracy
   - Average of 2 false detections of onset – $0.1h^{-1}$
   - Average latency of 5 seconds
3. Cross validation with 63% of training data (41 windows) and rest for validation
   - 17 out of 24 classified as seizures – 70% of accuracy
   - Average of 4 false detections of onset – $0.2h^{-1}$
   - Average latency of 4 seconds

Statistics for Patient 03

- Number of seizure windows – 84
- Training data – 20 hours
- Number of prominent channels – 13

1. Training using all seizure events
    - 79 out of 84 classified as seizures – 94% of accuracy
    - Average of 4 false detections of onset – $0.2h^{-1}$
    - Average latency of 3.3 seconds
2. Cross validation with 41% of training data (35 windows) and rest for validation
    - 34 out of 49 classified as seizures – 70% of accuracy
    - Average of 6 false detections of onset – $0.3h^{-1}$
    - Average latency of 8 seconds
3. Cross validation with 73% of training data (61 windows) and rest for validation
    - 19 out of 23 classified as seizures – 82% of accuracy
    - Average of 6 false detections of onset – $0.3h^{-1}$
    - Average latency of 2 seconds

Seizure Detection by Wavelet Analysis

Wavelet transform has been the most prominent approach in recent research related to seizure detection. Continuous wavelet transform is computationally intensive. Hence, Discrete Wavelet Transform or DWT was introduced to overcome this issue. Later, a variation of DWT was introduced called Discrete Wavelet Packet Transform or DWPT which displays for frequency details compared to DWT. Research has been conducted using different wavelet families [3]. Hence, the most frequently used wavelet family, Daubechies wavelet family was used in this project.

Furthermore, several features such as wavelet packet power, peak coefficient, entropy, etc. have been used in prior research. However, in this project we consider the wavelet packet power as the features and significant features are extracted using one-way ANOVA test.

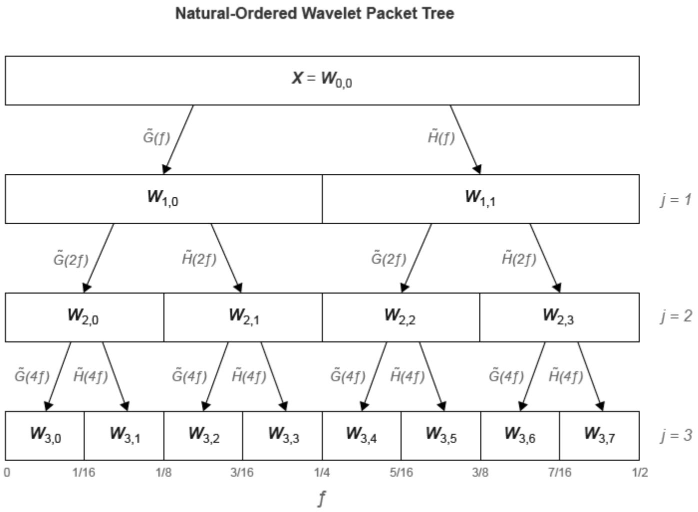


Figure 2: Wavelet Packet Decomposition

<u>Preprocessing</u>

Preprocessing is similar to the previous method with low pass filtering with same filter parameters.

<u>Spatial and Frequency Feature Extraction</u>

The significant channels and the corresponding wavelet packets are determined by one-way ANOVA test. The test is done between samples from seizure samples and non-seizure samples and the four features which give rise to smallest p value (higher significance) are considered for classification. For each feature, 10 non-seizure samples and 5 seizure samples were considered and p values were calculated between every pair consisting of one seizure and non-seizure event. The features with most number of p-values less than 0.01 were considered as the significant features. The algorithm implemented to extract significant features is shown in Appendix 2.

<u>Temporal Feature Extraction</u>

Just as it was done in spectral analysis, a sliding, non-overlapping window of length of 4s was used for temporal feature extraction.

<u>Classification and Results</u>

Similar approach to spectral analysis was employed. Same quantities were used for training and validation to compare the performance of the classifier. The algorithm implemented for spectral analysis is given in Appendix 3.

Statistics for Patient 01

1. Training using all seizure events
    - 58 out of 65 classified as seizures – 90% of accuracy

      Windows corresponding to the onset of the seizure have been misclassified and a window or two within the seizure event have been misclassified.

    - Average of 7 false detections of onset – $0.35h^{-1}$
    - Average latency of 2.4 seconds
2. Cross validation with 52% of training data (34 windows) and rest for validation
    - 28 out of 31 classified as seizures – 90% of accuracy
    - Average of 8 false detections of onset – $0.4h^{-1}$
    - 0 latency – Seizure is detected at the start
3. Cross validation with 70% of training data (45 windows) and rest for validation
    - 18 out of 20 classified as seizures – 90% of accuracy
    - Average of 4 false detections of onset – $0.2h^{-1}$
    - 0 latency – Seizure is detected at the start

Statistics for Patient 03

1. Training using all seizure events
    - 82 out of 84 classified as seizures – 97% of accuracy
    - > 1 false detections per hour
    - Average latency of 1.33 seconds
2. Cross validation with 58% of training data (49 windows) and rest for validation
    - 31 out of 35 classified as seizures – 88% of accuracy with 97% maximum accuracy
    - > 1 false detections per hour
    - Average latency of 2.66 seconds
3. Cross validation with 75% of training data (63 windows) and rest for validation
    - 19 out of 21 classified as seizures – 90% of accuracy with 95% maximum accuracy
    - > 1 false detections per hour
    - Average latency of 2 seconds

Challenges During Implementation

1. The first issue is the large amount of data which should be read by the program in order to compute the features. One record is about 400MB in size and takes a large duration to read. The program was run on a laptop with an Intel ™ i7 4th generation processor. Hence, this was overcome by exploiting the parallel processing power of the laptop using parfor in MATLAB. This reduced the time required to read the data by a significant factor.
2. Large feature space which results in memory issues when processing was another issue. This was overcome by using a non-overlapping window but it compromised the accuracy to a small extent and latency to a large extent.
3. Initially, a linear kernel function was used to train the classifier, but the accuracy was very low, hence RBF kernel was used which resulted in very high accuracies.
4. The large ratio between the non-seizure data and seizure data was another issue while training the classifier. Therefore, only a 1000 non-seizure data was used with seizure data to train the classifier. This too resulted in higher accuracies.

Discussion

The results generated by the classifiers have proven to be very effective and is in par with the accuracies achieved by prior research. Furthermore, cross validation accuracies are also higher and prove to be accurate. When comparing the two methods, it is evident that spectral analysis technique uses almost 4 times the features as wavelet transform. However, despite the sparse number of features used for wavelet analysis, it proves to be the more accurate approach for seizure detection. It can be concluded that through a reliable statistical method, such as ANOVA, dimensions can be reduced, while maintaining a high level of accuracy.

The issue in spectral analysis is the high latency in detecting seizure. This should be typically 1-2 seconds. But it exceeds 3 seconds in almost all the cases in spectral analysis. However, it is significantly less (around 2.5 seconds) in wavelet analysis. This could be reduced to a greater extent had an overlapping window been used.

The false detection rate is a tad below par with most research conducted ($0.1hr^{-1}$) but this is accentuated in wavelet analysis for patient 3. There is an unacceptable level of false detection despite the high accuracy.

Overall, this project can be deemed to be successful in terms of accuracy and latency, and low false detection to some extent. There is still more room for improvement in terms of latency and false detection.

Further Improvements

- Use other classifying methods such as ANN, random forest classifier, k-NN etc.
- Further decrease features and maintain accuracy
- Achieve even more accuracy, ideally 100%
- Introduce a correlation based feature extraction method – This approach is expected to reduce the number of training data required to train the classifier
- Test for variable window sizes and overlapping windows to reduce latency

Appendix

Appendix 1 – Spectral Analysis Code

```
close all
clear

% ----------------------------Seizure occurences--------------------------

% File Name: chb03_01.edf
% File Start Time: 13:23:36
% File End Time: 14:23:36
% Number of Seizures in File: 1
% Seizure Start Time: 362 seconds  - 91
% Seizure End Time: 414 seconds - 104

% File Name: chb03_02.edf
% File Start Time: 14:23:39
% File End Time: 15:23:39
% Number of Seizures in File: 1
% Seizure Start Time: 731 seconds - 183
% Seizure End Time: 796 seconds - 199

% File Name: chb03_03.edf
% File Start Time: 15:23:47
% File End Time: 16:23:47
% Number of Seizures in File: 1
% Seizure Start Time: 432 seconds - 108
% Seizure End Time: 501 seconds - 125

% File Name: chb03_04.edf
% File Start Time: 16:23:54
% File End Time: 17:23:54
% Number of Seizures in File: 1
% Seizure Start Time: 2162 seconds - 541
% Seizure End Time: 2214 seconds - 554

% File Name: chb03_05.edf
% File Start Time: 01:51:23
% File End Time: 2:51:23
% Number of Seizures in File: 1
% Seizure Start Time: 1982 seconds - 496
% Seizure End Time: 2029 seconds - 507

% File Name: chb03_06.edf
% File Start Time: 04:51:45
% File End Time: 5:51:45
% Number of Seizures in File: 1
% Seizure Start Time: 1725 seconds - 437
% Seizure End Time: 1778 seconds - 445

% Extraction of spatial features
% Prominent channels are selected
% 1, 2, 3, 4, 5, 6, 9, 13, 19, 20, 21, 22, 23

% Indices corresponding to seizure events
```

```
seizureInd = [(91:104) 900*1+(183:199) 900*2+(108:125) 900*3+(541:554)...
    900*4+(496:507) 900*5+(437:445)];

% Reading data and generating feature matrices
featureVector1 = generateSpectralFeat(0);
featureVector2 = generateSpectralFeat(10);

featureVector = [featureVector1 featureVector2];
```

Preparing the training dataset and target vectors

```
targetVectorNonSeiz = zeros(1, size(featureVector, 2) - length(seizureInd))';
targetVectorSeiz = ones(1, length(seizureInd))';

featureVectorSeiz = featureVector(:,seizureInd);
featureVectorSeiz = featureVectorSeiz';
featureVectorNonSeiz = featureVector;
featureVectorNonSeiz(:,seizureInd) = [];
featureVectorNonSeiz = featureVectorNonSeiz';
```

Create and train SVM model

Randomly select 1000 samples from non-seizure data

```
randSampleInd = randi(size(featureVectorNonSeiz, 1), 1, 1000);
finalFeatureVector = [featureVectorSeiz;featureVectorNonSeiz(randSampleInd,:)];
finalTargetVector = [targetVectorSeiz;targetVectorNonSeiz(randSampleInd,:)];

SVMmodel = fitcsvm(finalFeatureVector, finalTargetVector, 'Standardize',...
    true,'KernelFunction','RBF','KernelScale','auto');
```

Read EEG Data for patient 01 (In blocks of 10 records)

This function reads blocks of 10 records and a low pass filtering is applied and only channels of interest specified by the user is considered

```
function featureVector = generateSpectralFeat(index)

parfor i = 1:10
    fileName = sprintf('chb03/chb03_%d.edf', i+index);
    [~,records(:,:,i)] = edfread(fileName);
end

% Prominent channels for patient 01
promChannels = [1, 2, 3, 4, 5, 6, 9, 13, 19, 20, 21, 22, 23];

% Each channel is represented by a column for filtering
data = permute(records, [2, 1, 3]);
data = data(:,promChannels,:);

% Create low pass filter
Fs = 256;  % Sampling Frequency

N      = 50;      % Order
Fc     = 25;      % Cutoff Frequency
```

```
DpassU = 0.01;    % Upper Passband Ripple
DpassL = 0.01;    % Lower Passband Ripple
DstopU = 0.0001;  % Upper Stopband Attenuation
DstopL = 0.0001;  % Lower Stopband Attenuation

% Calculate the coefficients using the FIRCLS function.
b  = fircls(N, [0 Fc Fs/2]/(Fs/2), [1 0], [1+DpassU DstopU], [1-DpassL ...
    -DstopL]);
Hd = dfilt.dffir(b);

LPFiltered = filter(Hd, data);

featureVector = [];

% Perform Short Time Fourier Transform for time intervals of 4s
for i = 1:size(LPFiltered, 3)
    for j = 1:size(LPFiltered, 1)/1024
        [pxx, f] = periodogram(LPFiltered(1 + (1024*(j - 1)):1024*j,:,i),...
            [], [], 256);
        power0_16 = bandpower(pxx(1:65, :), f(1:65), 'psd');
        power16_25 = bandpower(pxx(65:101, :), f(65:101), 'psd');
        powerVector = [power0_16 power16_25];
        featureVector = [featureVector powerVector'];
    end
end
end
```

Published with MATLAB® R2016b

Appendix 2 – Extract Significant Features

```
%-------------------Read EEG Data for patient 03--------------------------
% Read first batch of data
parfor i = 1:10
    fileName = sprintf('chb03/chb03_%d.edf', i);
    [~,records(:,:,i)] = edfread(fileName);
end
```

Each channel is represented by a column for filtering Only channel with seizure data are considered, therefore indexing used will be from 1-6 later

```
records = records(:,:,[1 2 3 4 5 6]);
data = permute(records, [2, 1, 3]);

% Indices corresponding to seizure events
% Patient 01
% seizureInd = [900*2+(749:759) 900*3+(367:373) 900*14+(433:443)...
%     900*15+(254:266) 900*17+(430:452)];

% Indices corresponding to seizure events
% Patient 03
% seizureInd = [(91:104) 900*1+(183:199) 900*2+(108:125) 900*3+(541:554)...
%     900*4+(496:507) 900*5+(437:445)];

% Create low pass filter
```

```
Fs = 256;  % Sampling Frequency

N      = 50;      % Order
Fc     = 25;      % Cutoff Frequency
DpassU = 0.01;    % Upper Passband Ripple
DpassL = 0.01;    % Lower Passband Ripple
DstopU = 0.0001;  % Upper Stopband Attenuation
DstopL = 0.0001;  % Lower Stopband Attenuation

% Calculate the coefficients using the FIRCLS function.
b  = fircls(N, [0 Fc Fs/2]/(Fs/2), [1 0], [1+DpassU DstopU], [1-DpassL ...
            -DstopL]);
Hd = dfilt.dffir(b);

LPFiltered = filter(Hd, data);
records = permute(LPFiltered,[2 1 3]);
```

Consider samples of seizure and non-seizure events

```
Fs = 256;

seiz1 = records(:,362*Fs+1:414*Fs,1);
seiz2 = records(:,731*Fs+1:796*Fs,2);
seiz3 = records(:,432*Fs+1:501*Fs,3);
seiz4 = records(:,2162*Fs+1:2214*Fs,4);
seiz5 = records(:,1982*Fs+1:2029*Fs,5);
seiz6 = records(:,1725*Fs+1:1778*Fs,6);

seiz(:,:,1) =seiz1(:,1:12032);
seiz(:,:,2) =seiz2(:,1:12032);
seiz(:,:,3) =seiz3(:,1:12032);
seiz(:,:,4) =seiz4(:,1:12032);
seiz(:,:,5) =seiz5(:,1:12032);
seiz(:,:,6) =seiz6(:,1:12032);

nonSeiz(:,:,1) = records(:,1:47*Fs,1);
nonSeiz(:,:,2) = records(:,1500*Fs+1:1547*Fs,1);
nonSeiz(:,:,3) = records(:,100*Fs+1:147*Fs,2);
nonSeiz(:,:,4) = records(:,2000*Fs+1:2047*Fs,2);
nonSeiz(:,:,5) = records(:,100*Fs+1:147*Fs,3);
nonSeiz(:,:,6) = records(:,2100*Fs+1:2147*Fs,3);
nonSeiz(:,:,7) = records(:,1:47*Fs,4);
nonSeiz(:,:,8) = records(:,2500*Fs+1:2547*Fs,4);
nonSeiz(:,:,9) = records(:,700*Fs+1:747*Fs,5);
nonSeiz(:,:,10) = records(:,2500*Fs+1:2547*Fs,5);
nonSeiz(:,:,11) = records(:,1000*Fs+1:1047*Fs,6);
nonSeiz(:,:,12) = records(:,2000*Fs+1:2047*Fs,6);
```

For non seizure data - Feature computation

```
wdSize = 1024;
level = 5;
wName = 'db4';

for r = 1:size(nonSeiz, 3)
    for j = 1:size(nonSeiz, 2) / wdSize
```

```
        for k = 1:size(nonSeiz, 1)
            signal = nonSeiz(k,1 + (wdSize * (j - 1)):wdSize * j, r);
            wpt = wpdec(signal,level,wName);
            for m = 1:2^level
                wpCoef = wpcoef(wpt, [level, m - 1]);
                nonSeizFeatures(j, k, m, r) = log(sqrt(mean(wpCoef.*wpCoef)));
            end
        end
    end
end
```

For seizure data - Feature computation

```
for r = 1:size(seiz, 3)
    for j = 1:size(seiz, 2) / wdSize
        for k = 1:size(seiz, 1)
            signal = seiz(k,1 + (wdSize * (j - 1)):wdSize * j, r);
            wpt = wpdec(signal,level,wName);
            for m = 1:2^level
                wpCoef = wpcoef(wpt, [level, m - 1]);
                seizFeatures(j, k, m, r) = log(sqrt(mean(wpCoef.*wpCoef)));
            end
        end
    end
end
```

Anova testing for each feature

A matrix of p values is generated for each feature and the feature with maximum number of p values less than 0.001 are considered as significant features

```
for i = 1:1:size(seizFeatures, 2)
    for j = 1:1:size(seizFeatures, 3)
        p = [];
        for k = 1:1:size(nonSeizFeatures, 4)
            for h = 1:1:size(seizFeatures, 4)
                p(k, h) = anova1([nonSeizFeatures(:,i, j, k)...
                    seizFeatures(:,i, j, h)], [], 'off');
                close all
            end
        end
        signifMatrix = p < 0.001;
        sigScore(i, j) = sum(sum(signifMatrix));
    end
end
```

Published with MATLAB® R2016b

Appendix 3 – Wavelet Analysis Code


```
%%--------------------Seizure detection in continuous EEG-----------------
% In this method, seizure events are detected and classified using discrete
% wavelet packet transform.
% Seizure events of the ECG are identified by considering 4 second
% intervals in the EEG signal. Prominent channels and the wavelet packet
% are extracted by significance.m based on a one way ANOVA test

% ----------------------------Seizure occurences--------------------------

% File Name: chb03_01.edf
% File Start Time: 13:23:36
% File End Time: 14:23:36
% Number of Seizures in File: 1
% Seizure Start Time: 362 seconds  - 91
% Seizure End Time: 414 seconds - 104

% File Name: chb03_02.edf
% File Start Time: 14:23:39
% File End Time: 15:23:39
% Number of Seizures in File: 1
% Seizure Start Time: 731 seconds - 183
% Seizure End Time: 796 seconds - 199

% File Name: chb03_03.edf
% File Start Time: 15:23:47
% File End Time: 16:23:47
% Number of Seizures in File: 1
% Seizure Start Time: 432 seconds - 108
% Seizure End Time: 501 seconds - 125

% File Name: chb03_04.edf
% File Start Time: 16:23:54
% File End Time: 17:23:54
% Number of Seizures in File: 1
% Seizure Start Time: 2162 seconds - 541
% Seizure End Time: 2214 seconds - 554

% File Name: chb03_05.edf
% File Start Time: 01:51:23
% File End Time: 2:51:23
% Number of Seizures in File: 1
% Seizure Start Time: 1982 seconds - 496
% Seizure End Time: 2029 seconds - 507

% File Name: chb03_06.edf
% File Start Time: 04:51:45
% File End Time: 5:51:45
% Number of Seizures in File: 1
% Seizure Start Time: 1725 seconds - 437
% Seizure End Time: 1778 seconds - 445
```

```
% Prominent channels for the first patient
% Channels 2, 19, 20, 21
% Corresponding wavelet packets 2, 2, 2, 2

clear

% Indices corresponding to seizure events
seizureInd = [(91:104) 900*1+(183:199) 900*2+(108:125) 900*3+(541:554)...
    900*4+(496:507) 900*5+(437:445)];
```

Preparing feature vector and targets

```
featMatrix1 = generateFeatMat(0);
featMatrix2 = generateFeatMat(10);

featMatrix = cat(3, featMatrix1, featMatrix2);
featMatrix2D = reshape(featMatrix, size(featMatrix, 1), ...
    size(featMatrix, 2)*size(featMatrix, 3));

% Preparing the training dataset and target vectors
targetVectorNonSeiz = zeros(1, size(featMatrix2D, 2) - length(seizureInd))';
targetVectorSeiz = ones(1, length(seizureInd))';

featureVectorSeiz = featMatrix2D(:,seizureInd);
featureVectorSeiz = featureVectorSeiz';

featureVectorNonSeiz = featMatrix2D;
featureVectorNonSeiz(:,seizureInd) = [];
featureVectorNonSeiz = featureVectorNonSeiz';

% Create and train SVM model
% Randomly select 1000 samples from non-seizure data
randSampleInd = randi(size(featureVectorNonSeiz, 1), 1, 1000);
finalFeatureVector = [featureVectorSeiz;featureVectorNonSeiz(randSampleInd,:)];
finalTargetVector = [targetVectorSeiz;targetVectorNonSeiz(randSampleInd,:)];

SVMmodel = fitcsvm(finalFeatureVector, finalTargetVector, 'Standardize',...
    true,'KernelFunction','RBF','KernelScale','auto');
```

Read EEG Data for patient 01 (In blocks of 10 records)

```
function featMatrix = generateFeatMat(index)

parfor i = 1:10
    fileName = sprintf('chb03/chb03_%d.edf', i+index);
    [~,records(:,:,i)] = edfread(fileName);
end

% Channels being used for detection
chansOfInterest = [2 19 20 21];
waveletPkts = [2 2 2 2];

% Each channel is represented by a column for filtering
data = permute(records(chansOfInterest,:,:), [2, 1, 3]);

% Create low pass filter
```

```
Fs = 256;  % Sampling Frequency

N      = 50;      % Order
Fc     = 25;      % Cutoff Frequency
DpassU = 0.01;    % Upper Passband Ripple
DpassL = 0.01;    % Lower Passband Ripple
DstopU = 0.0001;  % Upper Stopband Attenuation
DstopL = 0.0001;  % Lower Stopband Attenuation

% Calculate the coefficients using the FIRCLS function.
b  = fircls(N, [0 Fc Fs/2]/(Fs/2), [1 0], [1+DpassU DstopU], [1-DpassL ...
    -DstopL]);
Hd = dfilt.dffir(b);

LPFiltered = filter(Hd, data);
records = permute(LPFiltered,[2 1 3]);

wdSize = 1024;
level = 5;
wName = 'db4';

% featMatrix = ones(numFeat*size(records, 1), size(records, 2)/wdSize,...
%     size(records, 1));      % Matrix to store the features
%                             % First dimension - WP Coeffs of each window
%                             % Last dimension - Record Number


% Extraction of features (Wavelet Packet Coefficients)
for i = 1:size(records, 3)
    for j = 1:size(records, 2) / wdSize
        for k = 1:size(records, 1)
            signal = records(k,1 + (wdSize * (j - 1)):wdSize * j, i);
            wpt = wpdec(signal,level,wName);
            wpCoef = wpcoef(wpt, [level, waveletPkts(k) - 1]);
            featMatrix(k, j, i) = log(sqrt(mean(wpCoef.*wpCoef)));
            % featMatrix((2*k), j, i) = var(wpCoef);
        end
    end
end
end
```

Published with MATLAB® R2016b

References

[1] Goldberger AL, Amaral LAN, Glass L, Hausdorff JM, Ivanov PCh, Mark RG, Mietus JE, Moody GB, Peng C-K, Stanley HE. PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals. Circulation 101(23):e215-e220 [Circulation Electronic Pages; http://circ.ahajournals.org/cgi/content/full/101/23/e215]; 2000 (June 13).

[2] Sheb, A., and John Guttag. "Application of machine learning to epileptic seizure detection." 2010 the 27th International Conference on Machine Learning, Haifa, Israel. 2010.

[3] Faust, Oliver, et al. "Wavelet-based EEG processing for computer-aided seizure detection and epilepsy diagnosis." Seizure 26 (2015): 56-64.