



# **Development of Electroencephalogram (EEG) Signals Classification Techniques**

A thesis submitted by

**Hadi Ratham Ghayab Al Ghayab**

For the award of

**Doctor of Philosophy**

**2019**



## **ABSTRACT**

Electroencephalography (EEG) is one of the most important signals recorded from humans. It can assist scientists and experts to understand the most complex part of the human body, the brain. Thus, analysing EEG signals is the most preponderant process to the problem of extracting significant information from brain dynamics. It plays a prominent role in brain studies. The EEG data are very important for diagnosing a variety of brain disorders, such as epilepsy, sleep problems, and also assisting disability patients to interact with their environment through brain computer interface (BCI). However, the EEG signals contain a huge amount of information about the brain's activities. But the analysis and classification of these kinds of signals is still restricted. In addition, the manual examination of these signals for diagnosing related diseases is time consuming and sometimes does not work accurately. Several studies have attempted to develop different analysis and classification techniques to categorise the EEG recordings.

The analysis of EEG recordings can lead to a better understanding of the cognitive process. It is used to extract the important features and reduce the dimensions of EEG data. In the classification process, machine learning algorithms are used to detect the particular class of EEG signal based on its extracted features. The performance of these algorithms, in which the class membership of the input signal is determined, can then be used to infer what event in the real-world process occurred to produce the input signal. The classification procedure has the potential to assist experts to diagnose the related brain disorders. To evaluate and diagnose neurological disorders properly, it is necessary to develop new automatic classification techniques. These techniques will help to classify different EEG signals and determine whether a person is in a good health or not. This project aims to develop new techniques to enhance the analysis and classification of different categories of EEG data.

A simple random sampling (SRS) and sequential feature selection (SFS) method was developed and named the SRS\_SFS method. In this method, firstly, a SRS technique was used to extract statistical features from the original EEG data in time domain. The extracted features were used as the input to a SFS algorithm for key

features selection. A least square support vector machine (LS\_SVM) method was then applied for EEG signals classification to evaluate the performance of the proposed approach.

Secondly, a novel approach that combines optimum allocation (OA) and spectral density estimation methods was proposed to analyse EEG signals and classify an epileptic seizure. In this study, the OA technique was introduced in two levels to determine representative sample points from the EEG recordings. To reduce the dimensions of sample points and extract representative features from each OA sample segment, two power spectral density estimation methods, periodogram and autoregressive, were used. At the end, three popular machine learning methods (support vector machine (SVM), quadratic discriminant analysis, and k-nearest neighbor (k-NN)) were employed to evaluate the performance of the suggested algorithm.

Additionally, a Tunable Q-factor wavelet transform (TQWT) based algorithm was developed for epileptic EEG feature extraction. The extracted features were forwarded to the bagging tree, k-NN, and SVM as classifiers to evaluate the performance of the proposed feature extraction technique. The proposed TQWT method was tested on two different EEG databases.

Finally, a new classification system was presented for epileptic seizures detection in EEGs blending frequency domain with information gain (InfoGain) technique. Fast Fourier transform (FFT) or discrete wavelet transform (DWT) were applied individually to analyse EEG recording signals into frequency bands for feature extraction. To select the most important feature, the infoGain technique was employed. A LS\_SVM classifier was used to evaluate the performance of this system.

The research indicates that the proposed techniques are very practical and effective for classifying epileptic EEG disorders and can assist to present the most important clinical information about patients with brain disorders.

## **CERTIFICATION OF DISSERTATION**

This Thesis is the work of Hadi Ratham Ghayab Al Ghayab except where otherwise acknowledged, with the majority of the authorship of the papers presented as a Thesis by Publication undertaken by the Student. The work is original and has not previously been submitted for any other award, except where acknowledged.

Principal Supervisor: Yan Li

Associate Supervisor: Shahab Abdulla

Associate Supervisor: Siuly Kabir

Student and supervisor's signatures of endorsement are held at the University.



## LIST OF CONTRIBUTIONS FROM PUBLICATION CO-AUTHORS

This section presents details of contributions by the various authors for each of the papers presented in this Thesis by Publication. The following detail is the agreed share of contribution for candidate and co-authors in the presented publications in this thesis:

Chapter 2, Al Ghayab et al., (2016)

**Al Ghayab, H.R.**, Li, Y., Abdulla, S., Diykh, M. and Wan, X., 2016. Classification of epileptic EEG signals based on simple random sampling and sequential feature selection. *Brain informatics*, 3(2), pp.85-91.

Author	Percent contribution	Tasks Performed
Al Ghayab, H.R	60%	Designed the method, simulation, analysis, interpretation, wrote entire draft of paper.
Li, Y, Abdulla, S., Diykh, M. and Wan, X.,	40%	Help with methodology design, suggested edits to manuscript, interpretation.

Chapter 3, Al Ghayab et al., (2018)

**Al Ghayab, H.R.**, Li, Y., Siuly, S. and Abdulla, S., 2018. Epileptic EEG Signal Classification using Optimum Allocation Based Power Spectral Density Estimation. *IET Signal Processing*, 12(6), pp.738-747.

Author	Percent contribution	Tasks Performed
Al Ghayab, H.R	65%	Designed the method, simulation, analysis, interpretation, wrote entire draft of paper.
Li, Y., Siuly, S. and Abdulla, S.,	35%	Help with methodology design, suggested edits to manuscript, interpretation.

Chapter 4, Al Ghayab et al., (2018)

**Al Ghayab, H.R.,** Li, Y., Siuly, S. and Abdulla, S., A Feature Extraction Technique based on Tunable Q-Factor WT for Brain Signal Classification, *Journal of Neuroscience Methods*, 312, pp.43-52.

Author	Percent contribution	Tasks Performed
Al Ghayab, H.R	70%	Developed the method, simulation, analysis, interpretation, wrote entire draft of paper.
Li, Y., Siuly, S. and Abdulla, S.,	30%	Suggested edits to manuscript, interpretation

Chapter 5, Al Ghayab et al., (2018)

**Al Ghayab, H.R.,** Li, Y., Siuly, S. and Abdulla, S., 2018. Epileptic seizures detection in EEGs blending frequency domain with information gain technique. *Soft Computing*, 23(1), pp.227-239.

Author	Percent contribution	Tasks Performed
Al Ghayab, H.R	70%	Developed the method, simulation performing, analysis, interpretation, wrote entire draft of paper,
Li, Y., Siuly, S. and Abdulla, S.,	30%	Suggested edits to manuscript, interpretation, data acquisition.



## **ACKNOWLEDGMENTS**

In the first, I humbly thanks Allah Almighty, who gave me health and thoughts to achieve this goal.

To my inspiration, my late parents Ratham and Fakhriya: I owe it all to you. Many thanks!

I would like to express my deepest gratitude to my supervisor, Professor Yan Li, for her exceptional guidance, support, caring, encouragement, advice, and individual feedback throughout my PhD journey. I feel very grateful and blessed to have worked under her supervision. I would also like to express my sincere gratitude to my associate supervisors Dr Siuly and Dr Shahab, for their continuous encouragement, support, valuable advice, and suggestions. I would like to thank Dr Barbara Harmes for her help and support. Also, I wish to thank Bill Bryon for making proofreading.

I would also like to thank my brothers and sisters. They have always supported me and encouraged me with their best wishes.

I wish to thank my affectionate and supportive wife, Sabreen, and my wonderful children, Rafal, Hassan and new-born Refka who provide unending inspiration.

I would like to acknowledge the support by Thi\_Qar University for give the opportunity to do my study.



## LIST OF RELATED PUBLICATIONS

- Al Ghayab, H.R.,** Li, Y., Siuly, S. and Abdulla, S., 2018. Epileptic EEG signal classification using Optimum Allocation Based Power Spectral Density Estimation. *IET Signal Processing* 12(6), pp.738-747.
- Al Ghayab, H.R.,** Li, Y., Siuly, S., & Abdulla, S., 2018. Epileptic seizures detection in EEGs blending frequency domain with information gain technique. *Soft Computing*, 23(1), pp.227-239.
- Al Ghayab, H.R.,** Li, Y., Siuly, S. and Abdulla, S., A Feature Extraction Technique based on Tunable Q-Factor WT for Brain Signal Classification, *Journal of Neuroscience Methods*, 312, pp.43-52.
- Al Ghayab, H.R.,** Li, Y., Abdulla, S., Diykh, M. and Wan, X., 2016. Classification of epileptic EEG signals based on simple random sampling and sequential feature selection. *Brain informatics*, 3(2), pp.85-91.
- Al Ghayab, H.R.,** Li, Y., Siuly, S., Abdulla, S. and Wen, P., 2017, October. Developing a Tunable Q-Factor Wavelet Transform Based Algorithm for Epileptic EEG Feature Extraction. In *International Conference on Health Information Science* (pp. 45-55). Springer, Cham, Moscow.



# TABLE OF CONTENTS

ABSTRACT.....	ii
LIST OF CONTRIBUTIONS FROM PUBLICATION CO-AUTHORS.....	vi
ACKNOWLEDGMENTS .....	viii
LIST OF RELATED PUBLICATIONS .....	x
TABLE OF CONTENTS .....	xii
LIST OF FIGURES .....	xvi
LIST OF TABLES .....	xvii
CHAPTER 1.....	19
1.1 Brain structures and their labor.....	20
• Cerebrum.....	20
• Cerebellum.....	21
• Brainstem.....	21
1.2 Brain and EEG recordings.....	21
1. Bipolar or sequential montage.....	25
2. Referential Montage .....	25
3. Average reference montage.....	26
4. Laplacian montage.....	26
1.3 The EEG wave patterns.....	27
1.4 Epileptic seizures.....	29
1.5 Overview techniques analysis and classification .....	30
1.5.1 Analysis of EEG signals.....	30
1.5.1.1 Analysis using time domain methods.....	31
1.5.1.2 Analysis using frequency domain methods .....	33
1.5.1.3 Analysis using time frequency domain methods.....	34
1.5.1.4 Analysis using traditional nonlinear methods.....	35
1.5.1.5 Analysis using graph theory approaches .....	35
1.6 Classification of EEG signals .....	36
1.7 Overview and Motivation.....	37
1.8 Research Problems.....	38
1.9 Contributions of the Thesis .....	39
1.10 Connections Between Chapters .....	41
1.11 Structure of the Thesis.....	42
CHAPTER 2.....	45
2.2 Chapter Summary.....	53

<b>CHAPTER 3.....</b>	<b>55</b>
<b>3.2 Chapter Summary.....</b>	<b>66</b>
<b>CHAPTER 4.....</b>	<b>69</b>
<b>4.2 Chapter Summary.....</b>	<b>80</b>
<b>CHAPTER 5.....</b>	<b>83</b>
<b>5.2 Chapter summary.....</b>	<b>97</b>
<b>CHAPTER 6.....</b>	<b>99</b>
<b>6.1 Summary and Conclusions of the Thesis.....</b>	<b>99</b>
<b>6.2 Future Work.....</b>	<b>103</b>
<b>REFERENCES.....</b>	<b>107</b>
<b>APPENDICES.....</b>	<b>115</b>
<b>Appendix A : Matlab simulation code for Simple Random Sampling and Sequential Feature Selection .....</b>	<b>117</b>
A.I Matlab code for simple random sampling (SRS) and sequential feature selection (SFS). .....	117
<b>Appendix B : Matlab simulation code for Optimum Allocation and Power Spectral Density Estimation .....</b>	<b>121</b>
B.I Matlab code for optimum allocation (OA) and power spectrum density estimation. ....	121
<b>Appendix C : Matlab simulation code for Tunable Q-Factor WT .....</b>	<b>135</b>
C.I Matlab code for Tunable Q-Factor WT (TQWT). .....	135
<b>Appendix D : Matlab simulation code for Frequency Domain with Information Gain Technique.....</b>	<b>173</b>
D.I Matlab code for Frequency Domain with Information Gain Technique. ....	173

## LIST OF FREQUETLY USED ABBREVIATIONS

WHO	World Health Organization
EEG	Electroencephalography
MEG	Magnetoencephalography
PET	Positron Emission Tomography
fMRI	Functional Magnetic Resonance Imaging
ST	Sampling Technique
SRS	Simple Random Sampling
SFS	Sequential Feature Selection
OA	Optimum Allocation
PSD	Power Spectral Density
PD	Periodogram
AR	Autoregressive
AP	all pairs
1vs1	One vs One
1vsA	One vs All
BC	Binary Complete
TC	Ternary Complete
OR	Ordinal
SR	Sparse Random
DR	Dense Random
SSC	Calculated Sample Size
SSCoC	Calculated Sample Size for each class
TQWT	Tunable Q-factor Wavelet Transform
WT	Wavelet Transform
FT	Fourier Transform
F	Focal
N	Non-Focal
Q	Q-Factor
R	Redundancy
SB	Sub-Bands
Min	Minimum
Max	Maximum
Mean	Mean
Med	Median
Q1	First Quartile
Q2	Second Quartile
Ran	Range
SD	Standard Deviation
Sk	Skewness
Var	Variance
ACA	Average of Classification Accuracy
FFT	Fast Fourier Transform
DWT	Discrete Wavelet Transform
InfoGain	Information Gain

LS_SVM	Least Square Support Vector Machines
SVM	Support Vector Machines
QSVM	Quadratic Kernel Support Vector Machine
QDA	Quadratic Discriminant Analysis
k-NN	k-Nearest Neighbour
BT	Bagging Tree



## LIST OF FIGURES

(Excluding publication included in Chapters 2-5)

Figure 1.1 Structure of the human brain (Standring, 2015). .....	20
Figure 1.2 Structure, link, and work of a neuron (Sanei & Chambers, 2013). .....	22
Figure 1.3 Location of electrodes on the scalp using the international 10-20 system (Abhang & Gawali, 2015; Klem et al., 1999). .....	24
Figure 1.4 Example of recording EEG signals from multi and single channel.....	25
Figure 1.5 Examples of five frequency bands of EEG rhythmic (Brain waves; Siuly, 2016). .....	28
Figure 1.6 Block diagram of the procedures for processing EEG signals. ....	31

## **LIST OF TABLES**

(Excluding publication included in Chapters 2-5)

Table 1.1 Examples of montage type for EEG recordings..... 26

Table 6.1 Total of the dimension reduction and the time consuming from each studies and the whole database. .... 102

# 1

## CHAPTER 1

### INTRODUCTION

The human brain is the most important organ in the human body as it controls all the body's activities. When a person is moving or doing specific activities, different brain cells are involved by sending electrical signals to the targeted part of the body to get the right response (Lindsay & Norman, 2013; Radocy & Boyle, 2012). To detect the neuron's responses of the brain, several methods are developed for instance: electroencephalography (EEG), magnetoencephalography (MEG), positron emission tomography (PET), functional magnetic resonance imaging (fMRI), and optical imaging. Although MEG, PET, fMRI and optical imaging are in favour of clinical experts, they are required a high cost to use them in diagnose of brain disorders (Wolpaw et al., 2002). In addition, PET, fMRI and optical imaging depend on reading several metrics from a patient such as blood pressure. Collecting and analysing these metrics could take long time and less accurate compared with EEG which records information quickly and efficiently, as well as being inexpensive (Wolpaw et al., 2002).

EEG record the brain activities as signals. These signals are obtained from electrodes placed on the scalp. Experts are then analysed and investigated them to decide whether a patient have neurologic disorders, such as sleep problems, and epilepsy. Many researchers have used EEG signals as a tool to discover brain activities using a variety of techniques (Faust et al., 2015; Salem et al., 2014; Zhang & Parhi, 2014; Zhu et al., 2014). These techniques are used to extract and select discriminative features, as well as to classify EEG recordings. Most of these techniques fall under

five different categories: time domain; frequency domain; time frequency domain; traditional nonlinear methods; and graph theory approaches (Acharya et al., 2013).

### 1.1 Brain structures and their labor

The human brain consists of three main parts: the cerebrum, brainstem and cerebellum. Each part of the brain is associated with different human activities. A brief explanation about these brain's parts is presented below:

- **Cerebrum**

The cerebrum is the largest part in the human brain. It is divided into two nearly equal halves that are the left and right hemispheres, by a deep longitudinal pothole (Davey, 2011). Their outer parts are the cerebral cortex, which are made up of grey matter coordinated in layers, while inside the cortex is the white matter (Kandel et al., 2000). Each side of the hemisphere is partitioned into four lobes: frontal, parietal, temporal and occipital lobes, and denoted according to the skull bones located below them, (Standring, 2015) as shown in Figure 1.1.

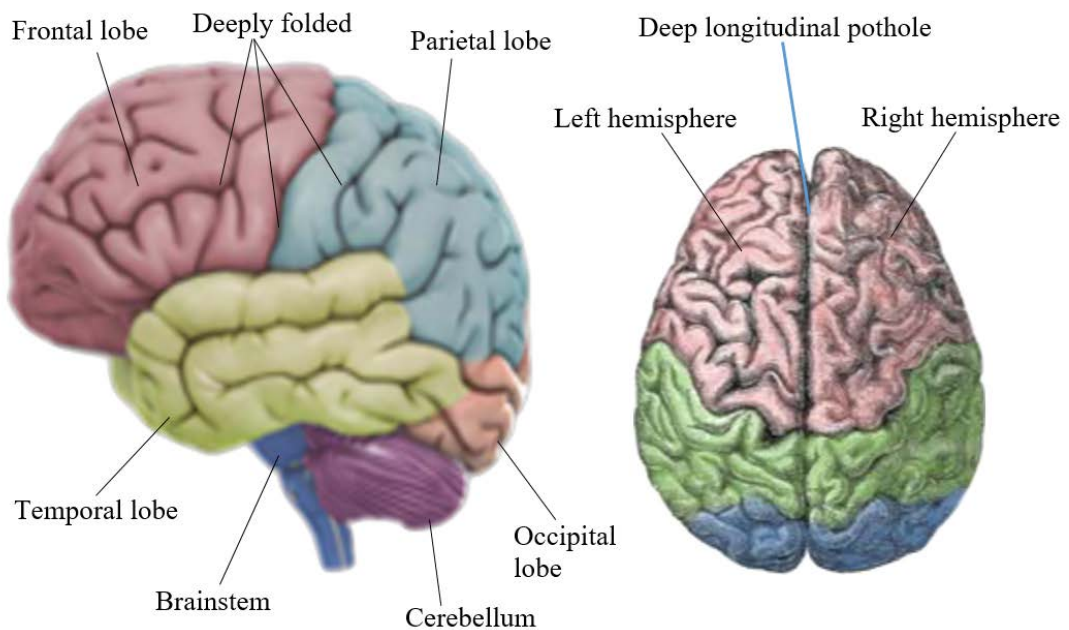


Figure 1.1 Structure of the human brain (Standring, 2015).

Each lobe is associated with certain bodily activities. However, some of these activities are with two or more lobes (Ackerman, 1992). The frontal lobe is associated with emotions, problem solving, planning, personality, speech and movement. The parietal lobe is associated with sensations such as pain and touch, orientation, recognition, movement and stimuli. However, processing of visualization is controlled by the occipital lobe while the temporal lobe is responsible for perception, differentiation of voices, memory, speech and stimuli.

- **Cerebellum**

The cerebellum is the second largest part of the brain and is located at the back of the cranial cavity lying underneath the occipital lobe. It is isolated from the occipital lobe by a sheet of fibre (Standring, 2015). The cerebellum is divided into three lobes; anterior; posterior; and flocculonodular. The anterior and posterior lobes are associated with the responses of motor movements, while the flocculonodular lobe is associated with maintaining the balance (Hall, 2015). In the cerebellum, the outer cortex is thinner when compared with the outer cortex of the cerebrum (Standring, 2015).

- **Brainstem**

The brainstem lies under the cerebrum and includes the midbrain, pons and medulla. It is located in the back of the skull, continuing downwards to the spinal cord (Standring, 2015). The brainstem has reticular formation repressing as a network of nuclei used to pass brain's commands to body organs. The brainstem maintains many essential processes such as breathing, control of eye movements and balance (Standring, 2015; Hall, 2015).

### **1.2 Brain and EEG recordings**

The brain contains billions of neurons that keep the brain active (Herculano-Houzel, 2009). These neurons, which have the same parts {soma (cell body), axon and dendrites} are interconnected by membranes which transport proteins and exchange ions with each other as illustrated in Figure 1.2. This process leads to many ions being pushed out of neurons and also repels the analogous ions. These ions can push their

neighbours, who ultimately push their neighbours etc., and in the end a wave is created (Sanei & Chambers, 2013).

This wave reaches the scalp and is recorded by using electrodes placed on the scalp. These electrodes reflect voltages of millions of neurons over time as electroencephalography (EEG) signals (Tatum, 2014). Sometimes, several neurons create abnormal waves due to do not have the same direction as other neurons, and do not pump their ions properly. These abnormal waves influence the normal EEG characteristics.

As characteristics of EEG signals are complex, aperiodic and nonlinear in nature. EEG signals are a combination of sustained oscillation and non-oscillation transients that exhibit by a variety of frequencies. To record and study brain activities, there are several tools used for this purpose, such as EEG, MEG, PET, fMRI, and optical imaging. EEG signals have a relatively poor spatial sensitivity. However, they have multiple advantages over other techniques. The EEG hardware costs are cheaper than those of most other techniques. In addition to this, EEG sensors can be used in different places compared with others. Furthermore, some techniques take a long time to collect data from the brain, while EEG only requires a short time (Wolpaw et al., 2002). That motivated us to use EEG signals in this study.

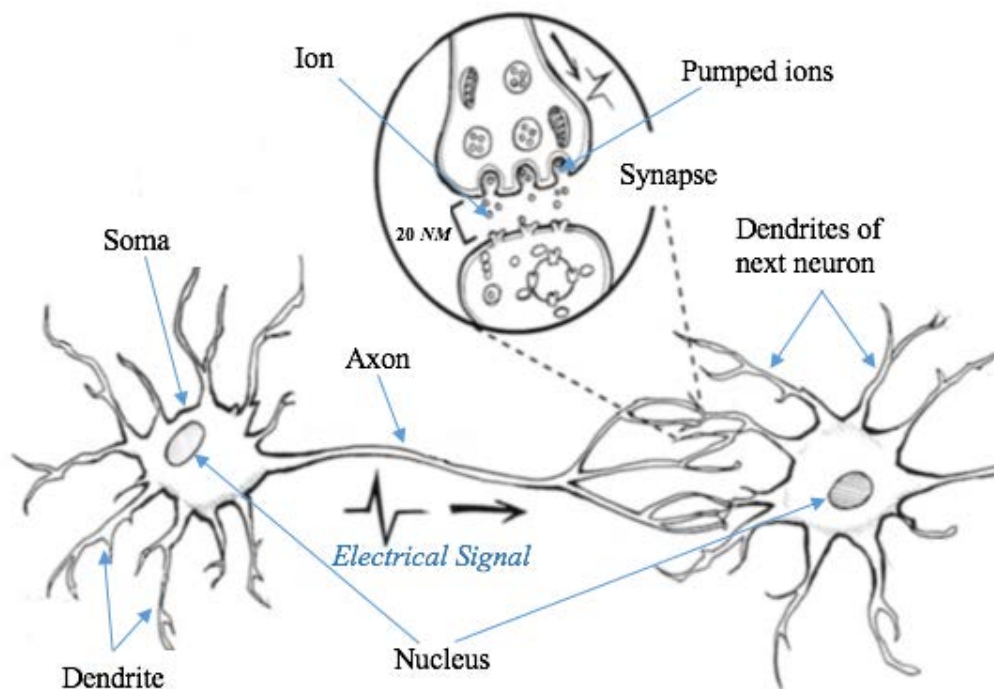


Figure 1.2 Structure, link, and work of a neuron (Sanei & Chambers, 2013).

EEG signals define as an electrophysiological monitoring method used to record electrical activity of the brain using electrodes placed on the scalp. In addition, EEGs measure the change of voltage in the swapped ions of the brain's neurons (Niedermeyer & Silva, 2005). As EEGs record the spontaneous electrical brain activity through time, EEGs have been used in several clinical research to trace abnormality in the brain's behaviours such as epileptic seizures.

In 1842, different studies tried to record the activity of the brain. Studies began when Richard Caton, who was a physician practising in Liverpool, recorded first brain activities. He proved in 1875 that the exposed cerebral hemispheres of animals released electrical signals (Swartz 1998). Thereafter, Adolf Beck presented his findings in 1890 about electrical activity, which included consistent oscillations in the brains of rabbits and dogs, by placing electrodes directly on the scalp (Coenen et al., 2014). Till (1924) reported the first brain signals (EEG) which were recorded by Hans Berger (Haas, 2003). Depending on the previous studies on animals, Hans Berger developed one of the most significant developments in human history, which was a device to record the activity of the human brain (Millet, 2002).

To record brain activities, we need to use a number of electrodes and set them in different places on the scalp with a conductive gel or paste and connect them to an amplifier that is attached to an EEG recording.

There are two types of recorded EEG signals, multi-channel and single-channel EEG recordings, depending on the number of electrodes used to capture these signals. For instance, using two electrodes leads to one single signal being recorded (single channel), while using more than two electrodes at the same time leads to multi-channel EEG signals being recorded. Multi channels can include up to 256 electrodes around the scalp using a cap or net of electrodes.

The International 10-20 system is used as a standard way to set the electrodes on the scalp (Towle, 1993; Klem et al., 1999). The locations of these electrodes are specified by dividing them into a 10% and 20% interval, depending on the distances among neighbouring electrodes (Abhang & Gawali, 2015). As shown in Figure 1.3, there are two points on the scalp help to determine the electrode positions, named as Nasion and Inion, where the nasion is located at the front of the head, alongside with

the eyes, and the inion is located in the back of the head. It can be seen in Figure 1.3 that each electrode has letters which correspond to the placement of the electrode. For example, the F refers to the front of the skull, T the temporal, C refers to the centre of the skull, P refers to parietal and O is the occipital. However, the letter "z" refers to the midline area from the nasion to inion point. In addition, from numbers associated to the electrodes, it can recognize their locations to determine whether they are placed in the left or right sides of the scalp. The electrodes with even numbers are placed to the right side of the skull while the electrodes with odd numbers are placed to the left side of the skull.

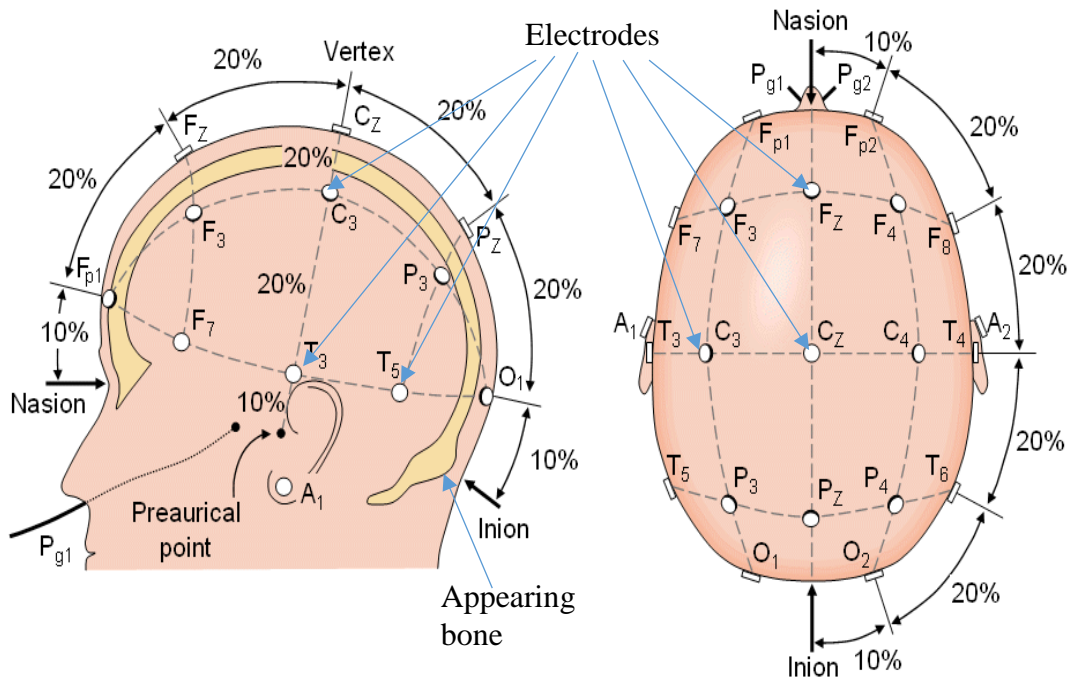


Figure 1.3 Location of electrodes on the scalp using the international 10-20 system (Abhang & Gawali, 2015; Klem et al., 1999).

Each electrode senses diverse voltages of each lobe of the cerebrum or other parts of the brain. To collect these voltages and transfer them to the computer as EEG signals, one amplifier per electrode pair is used to amplify the voltage for the active electrode and its reference is shown in Figure 1.4. As EEG signals symbolize different voltages for two electrodes, there are numerous ways of reading EEG signals known as “montages” as illustrated below:



## 1. Bipolar or sequential montage

Using this method, two neighbouring electrodes begin to register one channel of EEG (Nunez & Pilgreen, 1991; Niedermeyer & Silva, 2005) and the whole montage consists of a sequential number of channels. For instance, the two electrodes Fp1 and F3 produce a channel 'Fp1-F3', and the next electrodes, F3 and C3 generate the channel 'F3-C3', and so on, as shown in Figure 1.4.

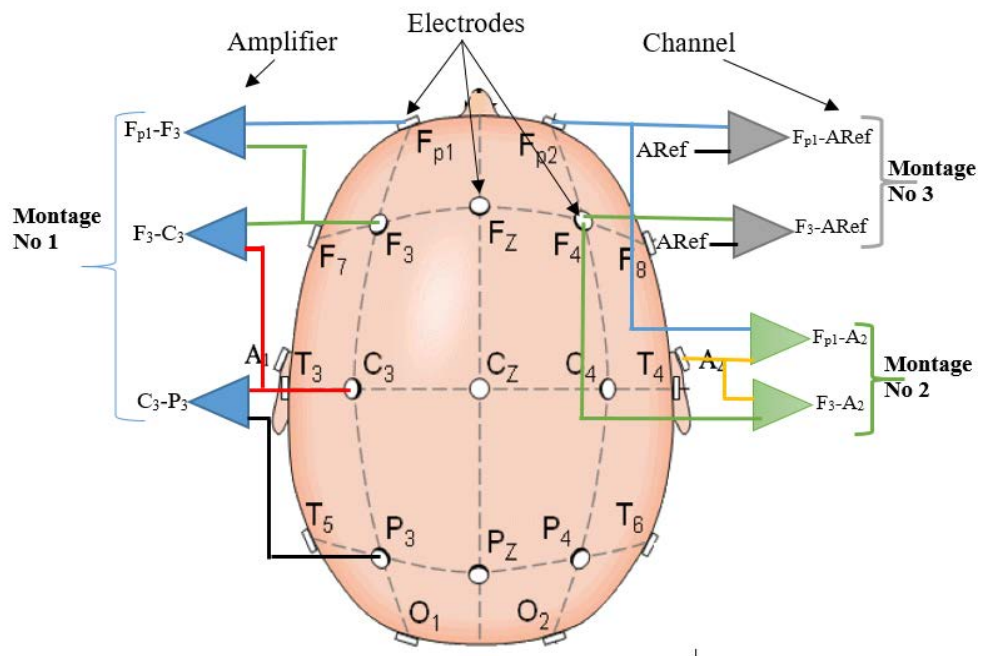


Figure 1.4 Example of recording EEG signals from multi and single channel.

## 2. Referential Montage

In this montage, there is no standard way of using the electrodes to record EEG signals. However, each channel represents the difference between a certain electrode and a particular reference electrode. Midline positions are often used because they do not amplify the signal in one hemisphere compared with other positions. Another method of referential is to take a physical or mathematical average of electrodes set to the earlobes or mastoids, which use electrodes A2, A1 as references (Siuly et al., 2016; Nunez & Pilgreen, 1991) as shown in Figure 1.4.

### 3. Average reference montage

This montage uses an average signal (ARef) of all the amplifiers as a designated reference for each channel (Siuly et al., 2016).

### 4. Laplacian montage

Each channel represents the difference between electrodes and a weighted average of the surrounding electrodes (Nunez & Pilgreen, 1991).

Table 1.1 Examples of montage type for EEG recordings

Bipolar sequential montage	or Referential Montage	Average reference montage
F <sub>p1</sub> -F <sub>3</sub>	F <sub>p1</sub> -A <sub>1</sub>	F <sub>p1</sub> -ARef
F <sub>3</sub> -C <sub>3</sub>	F <sub>3</sub> -A <sub>1</sub>	F <sub>3</sub> - ARef
C <sub>3</sub> -P <sub>3</sub>	C <sub>3</sub> -A <sub>1</sub>	C <sub>3</sub> - ARef
P <sub>3</sub> -O <sub>1</sub>	P <sub>3</sub> -A <sub>1</sub>	P <sub>3</sub> - ARef
	F <sub>7</sub> -A <sub>1</sub>	F <sub>7</sub> - ARef
F <sub>p1</sub> -F <sub>7</sub>	T <sub>3</sub> -A <sub>1</sub>	T <sub>3</sub> - ARef
F <sub>7</sub> -T <sub>3</sub>	T <sub>5</sub> -A <sub>1</sub>	T <sub>5</sub> - ARef
T <sub>3</sub> -T <sub>5</sub>	O <sub>1</sub> -A <sub>1</sub>	O <sub>1</sub> - ARef
T <sub>5</sub> -O <sub>1</sub>		
F <sub>p2</sub> -F <sub>4</sub>	F <sub>p2</sub> -A <sub>2</sub>	F <sub>p2</sub> - ARef
F <sub>4</sub> -C <sub>4</sub>	F <sub>4</sub> -A <sub>2</sub>	F <sub>4</sub> - ARef
C <sub>4</sub> -P <sub>4</sub>	C <sub>4</sub> -A <sub>2</sub>	C <sub>4</sub> - ARef
P <sub>4</sub> -O <sub>2</sub>	P <sub>4</sub> -A <sub>2</sub>	P <sub>4</sub> - ARef
	F <sub>8</sub> -A <sub>2</sub>	F <sub>8</sub> - ARef
F <sub>p2</sub> -F <sub>8</sub>	T <sub>4</sub> -A <sub>2</sub>	T <sub>4</sub> - ARef
F <sub>8</sub> -T <sub>4</sub>	T <sub>6</sub> -A <sub>2</sub>	T <sub>6</sub> - ARef
T <sub>4</sub> -T <sub>6</sub>	O <sub>2</sub> -A <sub>2</sub>	O <sub>2</sub> - ARef
T <sub>6</sub> -O <sub>2</sub>		

Table 1.1 presents in detail the type of montages that are used to collect a number of EEG channels. Through these montages, two types of EEG signals are identified analog; and digital. In analog EEG, the EEG specialist changes electrodes during the different montages to record and highlight the significant features of EEGs. In

addition, all the EEG signals are recorded digitally, and are stored and categorized within each montage, which leads to particularly easy usage.

### 1.3 The EEG wave patterns

An EEG recording contains an array of signals depending on the number of electrodes used to record the brain's electrical activities (brain waves). These waves are defined as the electric rhythmic changes. With these rhythms, the EEG is divided into five frequency bands: Delta, Theta, Alpha, Beta and Gamma. These bands are used to measure brain waves which fall into the range of 0.5-32 Hz frequency (Teplan, 2002) as shown in Figure 1.5. The following subsections clarify the frequency bands in detail:

- Delta falls in the range 0.5-4 Hz frequency. It demonstrates the highest amplitude and the slowest waves. It appears in EEG signals when humans are asleep and not dreaming or in deep sleep;
- Theta lies between 4 Hz to 7 Hz with  $20\mu\text{v}$ . It can be seen in EEGs recorded from teenagers and older adults (Cahn & Polich, 2013) and can also be observed in long distance drivers. Thus, the increase of theta frequency in the brain performs abnormal activity. It refers to a focal disorder in focal subcortical lesions, deep midline disorders, metabolic encephalopathy or other disorders, such as diffuse disorders. In other words, theta is associated with records of the relaxed, meditative and creative states;
- Alpha occurs between 7 Hz to 13 Hz with an amplitude range from 30 to 50  $\mu\text{v}$  (Gerrard & Malcolm, 2007). Alpha waves are appeared in an adult who is awake with eyes closed, relaxed and has cleared mind of any wandering thoughts. It is recorded from the occipital and parietal regions in the back of the skull;

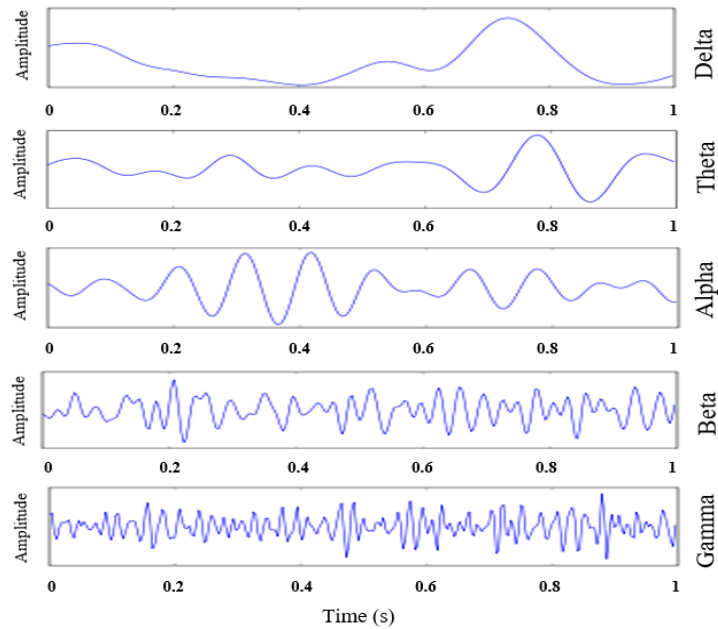


Figure 1.5 Examples of five frequency bands of EEG rhythmic (Brain waves; Siuly, 2016).

- Beta is in the frequency range of 14 Hz to 30 Hz with low amplitude. It is observed on both sides in parallel locations in the frontal area of the skull. Beta waves are closely related to body movements, thinking and active concentration (Pfurtscheller & Da Silva, 1999). In addition, it is linked with several brain disorders, such as drug effects and Duo 15q syndrome, which is a neuro disorder, caused by duplicating part of Chromosome 15 that accords a risk for epilepsy, intellectual disability, and autism (Frohlich et al., 2016);
- Gamma lies between the range of 30 Hz and 100 Hz with the lowest amplitude compared with other frequency bands. It refers to a number of neurons working together as a network for the purposes of a certain cognitive or motor function (Niedermeyer & Silva, 2005).

In nature, EEGs recorded from adults are different to EEGs from children, where the latter have slow oscillations compared with adult's EEGs. In addition, EEG signals are impacted by non-cerebral origin signals named as artifacts. These artifacts have a large amplitude compared to the amplitudes of important cortical signals. This is one of the reasons why it takes considerable experience to correctly interpret EEGs clinically.

EEG signals are most often utilized to differentiate between normal and abnormal signals. In this thesis, different techniques are developed to analyse and classify epileptic EEG signals. The following section provides more details about the epileptic seizures and how these neurological disorders affect EEG signals.

### **1.4 Epileptic seizures**

An epileptic seizure is an event or symptom leading to abnormal signals in the EEG recordings (Fisher et al., 2005) and affects a patient's daily life by a momentary loss of awareness or absence seizure. Seizures are not detected to epileptic patients alone, but also they can be seen in people who do not have epilepsy for a variety of reasons, such as drug use, low blood sugar, low levels of oxygen, brain trauma and high body temperature. A first episode of epilepsy does not usually require immediate anti-seizure medication, unless there is a determined a patient's problem with EEG signals (MEG, PET and so on) (Wilden & Cohen-Gadol, 2012). Often older people, who live up to 80 years, have at least one epileptic seizure (Berg, 2008; Wilden & Chohen-Gadol, 2012).

Epilepsy influences about 50 million people internationally each year (WHO, 2018). Epilepsy is defined as having two or more seizures. There are episodes of involuntary movements that have an effect on parts or all of the body, and their symptoms vary depending on the type of seizure (convulsive or non-convulsive) (Shearer et al., 2006). The convulsive type is the most widespread around the world, which leads to loss of awareness of patients and also loss of control of bladder function. Further, the convulsive seizure type is ordered by whether the source of the seizure is localized (focal seizure) or distributed (generalized seizure) within the brain (National, 2012). Although the brain activity is normally not concurrent, an epileptic seizure impacts the brain activity by making abnormal and excessive signals which are sent from a group of neurons in a synchronized way that performs in a wave of depolarization named as a paroxysmal depolarizing shift (National, 2012; McPhee & Hammer., 1995).

Using EEG recordings assists to understand more about epilepsy. Epilepsy is a neurological disorder that impacts on the nature of the brain activities and causes abnormal EEG signals. In order to diagnose and treat these kinds of disease, analysis and classification of the EEG data is required to detect epileptic seizures. In EEG

recordings, there are two abnormal categories which can be observed, ictal (can appear through epileptic seizures) and inter-ictal (can be seen through free epileptic seizures or between two seizures). These seizures lead to unexpected changes in frequency specifications and oscillations for a normal EEG signal, which is often within the alpha range with slow amplitude that may or may not be observed in the form of EEG recordings. Sometimes, this change of EEG charts occurs between beta and gamma bands with an increase in amplitude. In order to reveal whether people are healthy or have epilepsy disease, this thesis aims to develop techniques which can be used to analyse and classify the EEG signals.

### **1.5 Overview techniques analysis and classification**

EEG signals contain a huge amount of information about the brain's functions and human activities. Consequently, the analysis and classification of these signals play significant roles in biomedical and big data research. Analysing EEG recordings is vitally important to extract discriminative features and reduce the large amount of EEG data. However, classifying these features is no less paramount than its analysis, which work together for the diagnosis of brain related disorders. Further, a good classification approach supports the decision making on people's health by differentiating EEG categories.

The next sections explain in detail the EEG signal analysis and classification techniques used in these areas. For analysing and classifying EEGs, there are several procedures most researchers follow, as shown in Figure 1.6.

#### **1.5.1 Analysis of EEG signals**

Raw EEG data contain artifacts generated through different channels (e.g. physical problems in the electrodes or other parts of the recording machine, connectivity issues between head and device or head motion during the EEG recordings etc). This stage depends on two approaches, which are artifact removal and data filtering (Ames, 1971; Başar & Schürmann, 1999). The artifact removal is considered a challenging stage because artifacts create an abnormal EEG signal shape with high amplitudes. It is necessary to remove this unwanted noise and provide a clear EEG signal.

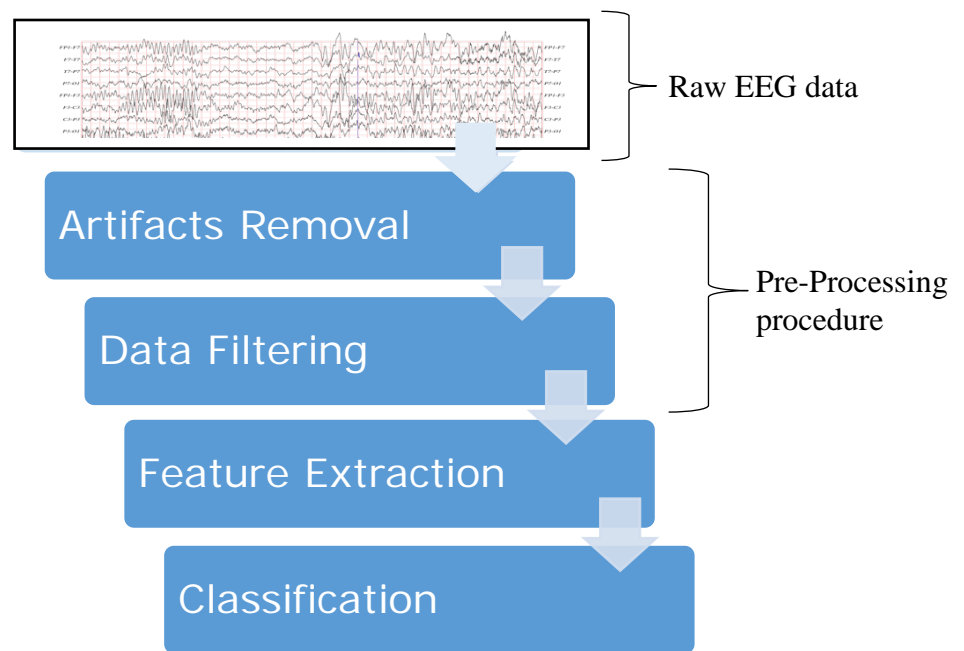


Figure 1.6 Block diagram of the procedures for processing EEG signals.

The next stage of EEG analysis is feature extraction and selection, which is used to derive and choose representative features from EEG data by applying various signal processing techniques to it (Kumar & Bhuvaneshwari, 2012). This stage is very important to extract representative features from EEG data and reduce the amount of unnecessary signals. In this stage, the extraction method impacts enormously on the results of the EEG classification. Additionally, these are excellent methods to use to extract relevant features and accurately describe EEG signals in order to obtain high quality EEG classifications.

From the literature, it is shown that a variety of methods have been used for feature extraction for epileptic EEG data. Most of those techniques fall under five different categories: time domain; frequency domain; time frequency domain; traditional nonlinear methods; and graph theory approaches (Acharya et al., 2013). The following sections provide details of the signal analysis techniques used and brief descriptions of the previous research undertaken in this regard.

### **1.5.1.1 Analysis using time domain methods**

In time domain, the processing shows how the EEG signal changes with time. The important aspects of time series analysis of EEG recordings are linear prediction and component analysis approach, using for example principal component analysis, independent component analysis, and linear discriminant analysis (Acharya et al., 2013). These methods are used to analyse EEG recordings with regard to a specified period of time.

In the last few decades, a sampling technique (ST) based on least square support vector machines (LS\_SVM) was proposed by Siuly et al. (2009). Firstly, they used the ST to extract features from two different classes of people such as healthy individuals with eyes open and epileptic patients during seizure activity. They divided the features into two datasets: training and testing datasets. They applied the LS\_SVM to evaluate the extracted features. Chen et al. (2017) proposed a method to detect epileptic seizures from the EEG data by using the feature extraction method, which is combining the maximum cross correlation and principal component analysis depending on LS-SVM classifier.

On the other hand, a genetic algorithm was used by Guo et al. (2011) to automatically extract features from EEG data in order to enhance the classifier's performance, as well as to reduce the feature's dimensionality. They used two groups of epileptic datasets. The first group included: two classes of healthy people and epileptic patients. The second group included: three classes of healthy people, interictal and ictal. The k-nearest neighbour classifier was used in this work to classify the two groups. They gained 88.6% and 99.2% accuracy for the first group without the genetic algorithm and with the genetic algorithm, respectively.

Husain and Rao (2014) presented an artificial neural network model using a back-propagation algorithm for classification of epileptic EEG signals. They decomposed the EEG signals into a finite set of band limited signals termed 'intrinsic mode' functions. They also applied Hilbert transform method on those intrinsic mode functions to calculate instantaneous frequencies. An optimum allocation based principal component analysis method was proposed by Siuly and Li (2015) to extract key features for the classification of multiclass EEG signals from epileptic EEG data. They used four different classifiers: LS\_SVM; naive bayes classifier; k-nearest neighbour algorithm; and linear discriminant analysis to determine which one was the most effective. They utilized four different output coding approaches for the multi-class



LS\_SVM. These were error correcting output codes; minimum output codes; one vs one (1vs1); and one vs all.

To reduce the dimension of extracted features from discrete wavelet transforms, Subasi and Gursoy (2010) utilized three-time domain methods: principal components analysis; independent components analysis; and linear discriminant analysis.

### **1.5.1.2 Analysis using frequency domain methods**

In frequency domain, the processing demonstrates that the number of the EEG signal falls within frequency bands through a number of frequencies. The most significant analysis methods in frequency domain are spectral estimation methods, which can be classified as parametric and non-parametric methods (Kiyimik et al., 2004). These methods are used to analyse EEG signals with regard to frequency and its bands.

In 2013, Shen et al. proposed a cascade of wavelet-approximate entropies for feature selection. They utilised fisher scores for adaptive feature selection, and support vector machine for feature classification to detect epileptic seizures. They applied this method to different epileptic EEG recordings: open source EEG data; and clinical EEG data. The method obtained overall classification accuracies of 99.97% and 98.73%, respectively.

Chua et al. (2011) gained features from raw EEG recordings by using higher order spectra. In order to select the best classifier, they forwarded extracted features into two different methods to compare their performances. They used a gaussian mixture model and a support vector machine classifier to detect epileptic EEG signals. They achieved average accuracies of 93.11% and 92.56% for higher order spectra based on the gaussian mixture model classifier and the support vector machine classifier, respectively, for the following different EEG classes: normal; pre-ictal; and epileptic.

A wavelet transform method was used by Gajic et al. (2014) to extract key features. They also used scatter matrices to reduce the dimensionality of the features. These features were used as the input to a quadratic classifier. The EEG epileptic database was classified into: healthy subjects; epileptic subjects during a seizure-free period (inter-ictal); and epileptic patients during seizure activity (ictal). They obtained a 99% classification accuracy.

The discrete wavelet transform (DWT) was employed by Amin et al. (2015) for analysing EEG recordings. Two datasets of the raven's advance progressive metric test and the eyes open test were utilized. This research applied different classifiers, including the SVM, K-nearest neighbouring, and multi-layer perceptron classifiers. An accuracy of above 98% was achieved by using these classifiers with approximation (A4) and detail coefficients (D4) from the DWT.

### **1.5.1.3 Analysis using time frequency domain methods**

Time frequency methods are applied to analyse EEG signals in both the time and frequency domains at the same time. These methods are often used to analyse two dimensional or multidimensional signals (Sejdic & Jiang, 2009). The important analysis methods in time frequency domain are wavelet transforms, such as continuous wavelet transform, discrete wavelet transform, and wavelet packet decomposition.

Recently, a sequential floating forward selection algorithm was proposed to detect epileptic seizures in EEG signals by Choi et al. (2012). They selected the best energy as the key features from frequency bands by using the sequential floating forward selection algorithm. Moreover, feature extraction was also carried out through an empirical mode decomposition. The extracted features were used to feed two classifiers, the classification and regression tree and the C4.5 classifiers. The method proposed by Martis et al. (2012) obtained an accuracy of 95.33% by using the C4.5 classifier. Bhattacharyya et al. (2018) presented an empirical wavelet transform to analyse the EEG signals and to reveal an area associated to the focal epilepsy.

In 2015, Gajic et al., extracted different features from time, frequency, time frequency domain, and used the non-linear analysis. Those features were obtained from sub-bands with good representative characteristics compared with those from the whole band. The researchers reduced the dimension of the features by using scatter matrices.

There are several types of wavelet transform approaches, which include wavelet transform; discrete wavelet transform; wavelet packet decomposition; and continuous wavelet transform, that were applied to determine the epileptic seizures from EEG recordings (Sharma et al., 2017; Chen et al., 2017; Alickovic et al., 2018; Abibullaev et al., 2010).

#### **1.5.1.4 Analysis using traditional nonlinear methods**

Naturally, the EEG recording is a nonlinear signal, which can be processed in an efficient way by utilizing nonlinear methods. There are a variety of useful nonlinear techniques used to analyse the EEG signals, such as Hurst exponent, Q-factor wavelet transform, correlation dimension, entropies and so on.

In recent times, higher order spectra have been applied by Chua et al. (2007; 2011) to identify epileptic seizures from the EEG signals based on SVM and the Gaussian mixture model. They achieved 86.89% and 88.78% average classification accuracies from SVM and the Gaussian mixture model, respectively. To differentiate between normal and epileptic EEG signals, Acharya et al. (2011) applied recurrence plots and extracted recurrence quantification analysis. The extracted features were forwarded to various classifier models. They reported that the extracted features based on the SVM and fuzzy classifiers yielded a 94.4% average efficiency.

In another study, Martis et al. (2015) applied several nonlinear methods: Hurst exponent; Higuchi fractal dimension; largest Lyapunov exponent; and sample entropy to extract the most representative features. To evaluate extracted features, they used the SVM classifier with diverse kernel functions and other classifiers. Their methods based on the SVM with a radial basis function obtained a 98% classification accuracy.

Acharya et al. (2012) proposed an automatic detection of normal, pre-ictal, and ictal conditions by extracting four entropy features, namely: approximate entropy; sample entropy; phase entropy 1; and phase entropy 2 from the EEG recordings and the extracted features fed seven classifiers. The optimum result they achieved was a 98.1% accuracy based on Fuzzy classifier.

#### **1.5.1.5 Analysis using graph theory approaches**

Graph theory (complex networks) techniques are utilized to analyse EEG data depending on the number of EEG channels. These are a relatively new technique dealing with multichannel EEGs by mapping each channel as a node in the graph methods. Also, studying the graph theory based on single channel EEG signals is an important factor, because it applies each feature of EEG as a node in complex networks. The most significant analysis techniques on complex networks are visibility

graphs, horizontal visibility graphs, and recurrence networks (Zhu et al., 2014; Diykh & Li, 2016; Diykh et al., 2017).

Zhu et al. (2014) suggested a weighted horizontal visibility graph to detect epileptic seizures in EEG signals. Their approach was applied to extract two representative features, which were mean degree and strength. They obtained 100% accuracy for some of the database. In order to detect epileptic seizures in EEG recordings, Diykh et al. (2017) provided an undirected weighted complex network. They employed four popular classifiers to evaluate the provided method. Their methodology gained a 98% accuracy rate.

A measurement of statistical mechanics of complex networks from horizontal visibility graphs was provided by Artameeyanant et al. (2017). To evaluate the performance of the proposed method, three well known classifiers were utilized. In the suggested method, the authors achieved an average of 98.83% classification accuracy. A new edge weight technique for visibility graphs in complex networks has been provided by Supriya et al. (2016) to extract the most powerful features from the EEG data. In order to evaluate their proposal, they applied SVM and linear discriminant analysis as classifiers.

Wang et al. (2017) applied three graph theories, namely visibility graphs, horizontal visibility graphs and difference visibility graphs in a complex network to analyse and classify epileptic EEG databases. The authors extracted new features, such as degree distributions, degree entropies and power law degree powers from different visibility graphs. They reported that the degree distributions can differentiate between epileptic seizures and non-seizures.

Many approaches for the classification of epileptic EEG signals have been developed. This chapter provided some of those techniques. It also reported that some researchers obtained 100% accuracy by using different methods. However, those methods required a long time to gain results or used only a part of the database in their studies.

### **1.6 Classification of EEG signals**

Classification is a processing method to group a number of features in different categories depending on particular characteristics. It is the most popular way of

learning, for instance, allocating materials to solid, liquid and gas or detecting a disease based on patient examination and symptoms. In this stage, the input data are classified based on certain categories that are determined by particular features called vectors. The features in vectors work as a description of instances or classes. In addition, the classification aims to allocate a number of labels to the extracted features by observing a set of information for a definite issue. Methods that are applied to classify the extracted features are called classifiers.

The classification methods are divided into three types based on the usage of features vector labels, such as labelled classification (called supervised classification), semi-labelled classification (denoted as semi-supervised classification) and unlabelled classification (named as unsupervised classification). For instance, the action of the supervised classification requires all input data to be labelled. In the semi-supervised classification approach, only a few pieces of input data are labelled, whilst the rest of them are not. However, unsupervised classification methods require no labelling at all (Huang et al., 2014).

In this thesis, four techniques have been employed: simple random sampling with sequential feature selection; optimum allocation based power spectral density estimation; tunable Q-factor wavelet transform; and frequency domain with information gain. Also, the five following classification methods: least square support vector machine (LS-SVM); quadratic discriminant analysis (QDA); k-nearest neighbour (k-NN); support vector machine (SVM); and bagging tree (BT) are used as tools to classify the extracted features.

### **1.7 Overview and Motivation**

This thesis focuses on developing novel methods for analysing epileptic EEG data. EEG signals are obtained using a number of electrodes connected to a computer. Recordings epileptic EEG normally take between 20 to 30 minutes (Niedermeyer & Silva, 2005). These recorded EEG signals are normally used to distinguish between epileptic seizures and other forms of fainting, such as sub-cortical movement disorder and psychogenic non-epileptic seizures.

Epilepsy is a conditional brain's disorder that impairs patient's daily live (Kumar et al., 2015; McGrogan, 1999). It is characterized by recurrent and sudden incidence of

epileptic seizures (Buck et al., 1997). According to a report released by World Health Organization (WHO) more than 50 million of the world's population are affected by epilepsy (WHO, 2017).

Developing new effective techniques to detect the effected brain regions with seizures is a crucial step in pre-surgical assessment. EEG signals have been proved to be a powerful tool for detecting and diagnosing epilepsy diseases. The main challenge of biomedical signal processing research is how to analyse and classify EEG recordings accurately to obtain a high level of accuracy. Much research regarding the classification of epileptic EEG signals have been designed (Acharya et al., 2012; Altunay et al., 2010; Chen et al., 2010; Joshi et al., 2014; Sharma et al., 2014; Sharma & Pachori, 2015; Arunkumar et al., 2016; Bajaj et al., 2017; Tzimourta et al., 2018; Usman & Hassan, 2018) and most of those research work have limitations. It is necessary to develop new techniques to cope these limitations to achieve accurate classification results. Thus, a goal of this thesis is to develop EEG classification methods to analyse and classify EEG recording signals.

### **1.8 Research Problems**

EEGs contain a huge amount of information that exhibit brain activities. Some of the recorded information could be irrelevant data that represent as a noise and artifact. However, in order to reduce unrelated data, an automatic technique is required to extract representative features and classify these features by suitable methods. Previously, the analyses and classification approaches of those signals have not been well developed. Those procedures are usually evaluated by experts who visually examine the EEG records (Kutlu et al., 2009; Subasi & Ercelebi, 2005). However, it is often difficult to recognize people who have brain diseases through visual inspection of EEG recordings (Siuly & Li, 2015; Siuly & Zhang, 2016). Also, the visual inspection of these signals takes a long time and error prone. This inspection method is therefore unsuitable to produce credible information. Improved analysis and classification of EEG signals automatically will lead to better diagnostic techniques for brain related diseases.

Recently, many approaches have been developed to extract the key features from EEG data and classify these features into different EEG categories. From previous

studies (Kiymik et al., 2004; Lawhern et al., 2012; Zhu et al., 2013; Siuly & Li, 2014; Acharya et al., 2015; Zhang et al., 2016; Diykh & Li, 2016; Satapathy et al., 2017), it seems that most of the techniques provided had limitations such as did not get high accuracy rates and took more time to perform the required analysis, as well as too complicated for practical applications. In addition, some of these methods tested with small datasets rather than a huge dataset that made them unsuitable to utilize in real-time applications.

To address these limitations, this thesis proposes four techniques for epileptic EEG data classification. In addition to classifying the EEG signals effectively, representative features need to be extracted and selected carefully, and time consumption must be reduced. Hence, the following research questions are identified:

1. How to enhance the performance of EEG signals classification?

This question can be divided into three sub-questions as seen below:

- a. How to analyse different EEG categories in effective ways?
- b. How to reduce the execution time required to classify and analyse EEG signals?
- c. How to reduce the feature space of the EEG recordings?

### **1.9 Contributions of the Thesis**

Naturally, EEG signals are aperiodic, complex, nonlinear and nonstationary (Selesnick, 2011; Siuly et al., 2017). From these characteristics, it is challenging to extract representative features from big EEG data sets for classification. This study focuses on the analysis of the EEG signals, extracting the most representative features, reducing irrelevant data, more efficient use of time, and investigating the most suitable classification methods, using various machine learning classification methods. The developed methods in this thesis were published in various journals, about the analysis and classification of the EEG signals. Those research work were reviewed and summarised in terms of their precision of classification. The outcomes of this thesis are:

1. Simple random sampling (SRS) combined with a sequential feature selection (SFS) technique was presented to classify epileptic EEG data. Firstly, a SRS technique

was applied to extract the most representative features from the raw EEG data. The SRS was used to divide the EEG recording into two stages. In stage one, ten samples were chosen randomly from each set of epileptic EEG signals. In the next stage, five subsamples were selected randomly from each sample of the first stage. However, a sample size of each stage was determined through using a sample size calculator. At the end of the SRS, nine statistical features were extracted from each subsample. Secondly, the SFS algorithm was used to reduce the dimensionality of the EEG data by selecting the discriminative features. Finally, a least square support vector machine (LS\_SVM) classifier was applied to classify the extracted features. The SRS\_SFS method was executed to classify the epileptic EEG signals. The results show that this method has the potential to classify the EEG data. Details of this approach are given in Chapter 2. The content of this chapter was published by the journal of *Brain Informatics*.

2. Optimum allocation (OA) based power spectral density estimation was developed to detect epileptic seizures from EEG recordings as can be seen in Chapter 3. The OA technique was introduced in two levels to determine representative sample points from the EEG recordings. In the first level, the sample size of each EEG class was defined by applying a sample size calculator. In the second level, the OA technique was applied to segment each EEG class and then choose the sample points based on sample size determination. Afterwards, two power spectral density estimation methods (periodogram and autoregressive) were used to reduce the dimensions of sample points and extract representative features from each OA sample segment. Three popular machine learning methods: support vector machine (SVM), quadratic discriminant analysis, and k-nearest neighbor (k-NN) were employed in order to evaluate the performance of the proposed method. The experimental results show that the proposed scheme with SVM classifier achieves high accuracy compared with recent studies. As a result, this method has the most potential to help experts to analyse and classify quantities of EEG data. The content of this chapter was published by the journal of *IET Signal Processing*.
3. A tunable Q-factor wavelet transform based algorithm was developed for epileptic EEG features extraction in order to classify multichannel EEG signals. This framework depends on two phases to analyse large amounts of EEG recordings. In the first phase, the EEG signals were decomposed into a number of sub-bands by using TQWT method. In the second phase, each sub-band was partitioned into n



windows and then ten statistical features were extracted from each window. Finally, the extracted features were forwarded to the bagging tree, using k-NN, and SVM as classifiers to evaluate the performance of the proposed feature extraction technique. The experimental results demonstrated that the proposed feature extraction algorithm with the k-NN classifier produces the best performance, when compared to the other two classifiers. The details were given in Chapter 4, which was published by the journal of *Neuroscience Methods*.

4. Finally, a frequency domain implemented with Information Gain (InfoGain) method to assist with the extraction of the most important features and detect epileptic seizures in EEG signals, was proposed. In this study, fast Fourier transform (FFT) and discrete wavelet transform (DWT) were applied individually to analyse EEG recording signals into frequency bands. A segmentation model was used to segment each band into k windows and extract a number of statistical features from these windows. Next, to select the most important feature, the InfoGain technique was employed. Finally, in order to evaluate the performance of this system, a LS\_SVM classifier was used. The results showed that FFT combined with InfoGain yields the best performance when the LS-SVM classifier was utilized. The details of this technique are provided in Chapter 5, the contents which were published by the journal of *Soft Computing*.

### **1.10 Connections Between Chapters**

This thesis focuses on analysing epileptic EEG signals to extract and select the most representative features and also to reduce the dimensions of the EEG recordings in different domains. For analysis and data extraction, this study used a SRS with SFS to classify the EEG data in time domain (Chapter 2). Furthermore, it introduced a novel classification method entitled optimum allocation-based power spectral density estimation to classify the EEG recordings in time frequency domain (Chapter 3). A Tunable Q-factor wavelet transform based algorithm is developed for feature extraction in nonlinear method (Chapter 4). A system that extracts and selects the most relevant data and ignores irrelevant data, whilst detecting simultaneously epileptic seizures by blending frequency domain with the InfoGain approach, is provided in frequency domain (Chapter 5).

### 1.11 Structure of the Thesis

This thesis consists of six chapters and each chapter provides important information on our study. The rest of the thesis is structured as follows:

**Chapter 2** presents a new SRS combined with SFS algorithm for the classification of epileptic EEG signals. A least square support vector machine (LS\_SVM) used in this study as a classifier method.

**Chapter 3** integrates the optimum allocation technique and the power spectral density estimation for EEG classification. This chapter also investigates a suitable machine learning classifier by using three popular classification methods: SVM; quadratic discriminant analysis; and  $k$ -NN. This chapter provides a comparative study between the proposed technique and other existing methods in terms of accuracy.

**Chapter 4** introduces a Tunable Q-factor wavelet transform for the differentiation of EEG categories. This algorithm decomposes the EEG signals into a number of sub-bands for feature extraction, and bagging tree,  $k$ -NN, and SVM are used as classifiers to evaluate the proposed algorithm.

**Chapter 5** provides a frequency domain blending with information gain that helps to detect epileptic seizures. In this chapter, two different frequency methods, FFT and DWT, are applied separately to decompose the EEG signals into frequency bands. In order to select the most important features from these bands, we used the information gain method. This chapter also utilizes the LS\_SVM classifier to evaluate the proposed scheme.

**Chapter 6** presents a summary and the findings of this study. Possible directions for future work are also discussed in this chapter.

**Appendices A-D** provide simulation code for our proposed methods, described in Chapters 2, 3, 4 and 5.

This page is intentionally left blank



# 2

## CHAPTER 2

### **CLASSIFICATION OF EPILEPTIC EEG SIGNALS BASED ON SIMPLE RANDOM SAMPLING AND SEQUENTIAL FEATURE SELECTION**

This chapter is an exact copy of a published article in Brain Informatics (2016). It describes a new method which extracts and selects features from multi-channel EEG signals. This research focuses on three main points. Firstly, simple random sampling (SRS) technique is used to extract features from the time domain of EEG signals. Secondly, the sequential feature selection (SFS) algorithm is applied to select the key features and to reduce the dimensionality of the data. Finally, the selected features are forwarded to a least square support vector machine (LS\_SVM) classifier to classify EEG signals. The LS\_SVM classifier classified the features that are extracted and selected from the SRS and the SFS.

In addition, appendix A provides MATLAB code for the proposed technique.

# Classification of epileptic EEG signals based on simple random sampling and sequential feature selection

Hadi Ratham Al Ghayab · Yan Li · Shahab Abdulla ·  
Mohammed Diykh · Xiangkui Wan

Received: 4 December 2015 / Accepted: 3 February 2016 / Published online: 27 February 2016  
© The Author(s) 2016. This article is published with open access at Springerlink.com

**Abstract** Electroencephalogram (EEG) signals are used broadly in the medical fields. The main applications of EEG signals are the diagnosis and treatment of diseases such as epilepsy, Alzheimer, sleep problems and so on. This paper presents a new method which extracts and selects features from multi-channel EEG signals. This research focuses on three main points. Firstly, simple random sampling (SRS) technique is used to extract features from the time domain of EEG signals. Secondly, the sequential feature selection (SFS) algorithm is applied to select the key features and to reduce the dimensionality of the data. Finally, the selected features are forwarded to a least square support vector machine (LS\_SVM) classifier to classify the EEG signals. The LS\_SVM classifier classified the features which are extracted and selected from the SRS and the SFS. The experimental results show that the method achieves 99.90, 99.80 and 100 % for classification accuracy, sensitivity and specificity, respectively.

**Keywords** Electroencephalogram · Epileptic seizures · Simple random sampling · Sequential feature selection · Least square support vector machine

## 1 Introduction

Epilepsy is a disorder which affects the human brain and hugely impairs patients' daily lives. It is characterized by recurrent and sudden incidence of epileptic seizures [1]. According to an estimation of the World Health Organization, more than 50 million of population are affected by epilepsy [2, 3]. Approximately, almost 1 % population have the neurological disorders [4–6]. It leads to numerous research works to identify epilepsy and related treatments. Electroencephalogram (EEG) signals have been proved as a powerful tool for detecting and diagnosing different neurological diseases. EEG signals are often used to detect and classify epilepsy [7]. It is often difficult for the experts to recognize the people who have a brain disorder through visual inspection of EEG signals [8]. In addition, visual inspection for discriminating EEG signals is a time consuming, error prone, costly process and not sufficient enough for reliable information. The analysis and classification of EEG signals can lead to better diagnostic techniques for brain-related disorders. It is thus important to develop better EEG classification methods.

Many researchers developed new techniques to extract the significant information from EEG signals. The information is used as the input to different classifiers. There are many approaches used to extract the key features as well as to further select features. Most of these fall under five broad categories: time domain, frequency domain, time–frequency domain, traditional non-linear methods and graph theory approaches [9].

---

H. R. A. Ghayab (✉) · Y. Li · S. Abdulla · M. Diykh  
Faculty of Health, Engineering and Sciences, University of  
Southern Queensland, Toowoomba, QLD 4350, Australia  
e-mail: HadiRathamGhayab.AlGhayab@usq.edu.au

Y. Li  
e-mail: Yan.Li@usq.edu.au

S. Abdulla  
e-mail: Shahab.Abdulla@usq.edu.au

M. Diykh  
e-mail: Mohammed.Diykh@usq.edu.au

X. Wan  
School of Electrical and Electronic Engineering, Hubei  
University of Technology, Wuhan 430068, China  
e-mail: wanxiangkui@163.com

One of the methods used in this paper for extracting epileptic EEG data is sample random sampling (SRS) technique. Researchers often applied the SRS in time domain. In this technique, each sample of the population has the same chance to be selected as a subject. The complete process of sampling is done in a single step, with each subject can be selected independently from the other samples of the population [10]. Then, we forwarded all these samples to the sequential feature selection (SFS) method for selecting the best features.

This study uses the selected features as the input for a classifier. One of the most popular classifiers, the least square support vector machines (LS\_SVMs) [11], is used to classify EEG data. This technique is used to identify the EEG data from healthy people and epileptic patients for epileptic seizures.

A lot of approaches for EEG signals classification have been developed [12]. There were reported a diverse of classification precisions for epileptic EEG data. Brief discussions of the previous research are provided below.

Gajic et al. [13] extracted different features from time, frequency, time–frequency domain and non-linear analysis.

These features were obtained from sub-bands with good representative characteristics. The researchers reduced the dimension of the features by using scatter matrices. This method yielded 98.7 % accuracy.

An optimum allocation-based principal component analysis method was proposed by Siuly and Li [8] to extract key features for the classification of multi-class EEG signals from epileptic EEG data. They used four different classifiers which were LS\_SVM, naive Bayes classifier,  $k$ -nearest neighbour (KNN) algorithm and linear discriminant analysis, to find out which one was the best classifier. They used four different output coding approaches for the multi-class LS\_SVM. These were error correcting output codes, minimum output codes, one versus one (1vs1) and one versus all. That method achieved a 100 % accuracy with LS\_SVM\_1vs1.

Feature extraction was carried out through an empirical mode decomposition. The extracted features were forwarded to two classifiers, the classification and regression tree and the C4.5 classifiers. The method using the C4.5 classifier suggested by Martis et al. [14] obtained good experimental results of 95.33, 98 and 97 % for accuracy, sensitivity and specificity, respectively.

Chua et al. [15] gained features from raw EEG recordings by using higher order spectra. They used a Gaussian mixture model (GMM) and a SVM classifiers to detect epileptic EEG signals. They achieved average accuracies of 93.11 and 92.56 % for the HOS based GMM classifier and the SVM classifier, respectively, for different EEG classes, such as normal, pre-ictal and epileptic EEGs.

On the other hand, a genetic algorithm (GA) was used by Guo et al. [16] to automatically extract features from EEG

data in order to enhance the classifier's performance, as well as, to reduce the feature's dimensionality. They used two groups of epileptic datasets. The first group was two classes of healthy people and epileptic patients. The second group was three classes of healthy people, inter-ictal and ictal. The KNN classifier was used in the work to classify the two groups. They gained 88.6 and 99.2 % accuracies for the first group without GA and with GA, respectively. They obtained of a 67.2 % accuracy without GA, and 93.5 % within GA, respectively, for the second group.

Ocak decomposed EEG signals, which were recorded from normal subjects and epileptic patients, by using discrete wavelet transform [17]. An approximate entropy (ApEn) was extracted from the approximation and the detail coefficients. The methodology achieved more than 96 % accuracy.

Srinivasan et al. used the ApEn to extract features and an artificial neural network classifier to identify epileptic EEG signals [18]. That approach achieved a high overall accuracy of 100 %.

Srinivasan et al. also proposed a special type of recurrent neural network, Elman network [19]. They used the feature extracted in time domain and frequency domain as the input to the proposed classifier. The Elman network method yielded a 99.6 % accuracy with a single input feature.

A wavelet transform method was used by Gajic et al. [20] to extract the key features. They also used scatter matrices to reduce the dimensionality of the features. These features were used as the input to a quadratic classifier. The EEG epileptic database was classified into healthy subjects, epileptic subjects during a seizure-free (inter-ictal) and epileptic patients during the seizure activity (ictal). They obtained a 99 % classification accuracy.

Shen et al. [12] proposed a cascade of wavelet-ApEn for feature selection. They used Fisher scores for adaptive feature selection, and SVM for feature classification to detect epileptic seizures. They applied the method to different epileptic EEG recordings: open source EEG data and clinical EEG data. The method obtained the overall classification accuracies of 99.97 and 98.73 %, respectively.

A sampling technique (ST) based on a LS\_SVM was proposed by Siuly et al. [21]. Firstly, they used the ST to extract features from two classes of, normal persons with eyes open and epileptic patients during a seizure activity. They applied the LS\_SVM to the extracted features. The total classification accuracy by that approach for both the training and testing datasets was 80.31 and 80.05 %, respectively.

Husain and Rao [22] presented an artificial neural network model using back propagation algorithm for the classification of epileptic EEG signals. They decomposed the EEG signals into a finite set of band limited signals termed as intrinsic mode functions. They also applied

Hilbert transform on these intrinsic mode functions to calculate instantaneous frequencies. They achieved a 99.80 % overall classification accuracy.

Rückstieß et al. [23] performed a SFS method to select the most representative features at each time step. Each successive features depended on the previous features. All the features were put into one vector and were forwarded to a classifier. This approach was applied for handwritten digits classification and a medical diabetes prediction task.

A sequential floating forward selection (SFFS) algorithm was proposed to detect epileptic seizures in EEG signals by Choi et al. [24]. They selected the most energy power as the features from frequency bands by using the SFFS algorithm. The total accuracy obtained by that method was 97.2 %.

In this study, we developed a new method combining the SRS with the SFS to acquire the best features set, and then we use the features as the input of the LS\_SVM classifier for the EEG classification. All the techniques are discussed in Sects. 3 and 4. The conclusion is presented in Sect. 5.

## 2 Experimental data

The data used in this study are open source EEG recordings and are publicly available<sup>1</sup> [25]. The database includes five sets of EEG recordings (sets **A–E**), with each containing 100 single-channel EEG signals of 23.6 s from five separate classes. References [13, 26] presented all details of these datasets from set **A** to **E**. This study selected set **A** which was taken from surface EEG recordings of five healthy people with eye open, and set **E** which was taken from EEG records of five pre-surgical epileptic patients during epileptic seizure activity.

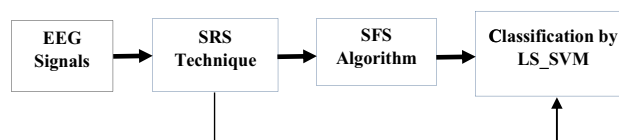
## 3 Methodology

The big EEG datasets cause the curse of dimensionality and make it difficult to estimate the accuracy of classification from a limited number of samples. This study develops a new structure for classifying epileptic EEG signals, as presented in Fig. 1. This work investigates and explores whether the SRS combined with SFS give the best features for epileptic EEG signals classification.

### 3.1 Simple random sampling (SRS) technique

SRS technique is a popular type of random or prospect sampling [21]. In this technique, each sample of the population has the same chance of being selected as a subject.

<sup>1</sup> <http://www.meb.unibonn.de/epileptologie/science/physik/eegdata.html>.



**Fig. 1** The structure of the proposed system

We put the number of population in a sample size calculator of the “Creative Research System” (available in sample size calculator online), to determine the sample size for both samples and subsamples. In this work, the dataset used are set **A** and set **E** (repeated). Each set has 100 data files, and each file has 4097 observations.

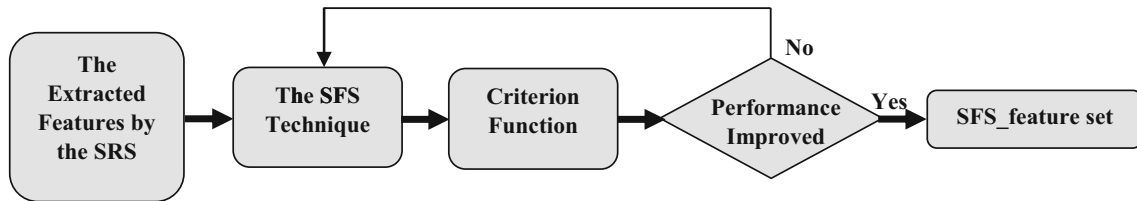
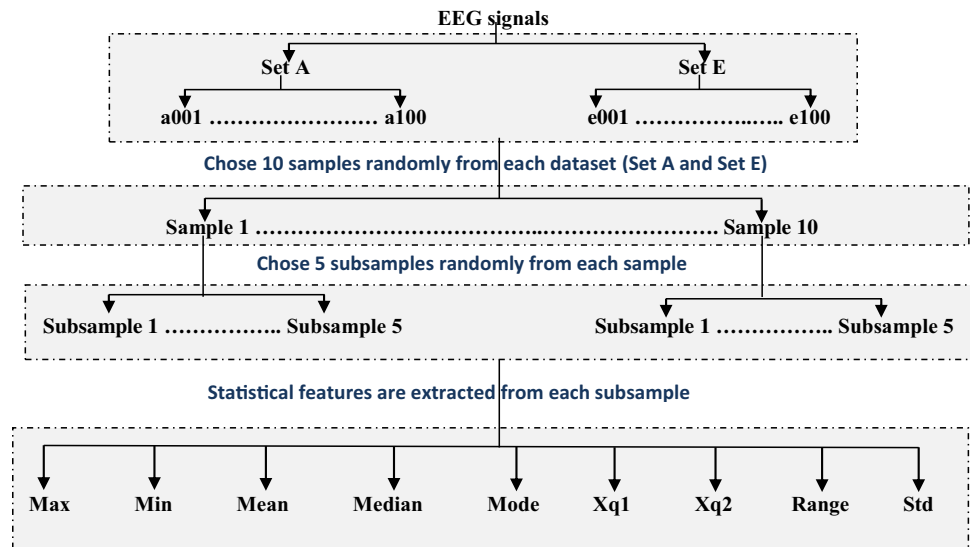
This research uses the sample size calculator to find the sample size needed as well as to find the subsample size. The sizes of the samples and the subsamples in this work are 3288 and 2746, respectively. The sizes were selected because they reflect the limitation of time to select samples and subsamples. Firstly, we randomly select 10 samples from size 3288 for each dataset (set **A** or **E**). Secondly, 5 subsamples are also random chosen from each 10 random samples, with a size of 2746. In each step, this study takes into account a 99–100 % confidence interval and a 99 % confidence level. In the last step, nine statistical features are extracted from each subsample. These features are {maximum value (Max), minimum value (Min), mean value, median value, mode, first quartile (Xq1), second quartile (Xq2), range value and standard deviation (Std)}. Figure 2 shows how samples, subsamples and features are taken from each class. We used MATLAB software package version 8.4, R2014b, for the experiments.

### 3.2 Sequential feature selection (SFS) algorithms

The SFS is used to reduce the dimensionality of the dataset selected randomly from the SRS. This method is used to generate fewer numbers of uncorrelated variables which are utilized as the features for the better classification of EEG signals. The aim of the presented sequential selection algorithm is to decrease the feature space,  $\mathbf{D} = x_1, x_2, \dots, x_n$ , to a subset of features,  $\mathbf{D} - n$ . It aims at enhancing or optimizing the computational execution of the classifier, as well as avoiding the curse of dimensionality [27]. This method is used to select a sufficiently reduced subset from the feature space  $\mathbf{D}$  without affecting the performance of the classifier. In order to choose a suitable feature subset size  $k$ , namely, a criterion function typically estimates the recognition rate of the classifier [28]. The SFS algorithm starts with an empty set  $\mathbf{S}$ , and progressively fills the set  $\mathbf{S}$  through adding features selected by the criterion function [29, 30]. It is searching on the feature space from bottom to up. Figure 3 illustrates how the SFS picks features from the original data. The SFS is applied to select the best features



**Fig. 2** The SRS technique to select samples, subsamples and statistical features



**Fig. 3** Features selection from the extracted features by the SRS

from the statistical features. The criterion is empirically chosen based on the experimental results. In this study, several experiments are made to define the best criterion. The criterion value is calculated based on the statistical relations among the features. Firstly, the Max value is chosen as the criterion as shown in Eq. (1).

$$\delta = \rho \sum_{i=1}^n fs2(i) \quad i = 1, 2, \dots, n, \tag{1}$$

where  $\delta$  refers to the criterion,  $\rho$  is one of the nine statistical features,  $n$  is the number of the features and  $fs2$  is the statistical feature set. Secondly, all the features are selected in the same way for Min, Mean, Mode and Std, in order to find the best features by the SFS algorithm. The best features (denoted as SFS\_feature) are selected based on Eqs. (2) and (3) as below:

$$\delta \leq fs2, \tag{2}$$

$$\delta > fs2. \tag{3}$$

3.3 The feature set

After decreasing the dimensions of the features through the SFS, the new feature set is forwarded to the LS\_SVM

classifier. In this study, we obtain a feature set that has 2000 data points of 35 dimensions. These features are divided into two groups, which are the training set and the testing set. The training set is directed to train a classifier. The testing set is employed to evaluate the performance of the methodology and it is utilized as the input of the classifier.

3.4 Least square support vector machines

In this subsection, we briefly review some basic work on LS\_SVMs for classification. LS\_SVMs are proposed by Suykens and Vandewalle. LS\_SVMs are the least square versions of SVMs, which are a set of related supervised learning methods that analyse data and recognize patterns. Moreover, they are used for classification and regression analysis [31]. In this research, the LS\_SVM classifier with a radial basis function kernel is used for the classification of epileptic EEG signals. These classifiers can avoid the problem of convex quadratic programming from the classical SVMs by using a set of linear equations [8]. In this paper, the classification is performed by LS\_SVMlab (version 1.8) toolbox in MATLAB<sup>2</sup> [32].

<sup>2</sup> <http://www.esat.kuleuven.ac.be/sista/lssvmlab/>.

### 3.5 Performance measures

This subsection presents assessing how the proposed method performs. The assessments include accuracy (also known as recognition rate), sensitivity (or recall) and specificity. The accuracy of a classifier is the percentage of the test set which is correctly classified by the classifier. The sensitivity is referred to the true positive rate which is the proportion of the positive set correctly identified.

The specificity is the true negative average which is the proportion of the negative set correctly identified. The following Eqs. (4)–(6) provide the definitions for the terms [33]:

$$\text{Accuracy} = \frac{TP + TN}{P + N}, \tag{4}$$

$$\text{Sensitivity} = \frac{TP}{P}, \tag{5}$$

$$\text{Specificity} = \frac{TN}{N}, \tag{6}$$

where TP is the number of true positives, TN is the number of true negatives and P and N are the positive and negative samples, respectively.

## 4 Results and discussions

In this study, we involved two datasets: sets **A** and **E** as mentioned in Sect. 2. SRS technique was used to extract features from the datasets. This technique selected features randomly by choosing 10 samples from each dataset (sets **A** and **E**). A five subsamples were selected from each sample. From each subsample, nine statistical features,

such as minimum, maximum, mean, median, mode, first quartile, third quartile, inter-quartile range and Std were extracted as aforementioned in Sect. 3.1.

A set of features obtained from the SRS included  $2000 \times 45$  dimensions. These features were used in two different ways. Firstly, the statistical features were directly fed to the LS\_SVM classifier and yielded the results, as shown in Table 1. Secondly, the SFS based on the criterion was employed to select the key features from the extracted features as mentioned in Sect. 3.2. As shown in the results, the good results of the best features are presented in Table 2. In Table 2, the good results are obtained by using the SRS algorithm and the SFS technique with the LS\_SVM classifier depending on the best criterion chosen. Furthermore, the LS\_SVM has two important parameters, which are  $\gamma$  and  $\sigma^2$  which should be suitably selected for achieving a desirable performance too. The LS\_SVM was affected by the value of these two parameters. This study trained the LS\_SVM with different groups of the parameters  $\gamma$  and  $\sigma^2$  to obtain best results. In this proposed method, we conducted with one group of the five EEG datasets and gained the best classification result with sets **A** and **E** when  $\gamma = 10$  and  $\sigma^2 = 1$  for the two methods applied in this paper. The results of the proposed method were compared with the results that were obtained from the SRS method and the LS\_SVM classifier. The experimental results showed that our approach yielded 99.90 % classification accuracy for the epileptic EEG data. Table 3 gives a better view for the results by the two different classification methods. On the other hand, in this study, the evaluation of time complexity between the presented approach and the SRS was conducted.

The SRS\_SFS\_LS\_SVM method took 0.16 s to classify the extracted features in Sect. 3.2. While the SRS\_LS\_SVM tackled the same features with 1.52 s as shown in Table 3. The performance of the proposed method is also compared with two existing methods in the literature. For fair comparison, the same dataset was used in comparison. The results show that the proposed method outperforms over the other two existing methods: a Huang–Hilbert transform and an artificial neural network model by Husain and Rao [22] and a ST and LS\_SVM methods by

**Table 1** Classification accuracy for epileptic EEG signals (sets **A** and **E**)

Statistical parameters	Results (%)
Accuracy	100
Sensitivity	100
Specificity	100

**Table 2** Experimental results using different statistic features as the criterion

Choose criterion	Accuracy (%)	Sensitivity (%)	Specificity (%)
Mean $\geq$ fs2 (SFS_feature)	99.90	99.80	100.00
Mean $\leq$ fs2 (SFS_feature)	98.90	98.00	99.80
Max $\leq$ fs2 (SFS_feature)	97.20	100.00	94.40
Min $\geq$ fs2 (SFS_feature)	99.10	99.20	99.00
Mode $\geq$ fs2 (SFS_feature)	97.70	95.40	100.00
Median $\leq$ fs2 (SFS_feature)	95.30	92.80	97.80
Std $\geq$ fs2 (SFS_feature)	95.60	91.20	100.00

**Table 3** Comparison of the results and time complexity for the proposed method with other methods

Methods	Accuracy (%)	Sensitivity (%)	Specificity (%)	Time (s)
SRS_LS_SVM	100.00	100.00	100.00	1.52
The proposed method with the best criterion (SRS_SFS_LS_SVM)	99.90	99.80	100.00	0.16

**Table 4** Comparison of performance of our proposed method with two recently reported methods for sets **A** and **E** of the EEG epileptic database

Different methods	Accuracy (%)	Sensitivity (%)	Specificity (%)
The proposed method with the best criterion (SRS_SFS_LS_SVM)	99.90	99.80	100
A Huang–Hilbert transform and an artificial neural network model [22]	99.80	99.75	100
A sampling technique and LS_SVM method [21]	80.05	74.97	87.70

Siuly et al. [21]. The performance comparison of the proposed method with the two reported methods to classify sets **A** and **E** is shown in Table 4. Husain and Rao in 2014 applied a Huang–Hilbert transform and an artificial neural network model on sets **A** and **E** (the same datasets used in this paper). They achieved a 99.80 % classification accuracy. While Siuly et al. in 2009 obtained 80.05 % classification accuracy when they used a ST and the LS\_SVM methods to classify the EEG signals for the same datasets. Moreover, the proposed method gains a 99.90 % classification accuracy for the same group of datasets. The results shown that the proposed technique in this paper has the potential to classify the EEG signals from healthy people and epileptic patients using the extracted and selected features from the SRS and SFS techniques.

## 5 Conclusions

This research concentrates on two classes of EEG signals from healthy people and epileptic patients. The study presents a SRS\_SFS method to extract and select the key features for classifying EEG signals into two classes. The LS\_SVM classifier is used to classify two-category EEG data after the feature extraction and selection. This method yields the results of 99.90, 99.80 and 100 % for classification accuracy, sensitivity and specificity, respectively. In addition, the proposed method is faster than the SRS technique. It means that the SRS\_SFS is useful for extracting and selecting the EEG features. To sum up, the proposed method is very efficient for analysing and classifying epileptic EEG signals. It will be also useful for the classification of other biomedical data.

**Open Access** This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use,

distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

## References

- Buck D, Baker GA, Jacoby A et al (1997) Patients' experiences of injury as a result of epilepsy. *Epilepsia* 38(4):439–444
- World Health Organization (WHO) (2011). Report: WHO. <http://www.who.int/mediacentre/factsheets/fs999/en/index.html>. Accessed Dec 2015
- Mcgrogan N (1999) Neural network detection of epileptic seizures in the electroencephalogram. <http://www.new.ox.ac.uk/~nmcgrogan/work/transfer>
- Boer H, Engel J, Prilipko L (2005) Global campaign against epilepsy. *Epilepsy Atlas* 82–83
- Iasemidis LD (2003) Epileptic seizure prediction and control. *IEEE Trans Biomed Eng* 50(5):549–558
- Kumar TS, Kanhangad V, Pachori RB (2015) Classification of seizure and seizure-free EEG signals using local binary patterns. *Biomed Signal Process Control* 15:33–40
- Adeli H, Zhou Z, Dadmehr N (2003) Analysis of EEG records in an epileptic patient using wavelet transform. *J Neurosci Methods* 123(1):69–87
- Siuly S, Li Y (2015) Designing a robust feature extraction method based on optimum allocation and principal component analysis for epileptic EEG signal classification. *Comput Methods Programs Biomed* 119(1):29–42
- Acharya UR, Sree SV, Swapna G et al (2013) Automated EEG analysis of epilepsy: a review. *Knowl Based Syst* 45:147–165
- Barreiro PL, Albandoz JP (2001) Population and sample. Sampling techniques. Management mathematics for European schools, MaMaEusch (994342-CP-1-2001-1-DECOMENIUS-C21)
- Wu F, Zhao Y (2005) Least squares support vector machine on Moret wavelet kernel function. In: International conference on neural networks and brain, 2005. ICNN&B'05. IEEE, Beijing, p 327–333
- Shen C-P, Chen C-C, Hsieh S-L et al (2013) High-performance seizure detection system using a wavelet-approximate entropy-fSVM cascade with clinical validation. *Clin EEG Neurosci* 44:247–256

13. Gajic D, Djurovic Z, Gligorijevic J et al (2015) Detection of epileptiform activity in EEG signals based on time–frequency and non-linear analysis. *Front Comput Neurosci*. doi:10.3389/fncom.2015.00038
14. Martis RJ, Acharya UR, Tan JH et al (2012) Application of empirical mode decomposition (EMD) for automated detection of epilepsy using EEG signals. *Int J Neural Syst* 22(6):1250027
15. Chua KC, Chandran V, Acharya UR et al (2011) Application of higher order spectra to identify epileptic EEG. *J Med Syst* 35(6):1563–1571
16. Guo L, Rivero D, Dorado J et al (2011) Automatic feature extraction using genetic programming: an application to epileptic EEG classification. *Expert Syst Appl* 38(8):10425–10436
17. Ocak H (2009) Automatic detection of epileptic seizures in EEG using discrete wavelet transform and approximate entropy. *Expert Syst Appl* 36(2):2027–2036
18. Srinivasan V, Eswaran C, Sriraam N (2007) Approximate entropy-based epileptic EEG detection using artificial neural networks. *IEEE Trans Inf Technol Biomed* 11(3):288–295
19. Srinivasan V, Eswaran C, Sriraam N (2005) Artificial neural network based epileptic detection using time-domain and frequency-domain features. *J Med Syst* 29(6):647–660
20. Gajic D, Djurovic Z, Di Gennaro S et al (2014) Classification of EEG signals for detection of epileptic seizures based on wavelets and statistical pattern recognition. *Biomed Eng Appl Basis Commun* 26:1450021
21. Siuly S, Li Y, Wen P (2009) Classification of EEG signals using sampling techniques and least square support vector machines. In: *Rough sets and knowledge technology*. Springer, Berlin, pp 375–382
22. Husain SJ, Rao K (2014) An artificial neural network model for classification of epileptic seizures using Huang–Hilbert transform. *Int J Soft Comput* 5(3):23
23. Rückstieß T, Osendorfer C, Van Der Smagt P (2011) Sequential feature selection for classification. In: *AI 2011: advances in artificial intelligence*. Springer, Berlin, pp 132–141
24. Choi K-S, Zeng Y, Qin J (2012) Using sequential floating forward selection algorithm to detect epileptic seizure in EEG signals. In: *2012 IEEE 11th international conference on signal processing (ICSP)*. IEEE, Beijing, pp 1637–1640
25. EEG time series (Nov 2005). <http://www.meb.unibonn.de/epileptologie/science/physik/eegdata.html>. Accessed Nov 2015
26. Andrzejak RG, Lehnertz K, Mormann F et al (2001) Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: dependence on recording region and brain state. *Phys Rev E* 64:061907
27. Marcano-Cedeño A, Quintanilla-Domínguez J, Cortina-Januchs M et al (2010) Feature selection using sequential forward selection and classification applying artificial metaplasticity neural network. In: *36th annual conference on IEEE Industrial Electronics Society, USA*, pp 2845–2850
28. Ferri F, Pudil P, Hatef M et al (1994) Comparative study of techniques for large-scale feature selection. *Pattern Recognit Pract IV*:403–413
29. Pudil P, Novovičová J, Kittler J (1994) Floating search methods in feature selection. *Pattern Recognit Lett* 15(11):1119–1125
30. Reunanen J (2003) Overfitting in making comparisons between variable selection methods. *J Mach Learn Res* 3:1371–1382
31. Suykens JA, Vandewalle J (1999) Least squares support vector machine classifiers. *Neural Process Lett* 9(3):293–300
32. LS-SVMlab toolbox (version 1.8). <http://www.esat.kuleuven.ac.be/sista/lssvmlab/>. Accessed Nov 2015
33. Han J, Kamber M, Pei J (2011) *Data mining: concepts and techniques*, 3rd edn. Elsevier, Amsterdam

**Hadi Ratham Al Ghayab** received his BSc Degree in Computer Science from Thi\_Qar University in 2007 and MSc Degree in Information Technology from University Utara Malaysia in 2010. He is currently a PhD Student in the Faculty of Health, Engineering and Sciences, University of Southern Queensland Toowoomba. His research interests include biomedical signal analysis, data mining and image processing.

**Yan Li** is currently an Associate Professor of Computer Sciences in the Faculty of Health, Engineering and Sciences at the University of Southern Queensland, Australia. Her research interests are in the areas of Big Data Technologies, Artificial Intelligence, Biomedical Engineering, and Signal/Image Processing.

**Shahab Abdulla** received his BSc and MSc Degrees from University of Technology Baghdad and PhD from USQ. He is currently a Lecturer in Language Centre, University of Southern Queensland. His research interests are in the areas of biomedical engineering, complex medical engineering, networked system, intelligent control, computer control systems, robotics and mathematics research, etc.

**Mohammed Diykh** received his BSc Degree in Computer Science from Thi\_Qar University and MSc Degree in Information Technology from Voronezh State University in 2002 and 2010, respectively. He is currently a PhD Student in the Faculty of Health, Engineering and Sciences, University of Southern Queensland Toowoomba. His research interests include biomedical signal analysis, data mining and image processing, etc.

**Xiangkui Wan** is a Professor of the school of Electrical and Electronic Engineering, Hubei University of Technology. He received MS Degree and PhD Degree in Mechanical and Electronic Engineering from the Chongqing University, in 2002 and 2005, respectively. From 2005 to 2008, he was with the Department of Information Engineering, Guangdong University of Technology, as a Lecturer. Since 2008, he was as an Associate Professor in the same Department. And since 2014, he is with the School of Electrical and Electronic Engineering, Hubei University of Technology. His professional research interests are in signal processing of biomedical signals, artifact and noise analysis and linear and nonlinear time-series analysis.

## 2.2 Chapter Summary

Al Ghayab et al. (2016) concentrates on two classes of EEG signals from healthy people and epileptic patients. A SRS\_SFS method was presented to extract and select the key features for classifying EEG signals into two classes. In order to classify these classes, the LS\_SVM classifier was applied. Al Ghayab et al. (2016) reported that, the SRS\_SFS is useful for extracting and selecting the EEG features. The proposed method is very efficacious for analysing and classifying two classes of epileptic EEG signals. However, using only two classes of epileptic EEG recording was not good enough to classify the epileptic EEG data with high accuracy. Investigating from other technique, such as an optimum allocation (OA) and power spectrum density estimation, improved the accuracy of epileptic EEG classification with all datasets. The next chapter will be discussing EEG classification of whole epileptic datasets based on OA and power spectrum density estimation.

This page is intentionally left blank

# 3

## CHAPTER 3

### **EPILEPTIC EEG SIGNAL CLASSIFICATION USING OPTIMUM ALLOCATION BASED POWER SPECTRAL DENSITY ESTIMATION**

In chapter 2, the SRS and SFS technique denoted as SRS\_SFS based on the LS\_SVM classifier is developed to classify the epilepsy patients from healthy people. The experimental results show that the SRS\_SFS technique works well to classify the epileptic seizures in EEG signals. However, the SRS\_SFS method is applied on two sets of the EEG time series, which are set A and set E rather than using whole database of EEGs and also the SRS method does not use all sample points to extract the representative features.

This chapter is an exact copy of a published paper in IET Signal processing (2018). It evolves a novel approach blending optimum allocation (OA) technique and spectral density estimation to analyse and classify epileptic EEG signals. This study employs the OA to determine representative sample points from the original EEG data and then applies periodogram (PD), autoregressive (AR), and the mixture of PD and AR to extract the discriminative features from each OA sample group. The obtained feature sets are evaluated by three popular machine learning methods: support vector machine (SVM), quadratic discriminant analysis (QDA), and k-nearest neighbor (k-NN). Several output coding approaches of the SVM classifier are tested for selecting the best feature sets. This scheme was implemented on a benchmark epileptic EEG database for evaluation and also compared with existing methods.

Further, appendix B provides Simulation code for the proposed method.



# Epileptic EEG signal classification using optimum allocation based power spectral density estimation

ISSN 1751-9675

Received on 28th March 2017

Revised 9th February 2018

Accepted on 21st February 2018

E-First on 22nd May 2018

doi: 10.1049/iet-spr.2017.0140

www.ietdl.org

Hadi Ratham Al Ghayab<sup>1,2</sup> ✉, Yan Li<sup>1</sup>, Siuly Siuly<sup>3</sup>, Shahab Abdulla<sup>4</sup>

<sup>1</sup>Faculty of Health, Engineering and Sciences, University of Southern Queensland, QLD 4350, Australia

<sup>2</sup>Collage of Computer Sciences and Mathematics, University of Thi-Qar, 64001, Iraq

<sup>3</sup>Centre for Applied Informatics, College of Engineering and Science, Victoria University, Melbourne, Australia

<sup>4</sup>Open Access College, Language Centre, University of Southern Queensland, QLD 4350, Australia

✉ E-mail: HadiRathamGhayab.AIGHayab@usq.edu.au

**Abstract:** This study proposes a novel approach blending optimum allocation (OA) technique and spectral density estimation to analyse and classify epileptic electroencephalogram (EEG) signals. This study employs the OA to determine representative sample points from the original EEG data and then applies periodogram (PD), autoregressive (AR), and the mixture of PD and AR to extract the discriminative features from each OA sample group. The obtained feature sets are evaluated by three popular machine learning methods: support vector machine (SVM), quadratic discriminant analysis (QDA), and  $k$ -nearest neighbour ( $k$ -NN). Several output coding approaches of the SVM classifier are tested for selecting the best feature sets. This scheme was implemented on a benchmark epileptic EEG database for evaluation and also compared with existing methods. The experimental results show that the OA\_AR feature set yields better performances by the SVM with an overall accuracy of 100%, and outperforms the state-of-the-art works with a 14.1% improvement. Thus, the findings of this study prove that the proposed OA-based AR scheme has significant potential to extract features from EEG signals. The proposed method will assist experts to automatically analyse a large volume of EEG data and benefit epilepsy research.

## 1 Introduction

The human brain is made up of a huge number of brain cells that are connected to each other. These links help transform the tiny electric signals generated by their activities and any defect in these connections can lead to a disorder in the brain. Electroencephalograms (EEGs) are the recordings of tiny electrical signals from the electrodes placed on the scalp. The EEG signals are affected by the neural activities. One of the abnormal activities is caused by epilepsy. Epilepsy is a neurological disease that affects the human brain. It causes seizures of involuntary activity that may involve parts or all of the body. Once the seizures appear, they impair patients' daily lives [1]. It is easy to treat the epileptic seizures with inexpensive medication when diagnosed early. However, it is not efficient to identify epileptic EEGs through visual examination by a specialist [2]. In order to detect epileptic seizures, a variety of methods have been developed to analyse and classify the EEG signals. Those methods can be grouped based on feature extraction. There are methods from time domain, frequency domain, time–frequency domain, traditional non-linear method, and graph theory [3].

One of the methods in time domain is the principal component analysis (PCA). This approach is used to reduce the large number of data and select the most important components as the features. The researchers in [4–6] applied the PCA to classify the epileptic EEG signals.

In terms of frequency domain, power spectral density (PSD) estimation methods are widely used to extract features in EEG signals. These methods can be grouped into categories of non-parametric approaches and parametric approaches. Periodogram (PD) is a non-parametric method and used to determine the estimate of the PSD [7]. An autoregressive (AR) model is a parametric approach [8]. So far much research work has been done by using the AR method for classifying EEG data [7, 9–11].

A wavelet transform is often employed to extract the features, the wavelet coefficients, in EEG signals. These coefficients represent EEG signals in time and frequency domain. A variety of

wavelet transforms have used the time–frequency domain to analyse and classify the EEG signals [11–13].

Complex networks based on graph theory were developed to classify EEG signals, as reported by [14]. To reduce the dimensionality of the EEGs, a statistical model was employed in that study. The statistical features were mapped into undirected complex networks to extract the representative features. The attributes of the structural networks were extracted and fed to different classifiers. Zhu *et al.* [15] used a horizontal visibility graph to estimate alcoholic EEG signals. A statistical method was employed to select the best channels and detect the abnormal EEG signals. The selected channels were forwarded to two different classifiers [support vector machine (SVM), and  $k$ -nearest neighbour ( $k$ -NN)].

Nicolaou and Georgiou [16] provided a technique to detect epileptic seizures automatically. This technique employed a permutation entropy method to extract the key features from epileptic EEG recordings. Those features were utilised as the input to a SVM classifier. An average of 86.10% accuracy was achieved with the SVM and a permutation entropy. Song and Zhang [17] detected epileptic seizures using EEG signals based on a discrete wavelet transform. This approach decomposed EEG signals into five EEG sub-bands and extracted non-linear features, which are permutation entropy, sample entropy, and Hurst exponent. These features were extracted and used as the features set. An extreme learning machine was applied to evaluate the extracted non-linear features. The approach attained an 85.9% classification accuracy. In order to enhance the accuracy, they developed a genetic algorithm technique to select the most significant features from the extracted features. This method obtained a 94.2% classification accuracy with the extreme learning machine.

Samiee *et al.* [18] used the conception of an adaptive and localised time–frequency for the epileptic seizure detection. A rational discrete short-time Fourier transform was implemented to extract the features from multichannel EEG signals. The extracted features were forwarded to a multilayer perceptron classifier. A 98.1% overall classification accuracy was achieved in that study.



In 2016, Alcin *et al.* [19] applied a time–frequency image based on a grey-level co-occurrence matrix and filter vector method to extract the key features for epileptic EEG classification. The obtained features were utilised as the input into an ELM classifier. A 96.4% overall classification accuracy was achieved. Al Ghayab *et al.* [20] employed a simple random sampling technique combined with a sequential feature selection method to represent different combinations of epileptic EEG features. The extracted features were evaluated by applying a LS-SVM classifier. This technique gained a 99.9% accuracy from the most discriminative epileptic EEG features. A complete ensemble empirical mode decomposition with noise was developed by Hassan and Subasi [21]. From this model, six spectral moment features were extracted and forwarded to a classifier. The researchers applied a linear programming boosting classifier to perform classification and the method achieved 99.2 and 100% accuracy for E versus (A, B, C, D) and E versus A cases, respectively. In the same year, Djemili *et al.* [22] proposed an empirical mode decomposition method to extract features from the epileptic EEG data. Those features were fed to a multilayer perceptron neural network classifier and yielded 100 and 97.7% classification rates between A versus E and D versus E classes, respectively.

More recently, Patidar and Panigrahi [23] presented a new technique to discriminate the seizure-free segments from the epileptic seizure EEG signals by using the tunable  $Q$ -factor wavelet transform and Kraskov entropy. The approach achieved 97.75% accuracy with a least square SVM (LS-SVM) classifier. Satapathy *et al.* [24] used an integration of the best attributes of the artificial Bee colony and a radial basis function network in order to analyse EEG signals, and to classify the epileptic seizures. A 98% of average accuracy was gained with the method and an inverse multi-quadric kernel. Bhati *et al.* [25] designed a time–frequency localised three-band wavelet filter bank to classify the epileptic EEG signals based on a multi-layer perceptron neural network. The designed method was yielded 99.33% classification rate. Also, two techniques were used to extract the key features, which were local neighbour descriptive pattern and one-dimensional local gradient pattern by Jaiswal and Banka [26]. In order to select the most accurate classifier, the researchers applied different classifiers, such as  $k$ -NN, the SVM, the decision tree, and an artificial neural network. The best accuracy gained was 99.82 and 99.8% with the two techniques and the artificial neural network, respectively.

From our team's work, Zhu *et al.* [27] proposed a multi-scale  $K$ -means approach to detect the epileptic seizures from EEG signals. In order to find these seizures, the researchers extracted six key features by using sample entropy method and yielded 100, 99, and 99.1% classification accuracies for three cases (A versus E, B versus E, and AB versus CDE), respectively. In 2014, Zhu *et al.* [28] provided a fast weighted horizontal visibility graph to differentiate the epileptic patients from healthy people. They obtained 100, 93, and 95.4% accuracies for A versus E, D versus E, and ABCD versus E, respectively.

In the same year (2014), a combination of simple random sampling technique with J48 algorithm was implemented and achieved 100, 95.6, and 97.3% classification accuracy for three group: A versus E, B versus E, and C versus E, respectively, by Wang *et al.* [29].

Also, in 2014, Siuly and Li [30] introduced an optimum allocation (OA) technique to select the most represented samples from all EEG samples. A multiclass LS-SVM classifier was employed to classify the features by the OA technique. The study investigated four output coding techniques for the multiclass LS-SVM. The techniques were one versus one (1vs1), one versus all (1vsA), minimum output codes, and error correcting output codes being applied with the multiclass LS-SVM for classifying the multichannel epileptic EEG signals. An average of 99.9% classification accuracy was obtained in that study. Moreover, a clustering technique based on a LS-SVM was proposed by Siuly *et al.* [31]. A 99.9% classification accuracy was yielded by using clustering technique.

In this paper, an automatic feature extraction scheme is proposed based on OA and PSD estimate for extracting discriminative patterns from epileptic EEG data. First, the OA

technique is used in two stages: (i) to determine sample size; and (ii) to partition data. The OA technique is used to reduce the huge number of EEG signals and to extract the most relevant features. Second, the extracted features are mapped into PSD estimation methods to reduce the high dimensionality. These methods are the PD and the AR. The obtained features from the proposed scheme are denoted as OA\_PD, OA\_AR, and OA\_PD\_AR sets. Then, the feature sets are fed to three popular classifiers, which are a SVM, quadratic discriminant analysis (QDA), and  $k$ -NN. For further investigation several output coding schemes for the SVM classifier, including 1vs1, all pairs (AP), 1vsA, binary complete (BC), ternary complete (TC), ordinal (OR), sparse random (SR), and dense random (DR), are tested for selecting the best parameters.

From the above discussion of the existing methods, there are several challenges in the epileptic EEG classification techniques. One of the challenges is their limited effectiveness. For example, some techniques do not work effectively for big data. Second, all the methods that applied the SVM classifier did not apply any of the eight output coding techniques: namely 1vs1, AP, 1vsA, BC, TC, OR, SR, and DR. In this research, the OA combined with a PSD estimate method is proposed to classify EEG signals.

The organisation of the rest of the paper is as follows. In Section 2, the data used in this study are described. In Section 3, the methodology of the proposed method is illustrated. The performance measurements are introduced in Section 4. Section 5 discusses the experimental results. Finally, the conclusion is presented in Section 6.

## 2 Epileptic EEG data

In this study, popular datasets have been used which were utilised in many research studies [16, 18, 20, 30, 32]. The datasets are open source and available in [33]. They were collected by Bonn University, Germany. The datasets include five classes of EEG recordings and are noted as: class A and class B from five healthy volunteers with eyes open and closed, respectively, while classes C, D, and E from another five epileptic patients. Classes C and D were recorded from five epileptic subjects during inter-ictal time periods (free of seizures). Class E was taken from five patients through ictal duration (seizures activity). Each class has 100 channels of EEG signals with each having 4096 data points. The EEG time duration for each class was 23.6 s, as explained in [34].

## 3 Proposed methodology

This study proposes a dynamic method to classify EEG epileptic data. Fig. 1 clarifies the structure of the proposed method. The sample size is calculated to determine the size of the selected samples (data points) in each class. This study divides EEG signals into  $k$  segments for making the signal quasi-stationary. The OA technique is applied to reduce the big data of EEG signals and select the most significant samples in the set of OA\_sample. This study also investigates whether the PSD estimation methods are suitable for the OA technique for reducing the dimensionality of epileptic EEG signals.

Based on a PD and an AR, these estimation methods of PSD are conducted to generate three distinct feature sets which are denoted as OA\_PD, OA\_AR, and OA\_PD\_AR. The features sets with various classifiers are extracted and tested. The classifiers are a SVM, QDA, and  $k$ -NN, which are popular and widely used. All details of the proposed scheme are explained in the following sections.

### 3.1 Feature extraction

EEG signals normally contain a huge number of data points. Feature extraction techniques are applied to reduce the data size, and to achieve better performance. In this framework, the feature extraction stage includes several phases in this paper. Those phases are explained in the following subsections. The diagram of the methodology is shown in Fig. 1.

**3.1.1 Sample size calculation (SSC):** In this study, we need to calculate the sample size from each class of the EEG data (classes

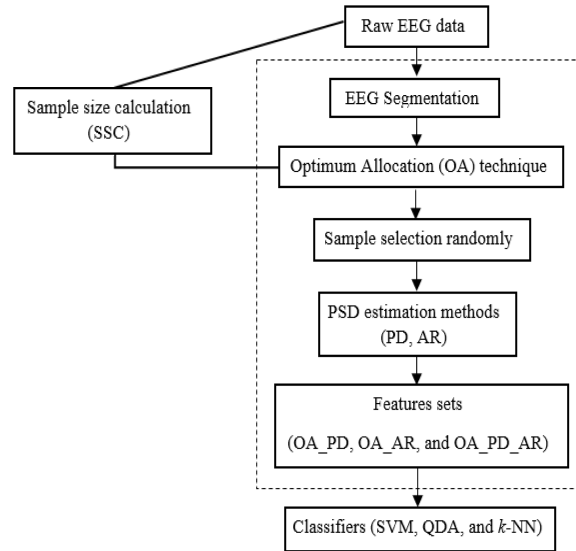


Fig. 1 Block diagram of the proposed scheme for the automatic epileptic EEG signals classification

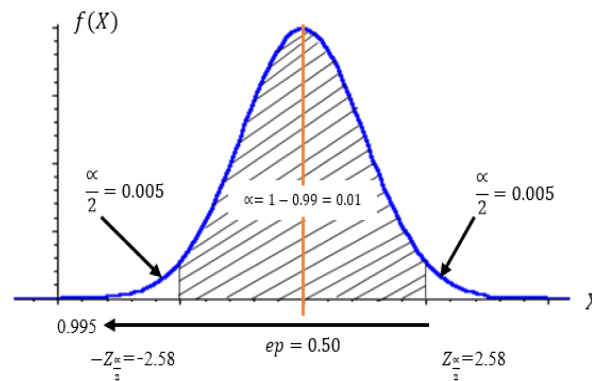


Fig. 2 Curve of the standard normal distribution to calculate Z-score of confidence level 99%

A–E of EEG recordings as mentioned in Section 2). The samples are a part of a statistical population whose properties are studied to obtain the information about the whole data. In this study, a number of samples are chosen from each class to extract key features. Thus, the sample size is the number of the key features that are included in the sample. It is determined by using sample size calculator software [35]. The formula of the SSC is presented below [6, 36]:

$$SSC = \frac{Z^2 \times ep \times (1 - ep)}{\alpha^2} \quad (1)$$

where SSC is the calculated sample size;  $Z$  refers to the value of confidence level ( $Z$ -score);  $ep$  is the estimated proportion of a feature in the whole EEG data (if  $ep$  is not known the biggest sample size needs to be created). The value of  $ep$  is set at 0.5;  $\alpha$  is the confidence interval (margin of error).

In order to select the appropriate  $Z$ -score and  $\alpha$  when the confidence level is 99%, the  $\alpha = 0.01$  and  $\alpha$  is split evenly into two parts as shown in Fig. 2. From Fig. 2 the calculated  $Z$ -score is ( $Z = 2.58$ ) [37]. Based on (1), the SSC is calculated for the whole dataset for the five epileptic EEG classes. This research used (2) to calculate the SSC for each class [6]

$$SSCoC = \frac{SSC}{1 + (SSC - 1)/C} \quad (2)$$

where SSCoC is the calculated sample size for each class (refer to Section 3.1.1 for details);  $C$  is the size of an EEG class ( $C = 4097$ ). From (2), we obtain the sample size of each class as SSCoC = 3288.

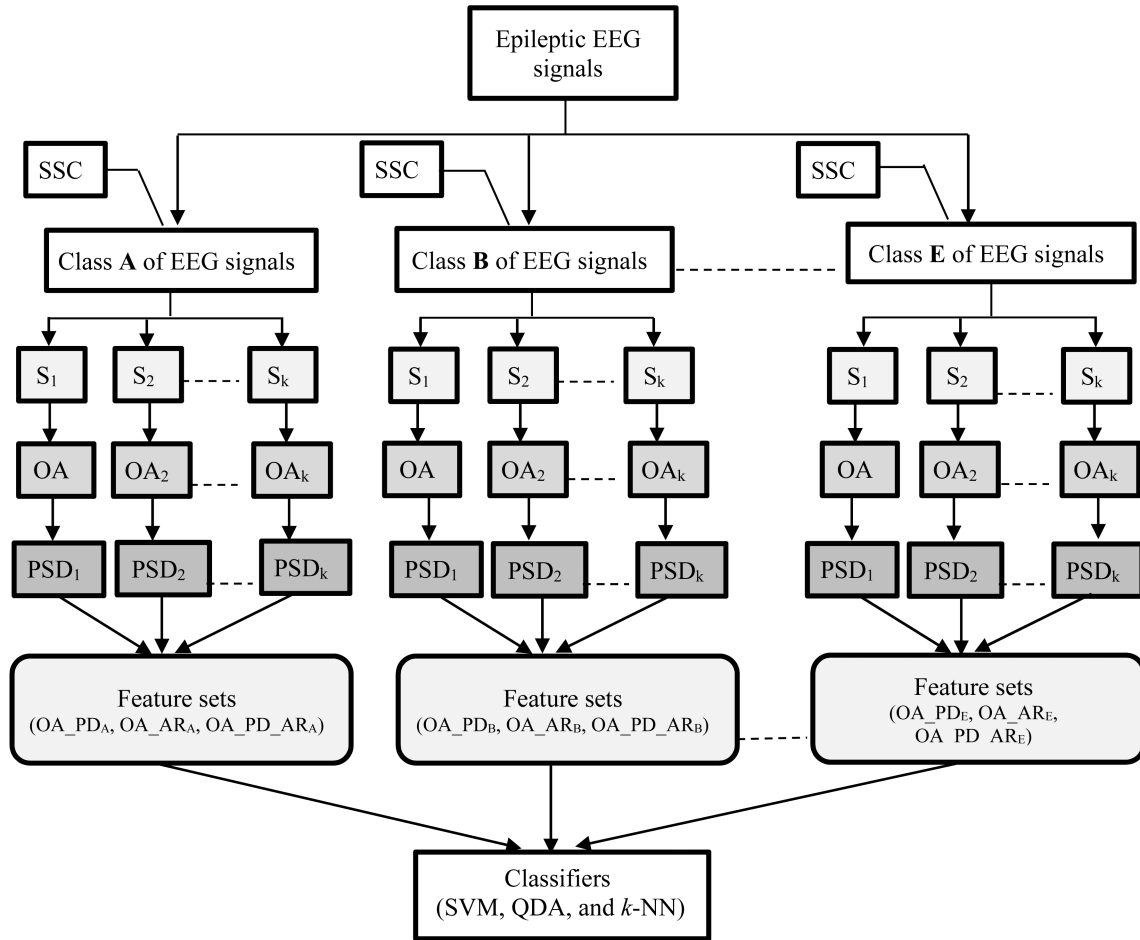
**3.1.2 EEG segmentation:** EEG signals are non-stationary, having a random distribution or pattern that can be analysed statistically but cannot be predicted accurately [30]. To make a signal more stable for obtaining better classification results, the signal was split into smaller segments, with a short time period.

The signal becomes quasi-stationary within the small period of time. In this study, the EEG recordings were divided into a number of small units called segments ( $S_1, S_2, \dots, S_k$ ) as shown in Fig. 3.  $K$  segments are chosen experimentally in this paper. The formula of the  $k$ -segment determination is presented below:

$$K = \text{floor}\left(\frac{DS}{P}\right) \quad P = S_1, S_2, \dots, S_n \quad (3)$$

where  $K$  is the number of segments,  $DS$  is the data size in each class, and  $P$  is the number of data points per segment. In this scheme, each class of the EEG signals is partitioned into four  $k$ -segments and each segment has 5.9 s, when the data size of each segment is  $S_1 = 1024$ ,  $S_2 = 1024$ ,  $S_3 = 1024$ , and  $S_4 = 1025$ , respectively. This technique has an ability to analyse the EEG signals in real time. As the proposed algorithm splits the EEG signals into smaller segments with a shorter time period, that makes it easier to apply in real time.

**3.1.3 Optimum allocation:** The OA is an approach used in stratified sampling to allocate the numbers of sample units to different segments to provide the best accuracy [17]. In this framework, the OA is applied to randomly select the number of samples from different segments for each class as appeared in Fig. 3. This technique can predict the number of samples from each class by using (4) [6], based on the SSC as mentioned in Section 3.1.1



**Fig. 3** Extracted feature sets based on the OA and PSD methods. SSC, sample size calculation;  $S_k$ , the number of segments;  $OA_k$ , optimum allocation in  $k$  segment;  $PSD_k$ , power spectral density methods (PD, AR) in  $k$  segments

$$SSC(i) = \frac{S_i \sqrt{\sum_{j=1}^p E_{ij}^2}}{\sum_{i=1}^k (S_i \sqrt{\sum_{j=1}^p E_{ij}^2})} \times SSCoC \quad (4)$$

where  $SSC(i)$  refers to the number of samples selected from each segment,  $S_i$  is the data size of each segment,  $E_{ij}^2$  is the variance of the  $j$ th channel of the  $i$ th segment,  $SSCoC$  is the sample size of each epileptic EEG class calculated by (2). The sum of  $SSC(i)$  from each segment of the EEG class is 3288 samples.

In this schema, the EEG data of each class contains 4096 observations and 100 channels. Each class was divided into four  $k$  segments ( $k=4$ ) (refer to Section 3.1.2 for details) and  $p$  is the number of channels in each EEG class ( $p=100$ ). The size of the four segments of  $S_1$ ,  $S_2$ ,  $S_3$ , and  $S_4$  are 1024, 1024, 1024, and 1025, respectively. The OA technique is used to choose the number of samples from each segment of that epileptic EEG class.

**3.1.4 Power spectral density estimate method:** In this study, two estimate approaches of PSD are utilised. One of these methods is PD. The PD is basically equal to the Fourier transform of the biased autocovariance, which is a non-parametric method. This method can detect the EEG power density of the frequency components in a signal. The formula of the PD is presented in the following equation [7]:

$$\widehat{PD}(f) = \frac{\Delta t}{N} \left| \sum_{n=0}^{N-1} x_n e^{-i2\pi f n \Delta t} \right|^2, \quad -1/2\Delta t < f \leq 1/2\Delta t \quad (5)$$

where  $\widehat{PD}(f)$  is the estimated value (PD),  $\Delta t$  is the sampling interval,  $N$  is a number of samples in a signal,  $x_n$  denotes the signal, and  $f$  is the sampling frequency. In this study, after the OA technique is applied to select the most discriminative sample from

each segment, the PD is used to extract the power density values from the selected samples as in Fig. 3.

Also an AR method is used in this investigation. The AR model is a linear prediction method to predict an output of a signal based on the previous outputs. The AR method is a spectral density estimation method, which is a parametric method for the PSD estimation [8]. The AR is used to determine the features of an EEG signal in this paper. The order  $p$  of the AR is presented below [9]:

$$AR_t = - \sum_{m=1}^p \alpha_m AR(t-m) + \epsilon_t \quad (6)$$

where  $AR_t$  is a time series to be modelled,  $p$  refers to the number of time points in the past,  $\alpha_m$  is modelling coefficients, and  $\epsilon_t$  is a white noise (error term) which is selected best on the previous points, respectively. From the data  $p+2$  parameters, which are the coefficients, expected sample value, and variance of error term, are estimated in model 1. There are a set of equations, which are used to resolve the estimation problem of these parameters. The AR power spectral estimation of data from the order  $p$  is given in the following equation [7]:

$$p(f) = \sigma_p^2 \Delta t / \left| 1 + \sum_{m=1}^p \alpha_{pm} \exp(-j2\pi f m \Delta t) \right|^2 \quad (7)$$

where  $\alpha_{p0} = 1$ . In order to estimate the AR parameters, only  $p$  number of  $\alpha_{pm}$  and  $\sigma_p^2$  parameters are appropriate AR coefficients, which can be used to identify the amplitude rates, and can be calculated using the Burg method. The Burg algorithm, which is the most popular algorithm for estimating parameters, has been applied in this research [7, 38–40]. The initial values of the

**Table 1** Results of the model selection of  $p$  for AR technique

P order Classifier	Accuracy, %		
	$p = 6$ OA_AR	$p = 7$ OA_AR	$p = 8$ OA_AR
SVM	100	100	100
QDA	99.92	99.94	99.84
$k$ -NN	100	100	99.96

algorithm for computing the parameters of the AR for order  $p$  are chosen as appeared in formulas (8)–(14)

$$\alpha_{11} = \frac{-R_{xx}(1)}{R_{xx}(0)} \quad (8)$$

$$\sigma_1^2 = [1 - \alpha_{11}^2]R_{11}(0) \quad (9)$$

For  $k = 2, 3, 4, \dots$

$$\alpha_{kk} = -R_{xx}(k) + \sum_{m=1}^{k-1} \alpha_{k-1,m} R_{xx}(k-m) / \sigma_{k-1}^2 \quad (10)$$

$$\alpha_{ki} = \alpha_{k-1,i} + \alpha_{kk} \alpha_{k-1,k-i}, \quad i = 1, 2, 3, \dots, k-1 \quad (11)$$

$$\sigma_k^2 = [1 - \alpha_{kk}^2] \sigma_{k-1}^2 \quad (12)$$

where  $R_{xx}(k)$  refers to the estimation of the eigen relationship function of the process. When the processes are finished

$$\alpha_i = \alpha_{pi}, \quad i = 1, 2, 3, \dots, p \quad (13)$$

$$\sigma^2 = \sigma_p^2 \quad (14)$$

Equations (8)–(14) are used to determine the parameters of the AR method. In this method, it is very important to select the order of the model. If the order of the AR method is low, their specific peak does not exist. If the order of this model is very high, misleading and wrong peaks would occur and spectra degenerate. In order to select the order of the AR method, the Akaike information criterion, which is a well-known criterion for selecting the model order [39], is used in this study. The order  $p$  is selected as 7 [7] as shown in Table 1.

### 3.2 Classification methods

After the features extraction, three different feature sets (OA\_PD, OA\_AR, and OA\_PD\_AR) were extracted by applying the OA technique and PSD estimation methods and forwarded into different classifiers as shown in Fig. 3. In this study, the SVM, QDA, and KNN classifiers were employed to select a suitable classifier for the obtained feature sets. All details of these classifiers are explained below.

**3.2.1 Support vector machine:** The SVM was developed by Cortes and Vapnik [41] and has become one of the most widespread classification methods. Generally, the SVM is used to separate the extracted feature sets into two classes through finding an optimal hyperplane. A study by Dagher [42] presented a quadratic kernel-free non-linear SVM which was used in this research. The quadratic function was utilised to split the feature sets non-linearly as can be found in [42]. Furthermore, various output codes are examined to solve the multiclass categorisation problem [43]. In this paper, eight output coding approaches: 1vs1, AP, 1vsA, BC, TC, OR, SR, and DR, are used to reformulate the problem into a set of binary classification problems. The best output coding is selected for the extracted feature sets.

**3.2.2 Quadratic discriminant analysis:** The QDA is a common supervised classification method [44] and is closely related to a linear discriminant analysis. The QDA presumes the covariance

matrix can be different for each class and uses a simple max gate function as a classification rule [45]. The equation of the QDA is given below [46, 47]:

$$f_i(X) = -\frac{1}{2}(X - \mu_i)^T \text{Cov}_i^{-1}(X - \mu_i) + \log \pi_i \quad (15)$$

where  $X$  is a member of class  $i$ ,  $\mu$  refers to the mean value of each class,  $T$  is a transpose operator, and  $\text{Cov}_i$  is a covariance matrix for the  $i$ th class.

**3.2.3  $K$ -nearest neighbours:** The  $k$ -NN is one of the most commonly used classification algorithms as it is among the simplest ones of all machine learning algorithms. The  $k$ -NN is a non-parametric method. This algorithm classifies the extracted key features based on the nearest training features [48]. It aims to classify an unlabelled input to its  $k$ -NN within the training set [35]. One important parameter ( $k$ ) in the  $k$ -NN classifier should be chosen correctly. In this study, the number of the neighbours  $k$  is selected as  $k = 1$ . For details of this algorithm readers may refer to [6, 32, 38, 48–50].

## 4 Classification performance measurements

There are several types of classification measurement methods. One of these assessments is the  $n$ -fold cross-validation method [6]. In this study, a five-fold cross-validation is used as the training and testing processes to achieve the assessment of the classification execution. As can be seen in Table 2, this work obtains a total of 10,260, 2580, and 12,840 feature sets from OA\_PD, OA\_AR, and OA\_PD\_AR, respectively. The cross-validation technique divides the input datasets into five parts. From each partition, one part is used as a testing set for the classification and the remaining parts (four folds) are utilised as classifier training. This process is iterated five times, with each time a different fold being used as the testing set. The training parameters are obtained from the training process, which are used for the testing process to evaluate the algorithm. In each iteration, the training set consists of 8208, 2064, and 10,272 features from the total of feature sets, respectively. Whereas, the testing set consists of 2052, 516, and 2568 features from the same feature sets, respectively, as can be seen in Table 3.

In this procedure, the five classification accuracies can be obtained from each testing set. In the end, the average classification accuracy is obtained as the final performance measurement. Table 4 shows the results from each testing set by the cross-validation technique and their overall classification accuracy.

The performance of this proposed method is also evaluated by using different statistical methods. One of these measurements is the accuracy or the recognition rate, which is the percentage of the correctly classified ones by a classifier for the testing data. The second assessment is sensitivity, which is the proportion of the correctly identified positive set. The other statistical method is specificity, which is the percentage of the correctly identified negative set [20].

## 5 Experimental results and discussions

In this part of study, the experimental results were yielded based on the epileptic EEG signals described in Section 2. The datasets include five classes (class A, B, C, D, or E). The OA technique was used to extract the most discriminative samples based on the following three steps.

**Table 2** Total number of the extracted features from each class of epileptic EEG signals

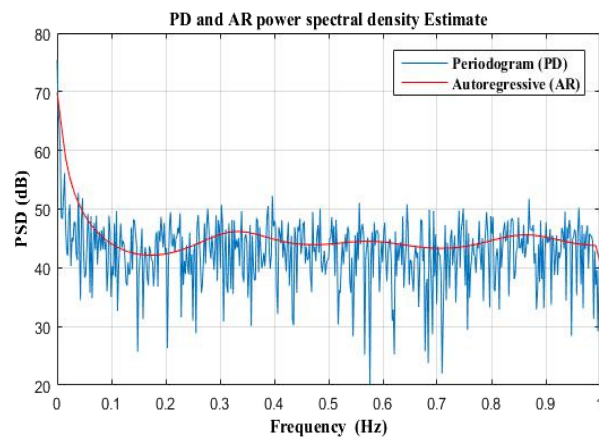
Methods	Dataset					
	Set A dimension	Set B dimension	Set C dimension	Set D dimension	Set E dimension	Total dimension
raw datasets	4097 × 100	4097 × 100	4097 × 100	4097 × 100	4097 × 100	20,485 × 100
OA	3288 × 100	3288 × 100	3288 × 100	3288 × 100	3288 × 100	16,440 × 100
OA_PD	2052 × 100	2052 × 100	2052 × 100	2052 × 100	2052 × 100	10,260 × 100
OA_AR	516 × 100	516 × 100	516 × 100	516 × 100	516 × 100	2580 × 100
OA_PD_AR	2568 × 100	2568 × 100	2568 × 100	2568 × 100	2568 × 100	12,840 × 100

**Table 3** Number of the training and testing sets are used in this study

Methods	Total	Training	Testing
OA_PD	10,260	8208	2052
OA_AR	2580	2064	516
OA_PD_AR	12,840	10,272	2568

**Table 4** Accuracies of the three features sets by using different classifiers

Classifiers	Feature sets		
	Overall accuracy of OA_PD, %	Overall accuracy of OA_AR, %	Overall accuracy of OA_PD_AR, %
SVM (1vs1)	99.8	100	99.9
SVM (AP)	99.8	99.9	99.8
SVM (1vsA)	97.7	99.9	98.2
SVM (BC)	99.6	99.9	99.7
SVM (TC)	99.8	100	99.9
SVM (OR)	99.1	99.9	99.3
SVM (SR)	99.6	100	99.8
SVM (DR)	99.7	99.9	99.7
QDA	98.5	99.9	99.2
k-NN	96.2	100	97.4

**Fig. 4** Example of PD and AR for a single-channel EEG

First, the sample size calculator is used to determine the required sample size. Second, in order to make the EEG signals quasi-stationary, this study divided the signal of each class into smaller segments. From each segment, the representative samples were extracted by using the OA technique based on the determined sample size and named as OA\_sample set. The OA\_sample for each class of EEG signals contains 3288 observations with 100 channels. While the whole OA\_sample set, which is obtained from five classes of the epileptic EEG signals, consists of 16,440 observations of 100 channels. The features went through two PSD methods (PD and AR) for reducing the dimensionality of the extracted features from each segment in an EEG class. The obtained features from the two methods are denoted as OA\_PD, OA\_AR, and OA\_PD\_AR sets as seen in Table 2. The OA\_PD set was yielded when using the OA technique and the PD estimation method on each segment, respectively. On the other hand, the OA\_AR set was extracted by employing the OA and AR techniques on each segment. While, the OA\_PD\_AR set was

obtained by implementing the OA technique and the two PSD estimation methods (PD and AR) together as the features extraction technique on each segment.

Fig. 4 shows an example of the PD and AR estimates for a single-channel EEG. As shown in Table 4, the highest accuracies are obtained by using the SVM classifier with 1vs1 and TC output coding for three feature sets, which are OA\_PD, OA\_AR, and OA\_PD\_AR (complex feature set). The accuracies were 99.8, 100, and 99.9% for the OA\_PD set, OA\_AR set, and OA\_PD\_AR set, respectively, by applying the SVM classifier with two output codings of 1vs1 and TC.

Further, the second highest accuracies for the three feature sets were obtained through the implementation of the SVM with the AP, and the SVM with the SR gained the third highest classification accuracies. Also, the SVM with other output coding of 1vsA, BC, OR, and DR yielded a 97.7, 99.6, 99.1, and 99.7% overall classification accuracies, respectively, for the OA\_PD set.

**Table 5** Accuracies from each testing set by the cross-validation technique and the average accuracies

Classifiers	Methods	Accuracy, %					Average accuracy, %
		Fold <sub>1</sub>	Fold <sub>2</sub>	Fold <sub>3</sub>	Fold <sub>4</sub>	Fold <sub>5</sub>	
SVM	OA_PD	100	99.7	100	99.8	99.7	99.84
	OA_AR	100	100	100	100	100	100
	OA_PD_AR	100	99.7	99.9	99.8	100	99.88
QDA	OA_PD	99.6	97.4	95.6	99.8	100	98.48
	OA_AR	100	100	100	99.5	100	99.9
	OA_PD_AR	99.4	98.8	97.9	99.8	100	99.18
<i>k</i> -NN	OA_PD	98.9	89.2	94.6	98.5	99.8	96.2
	OA_AR	100	100	100	100	100	100
	OA_PD_AR	99.2	92.4	96.3	99.1	100	97.4

**Table 6** Results of sensitivity, specificity, overall accuracy, and overall error for each of the proposed classifiers with the extracted features sets

Classifier	Feature sets	Sensitivity, %	Specificity, %	Overall accuracy, %	Overall error, %
SVM	OA_PD	98.89	99.98	99.8	0.2
	OA_AR	100	100	100	0.0
	OA_PD_AR	99.23	99.98	99.9	0.1
QDA	OA_PD	93.10	99.94	98.5	1.5
	OA_AR	99.61	100	99.9	0.1
	OA_PD_AR	96.53	99.89	99.2	0.8
<i>k</i> -NN	OA_PD	84.35	99.82	96.2	3.8
	OA_AR	100	100	100	0.0
	OA_PD_AR	88.95	99.91	97.4	2.6

In the SVM with the same output techniques, an accuracy of 99.9% was achieved for the OA\_AR set as presented in Table 4.

On the other hand, with the OA\_PD\_AR set, the SVM obtained a 99.2, 99.7, 99.3, and 99.7% accuracy with the same output coding, respectively. Although the *k*-NN classifier only achieved the lowest classification accuracy of 96.2 and 97.4% for the two features sets of the OA\_PD set and the OA\_PD\_AR, respectively, it achieved 100% classification accuracy for the OA\_AR set.

In order to see more details about the fold cross-validation, Table 5 illustrates the obtained accuracy from each testing fold of the cross-validation technique and overall classification accuracy for the proposed approach with three different classifiers, such as SVM, QDA, and *k*-NN. Clearly, from Table 5, the SVM and *k*-NN classifiers achieved 100% classification rates for each testing fold by the OA\_AR scheme.

In addition, Table 6 shows the results of sensitivity, specificity, overall accuracy, and the overall error. Those results were obtained and compared with the proposed classifiers of the SVM with 1vs1 and TC output techniques, QDA, and *k*-NN classifiers. As appeared in Table 6, the best performances with the lowest errors were obtained by the SVM classifier for different feature sets. The QDA classifier yielded the second highest accuracies of 98.5 and 99.2% for the OA\_PD set and OA\_PD\_AR, respectively. For OA\_AR set, the QDA achieved a 99.9% overall accuracy and 0.1% overall error.

In the feature sets of the OA\_PD set and the OA\_PD\_AR set, the lowest results with a high overall error were from the *k*-NN classifier. Obviously, from Tables 4–6, the performance of the SVM classifier is better than the performances of the QDA and *k*-NN classifiers.

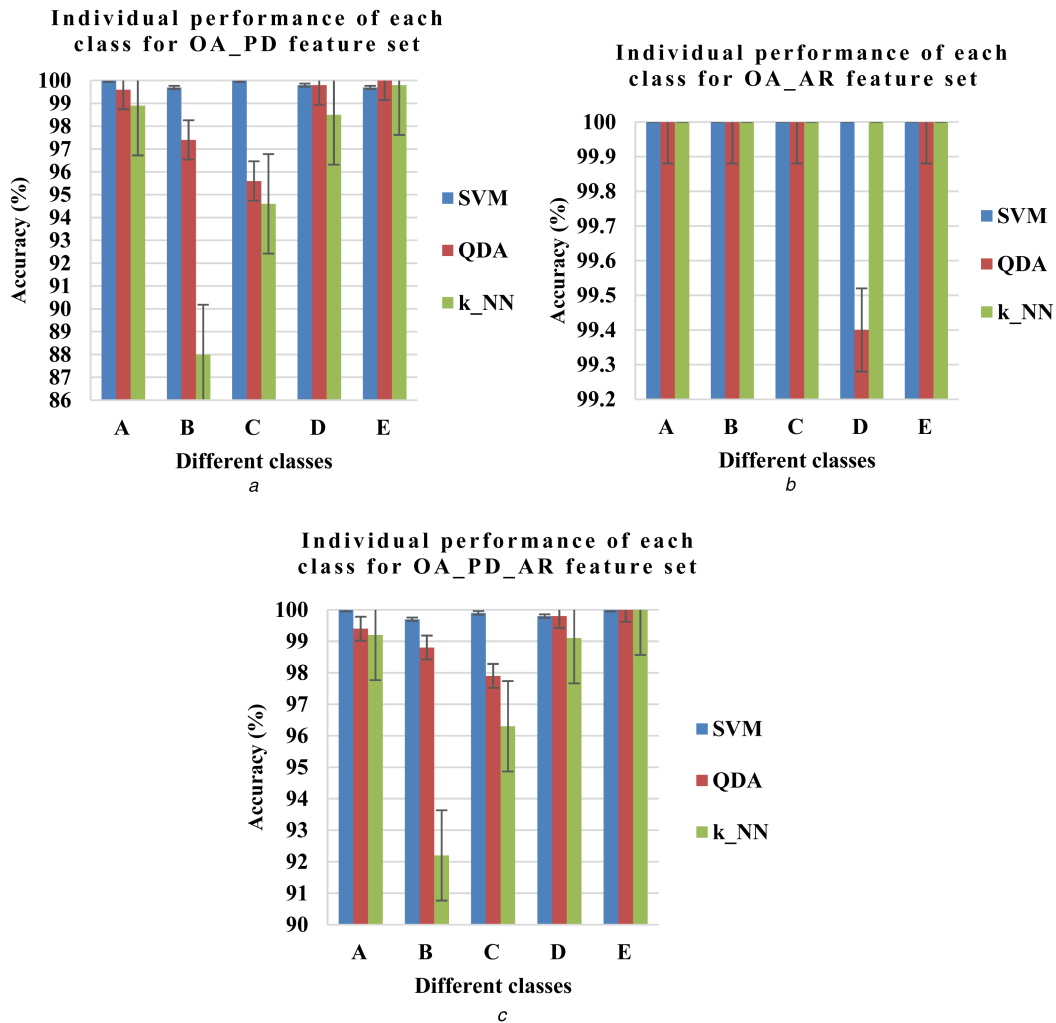
In addition, to investigate a suitable selection of order *p*, this research implemented the AR algorithm in different order of *p* as shown in Table 1. Table 1 shows that, a high classification accuracy was achieved when the *p* value was equal to 7.

Figs. 5a–c present a clear picture of individual performance for the three extracted features from each class. The figures show the classification accuracies from the three different classifiers for each EEG class. The error bars in Fig. 5 represent the standard errors, which indicate the error rates in the performances by the classifiers. As can be seen from Fig. 5, the accuracies from the SVM classifier are significantly better than the accuracy from the other two classifiers with the extracted features from each class.

In addition, this study investigates the effectiveness of the used classifiers with the extracted feature sets. Table 7 illustrates a comparison of the overall accuracy among the classifiers using the proposed scheme and some existing methods from the related work with the same epileptic EEG database. From Table 7, Nicolaou and Georgiou [16] developed an epileptic detection method based on permutation entropy and SVM. Overall, they gained an average of classification accuracy of 86.1%. However, our proposed method achieved higher accuracy than those by Nicolaou and Georgiou [16]. Song and Zhang [17] employed wavelet transform pattern recognition and extreme learning machine and same method with genetic algorithm. These two methods achieved 85.9 and 94.2% average accuracies, respectively. Based on the obtained results, the proposed method outperformed Song and Zhang [17]. Another study was made by Samiee *et al.* [18] in which rational discrete short-time Fourier transform and multilayer perceptron classifier were used. The authors reported a 98.3% classification accuracy for A, B, C, D versus E case. The obtained results in our technique were higher than those by Samiee *et al.* [18].

Alcin *et al.* [19] used time–frequency image based on filter vector and extreme learning machine to detect epileptic seizures. An overall classification accuracy of 96.4% for all sets was yielded. Their result was lower than the proposed scheme. Al Ghayab *et al.* [20] classified two classes (A versus E) using simple random sampling technique and sequential feature selection method with LS-SVM. They achieved 99.9% accuracy rate for class A versus class E. Their result was lower than the proposed method. Hassan and Subasi [21] obtained 100% accuracy for only A versus E sets by using complete ensemble empirical mode decomposition and adaptive noise with a linear programming boosting compared with our method, which used all datasets and yielded 100% accuracy. Although the achieved results by the authors of [22, 27, 28] were 100% accuracies, they used part of datasets that was A group versus E group. The detection performance of the proposed technique was also higher than those by the authors of [24–26, 30, 31].

The existing methods were conducted with two sets and achieved an average of accuracy between 99 and 100% compared with the proposed method which was tested using the whole datasets. The highest overall classification accuracies are highlighted in bold font. The best performance was obtained from the OA technique combined with the AR estimation method as



**Fig. 5** Individual performance for each class with (a) OA\_PD feature set, (b) OA\_AR feature set, (c) OA\_PD\_AR feature set, using different classifiers

well as SVM classifier, compared with the results obtained by the authors of [16–31].

Even though the extracted features by using the OA combined with the mentioned spectral methods were scored the second highest accuracy with a 99.9%, it is considered acceptable in the research of epileptic seizures detection. The results affirmed that the proposed method has achieved a more accurate classification rate than the other existing methods which were conducted with the same datasets.

The advantages of this method are: (i) most of the research that applied the SVM classifier did not apply the eight output coding techniques, which are: 1vs1, AP, 1vsA, BC, TC, OR, SR, and DR. The proposed method tested these output coding techniques to select the best parameters of the SVM. (ii) The outcomes of this research can help physicians and doctors to better diagnose brain disorders. The main advantage of this method is: our proposed OA\_AR technique can reduce big size EEG data into a small set selection the best representative data points from every segmentation of a dataset considering the variability of the observations. Due to the reduction of data, this approach can handle massive size of data with less computation cost compared to the existing methods. One of the limitations of the proposed method is that the delay time could be increased when it implements with a real-time application. However, the proposed method was implemented with off-line datasets collected by Bonn University, Germany.

## 6 Conclusions

In this study, a novel method was developed to classify the epileptic EEG signals. It presented an OA technique combined with two PSD estimation methods: PD and AR, for extracting and

reducing the high dimensionality of EEG recordings. The feature sets were gained and denoted as OA\_PD, OA\_AR, and OA\_PD\_AR sets from the OA technique with the PD estimation method, OA technique with AR method, and OA technique with two PSD estimation methods (PD and AR) together as a features extraction technique, respectively, on each segment of each class from the epileptic EEG data. This paper also investigated the best matching classifier for the extracted features sets by implementing three well-known classifiers: SVM, QDA, and *k*-NN. The experimental results showed that the OA\_AR set achieved better performances with the SVM classifier with 1v1 and TC output coding for the epileptic EEG signals, with an overall classification accuracy of 100%. On the other hand, the OA\_PD\_AR set yielded a 99.9% overall accuracy, which is the second highest accuracy with the SVM.

The proposed algorithm was compared with six existing methods. It was proved that the proposed scheme outperformed the other methods in terms of the accuracy. Also, the proposed method had a possibility to analyse other EEG signals, which can lead to assist physicians to diagnose and treat brain disorders. As a follow-on study, the proposed method will be implemented in real time and applied to other EEG recordings, such as the EEGs from Alzheimer, sleeping disorder, and alcoholic subjects.

## 7 Acknowledgment

The authors acknowledge that the paper has not been funded, and supported from any institute.

**Table 7** Comparisons to the previous studies with the epileptic EEG data

Authors	Techniques	classes	Overall accuracy, %
Nicolaou and Georgiou [16]	• permutation entropy + SVM	A versus E, B versus E, C versus E, D versus E	86.10
Song and Zhang [17]	• wavelet transform pattern recognition + extreme learning machine	A, B versus C, D, E	85.9
	• wavelet transform pattern recognition + genetic algorithm + extreme learning machine	A, B versus C, D, E	94.2
Samiee <i>et al.</i> [18]	• rational discrete short-time Fourier transform + multilayer perceptron classifier	A, B, C, D versus E	98.1
Alcin <i>et al.</i> [19]	• time–frequency image + filter vector + extreme learning machine	All sets (A – E)	96.4
Al Ghayab <i>et al.</i> [20]	• simple random sampling technique + sequential feature selection method + least square support vector machine	A versus E	99.9
Hassan and Subasi [21]	• complete ensemble empirical mode decomposition + adaptive noise + a linear programming boosting	A, B, C, D versus E	99.2
		A versus E	100
Djemili <i>et al.</i> [22]	• empirical mode decomposition + multilayer perceptron neural network	A versus E	100
		D versus E	97.7
Patidar and Panigrahi [23]	• tunable Q-factor wavelet transform + Kraskov entropy + least square support vector machine	C, D versus E	97.75
Satapathy <i>et al.</i> [24]	• integrating the best attributes of artificial Bee colony + radial basis function network	A versus E	72.5
		D versus E	98
		A, D versus E	82.3
Bhati <i>et al.</i> [25]	• time–frequency localised three-band wavelet filter banks + multi-layer perceptron neural network	C, D versus E	99.33
Jaiswal and Banka [26]	• local neighbour descriptive pattern + artificial neural network	A versus E	99.82
	• one-dimensional local gradient pattern + artificial neural network	A, B, C, D versus E	98.72
		A versus E	99.80
		A, B, C, D versus E	98.65
Zhu <i>et al.</i> [27]	• sample entropy + multi-scale K-means	A versus E	100
		B versus E	99.0
		A, B versus C, D, E	99.1
Zhu <i>et al.</i> [28]	• fast weighted horizontal visibility graph + k-NN	A versus E	100
		D versus E	93.0
		A, B, C, D versus E	95.4
Wang <i>et al.</i> [29]	• simple random sampling + J48	A versus E	100
		B versus E	95.6
		C versus E	97.3
Siuly and Li [30]	• OA + multiclass least square support vector machine	multiple classes	99.9
Siuly <i>et al.</i> [31]	• clustering technique + LS-SVM	A versus E	99.9
proposed scheme	• OA_PD + SVM	all sets (A–E)	99.8
	• OA_AR + SVM	all sets (A–E)	100
	• OA_PD_AR + SVM	all sets (A–E)	99.9

## 8 References

- [1] Yuan, Q., Zhou, W., Li, S., *et al.*: ‘Epileptic EEG classification based on extreme learning machine and nonlinear features’, *Epilepsy Res.*, 2011, **96**, (1), pp. 29–38
- [2] Sharma, R., Pachori, R.B.: ‘Classification of epileptic seizures in EEG signals based on phase space representation of intrinsic mode functions’, *Expert Syst. Appl.*, 2015, **42**, (3), pp. 1106–1117
- [3] Acharya, U.R., Sree, S.V., Swapna, G., *et al.*: ‘Automated EEG analysis of epilepsy: a review’, *Knowl.-Based Syst.*, 2013, **45**, pp. 147–165
- [4] Subasi, A., Gursoy, M.I.: ‘EEG signal classification using PCA, ICA, LDA and support vector machines’, *Expert Syst. Appl.*, 2010, **37**, (12), pp. 8659–8666
- [5] Acharya, U.R., Sree, S.V., Alvin, A.P.C., *et al.*: ‘Use of principal component analysis for automatic classification of epileptic EEG activities in wavelet framework’, *Expert Syst. Appl.*, 2012, **39**, (10), pp. 9072–9078
- [6] Siuly, S., Li, Y.: ‘Designing a robust feature extraction method based on optimum allocation and principal component analysis for epileptic EEG signal classification’, *Comput. Methods Programs Biomed.*, 2015, **119**, (1), pp. 29–42
- [7] Kiyimik, M.K., Subasi, A., Ozcalik, H.R.: ‘Neural networks with periodogram and autoregressive spectral analysis methods in detection of epileptic seizure’, *J. Med. Syst.*, 2004, **28**, (6), pp. 511–522
- [8] Percival, D.B., Walden, A.T.: ‘*Spectral analysis for physical applications*’ (Cambridge University Press, London UK, 1993)
- [9] Lawhern, V., Hairston, W.D., McDowell, K., *et al.*: ‘Detection and classification of subject-generated artifacts in EEG signals using autoregressive models’, *J. Neurosci. Methods*, 2012, **208**, (2), pp. 181–189
- [10] Acharya, U.R., Fujita, H., Sudarshan, V.K., *et al.*: ‘Application of entropies for automated diagnosis of epilepsy using EEG signals: a review’, *Knowl.-Based Syst.*, 2015, **88**, pp. 85–96
- [11] Zhang, Y., Liu, B., Ji, X., *et al.*: ‘Classification of EEG signals based on autoregressive model and wavelet packet decomposition’, *Neural Process. Lett.*, 2016, **45**, pp. 1–14
- [12] Nguyen, T., Khosravi, A., Creighton, D., *et al.*: ‘EEG data classification using wavelet features selected by Wilcoxon statistics’, *Neural Comput. Appl.*, 2015, **26**, (5), pp. 1193–1202
- [13] Patidar, S., Pachori, R.B., Upadhyay, A., *et al.*: ‘An integrated alcoholic index using tunable-Q wavelet transform based features extracted from EEG signals for diagnosis of alcoholism’, *Appl. Soft Comput.*, 2017, **50**, pp. 71–78
- [14] Diykh, M., Li, Y.: ‘Complex networks approach for EEG signal sleep stages classification’, *Expert Syst. Appl.*, 2016, **63**, pp. 241–248
- [15] Zhu, G., Li, Y., Wen, P.P., *et al.*: ‘Analysis of alcoholic EEG signals based on horizontal visibility graph entropy’, *Brain Inform.*, 2014, **1**, (1–4), pp. 19–25
- [16] Nicolaou, N., Georgiou, J.: ‘Detection of epileptic electroencephalogram based on permutation entropy and support vector machines’, *Expert Syst. Appl.*, 2012, **39**, (1), pp. 202–209
- [17] Song, Y., Zhang, J.: ‘Automatic recognition of epileptic EEG patterns via extreme learning machine and multiresolution feature extraction’, *Expert Syst. Appl.*, 2013, **40**, (14), pp. 5477–5489
- [18] Samiee, K., Kovács, P., Gabbouj, M.: ‘Epileptic seizure classification of EEG time-series using rational discrete short-time Fourier transform’, *IEEE Trans. Biomed. Eng.*, 2015, **62**, (2), pp. 541–552



- [19] Alçin, Ö.F., Siuly, S., Bajaj, V., *et al.*: 'Multi-category EEG signal classification developing time-frequency texture features based fisher vector encoding method', *Neurocomputing*, 2016, **218**, pp. 251–258
- [20] Al Ghayab, H.R., Li, Y., Abdulla, S., *et al.*: 'Classification of epileptic EEG signals based on simple random sampling and sequential feature selection', *Brain Inform.*, 2016, **3**, (2), pp. 85–91
- [21] Hassan, A.R., Subasi, A.: 'Automatic identification of epileptic seizures from EEG signals using linear programming boosting', *Comput. Methods Programs Biomed.*, 2016, **136**, pp. 65–77
- [22] Djemili, R., Bourouba, H., Korba, M.A.: 'Application of empirical mode decomposition and artificial neural network for the classification of normal and epileptic EEG signals', *Biocybernetics Biomed. Eng.*, 2016, **36**, (1), pp. 285–291
- [23] Patidar, S., Panigrahi, T.: 'Detection of epileptic seizure using Kraskov entropy applied on tunable-Q wavelet transform of EEG signals', *Biomed. Signal Proc. Control*, 2017, **34**, pp. 74–80
- [24] Satapathy, S.K., Dehuri, S., Jagadev, A.K.: 'ABC optimized RBF network for classification of EEG signal for epileptic seizure identification', *Egypt. Inform. J.*, 2017, **18**, (1), pp. 55–66
- [25] Bhati, D., Sharma, M., Pachori, R.B., *et al.*: 'Time–frequency localized three-band biorthogonal wavelet filter bank using semidefinite relaxation and nonlinear least squares with epileptic seizure EEG signal classification', *Digit. Signal Process.*, 2017, **62**, pp. 259–273
- [26] Jaiswal, A.K., Banka, H.: 'Local pattern transformation based feature extraction techniques for classification of epileptic EEG signals', *Biomed. Signal Proc. Control*, 2017, **34**, pp. 81–92
- [27] Zhu, G., Li, Y., Wen, P.P., *et al.*: 'Unsupervised classification of epileptic EEG signals with multi scale K-means algorithm'. Int. Conf. on Brain and Health Informatics, Maebashi, Springer, Cham, 2013
- [28] Zhu, G., Li, Y., Wen, P.P.: 'Epileptic seizure detection in EEGs signals using a fast weighted horizontal visibility algorithm', *Comput. Methods Programs Biomed.*, 2014, **115**, (2), pp. 64–75
- [29] Wang, S., Zhu, G., Li, Y., *et al.*: 'Analysis of epileptic EEG signals with simple random sampling J48 algorithm', *Int. J. Biosci. Biochem. Bioinf.*, 2014, **4**, (2), p. 78
- [30] Siuly, S., Li, Y.: 'A novel statistical algorithm for multiclass EEG signal classification', *Eng. Appl. Artif. Intell.*, 2014, **34**, pp. 154–167
- [31] Siuly, S., Li, Y., Wen, P.P.: 'Clustering technique-based least square support vector machine for EEG signal classification', *Comput. Methods Programs Biomed.*, 2011, **104**, (3), pp. 358–372
- [32] Al Ghayab, H.R., Li, Y., Siuly, S., *et al.*: 'Developing a tunable Q-factor wavelet transform based algorithm for epileptic EEG feature extraction'. Int. Conf. on Health Information Science, Cham, 2017, pp. 45–55
- [33] 'EEG time series: epileptic EEG data', available at <http://www.meb.uni-bonn.de/epileptologie/science/physik/eeegdata.html>, accessed 20 April 2016
- [34] Andrzejak, R.G., Lehnertz, K., Mormann, F., *et al.*: 'Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: dependence on recording region and brain state', *Phys. Rev. E*, 2001, **64**, (6), p. 061907
- [35] 'Sample size calculator', available at <http://www.surveysystem.com/sscalc.htm#two>, accessed 8 July 2016
- [36] Siuly, S., Kabir, E., Wang, H., *et al.*: 'Exploring sampling in the detection of multi-category EEG signals', *Comput. Math. Methods Med.*, 2015, **2015**, pp. 1–12
- [37] 'Computation of Z', available at <http://www.intmath.com/counting-probability/14-normal-probability-distribution.php>, accessed 10 July 2016
- [38] Ergen, B.: 'Scale invariant and fixed-length feature extraction by integrating discrete cosine transform and autoregressive signal modeling for palmprint identification', *Turk. J. Electr. Eng. Comput. Sci.*, 2016, **24**, (3), pp. 1768–1781
- [39] Bozkurt, M.R., Subaşı, A., Köklükaya, E., *et al.*: 'Comparison of AR parametric methods with subspace-based methods for EMG signal classification using stand-alone and merged neural network models', *Turk. J. Electr. Eng. Comput. Sci.*, 2016, **24**, (3), pp. 1547–1559
- [40] Akin, M., Kiyimik, M.K.: 'Application of periodogram and AR spectral analysis to EEG signals', *J. Med. Syst.*, 2000, **24**, (4), pp. 247–256
- [41] Cortes, C., Vapnik, V.: 'Support-vector networks', *Mach. Learn.*, 1995, **20**, (3), pp. 273–297
- [42] Dagher, I.: 'Quadratic kernel-free non-linear support vector machine', *J. Glob. Optim.*, 2008, **41**, (1), pp. 15–30
- [43] Crammer, K., Singer, Y.: 'On the learnability and design of output codes for multiclass problems', *Mach. Learn.*, 2002, **47**, (2–3), pp. 201–233
- [44] Srivastava, S., Gupta, M.R., Frigyik, B.A.: 'Bayesian quadratic discriminant analysis', *J. Mach. Learn. Res.*, 2007, **8**, (Jun), pp. 1277–1305
- [45] Kim, K.S., Choi, H.H., Moon, C.S., *et al.*: 'Comparison of K-nearest neighbor, quadratic discriminant and linear discriminant analysis in classification of electromyogram signals based on the wrist-motion directions', *Curr. Appl. Phys.*, 2011, **11**, (3), pp. 740–745
- [46] James, G., Witten, D., Hastie, T., *et al.*: 'An introduction to statistical learning', vol. 6 (Springer, New York, 2013)
- [47] Friedman, J., Hastie, T., Tibshirani, R.: 'The elements of statistical learning', *Springer series in statistics* (Springer, Berlin, 2001)
- [48] Duda, R.O., Hart, P.E., Stork, D.G.: 'Pattern classification' (John Wiley & Sons, New York USA, 2012)
- [49] Lotte, F., Congedo, M., Lécuyer, A., *et al.*: 'A review of classification algorithms for EEG-based brain–computer interfaces', *J. Neural Eng.*, 2007, **4**, (2), p. R1
- [50] Cover, T., Hart, P.: 'Nearest neighbor pattern classification', *IEEE Trans. Inf. Theory*, 1967, **13**, (1), pp. 21–27

### **3.2 Chapter Summary**

Al Ghayab et al. (2018) developed a novel method to classify the epileptic EEG signals. They presented an optimum allocation (OA) technique combined with two power spectral density estimation methods: periodogram and autoregressive, for extracting and reducing the large amounts of EEG recordings. The extracted features were forwarded to three well known classifiers: SVM, QDA, and k-NN. The experimental results showed that the proposed scheme achieved better performances with the SVM classifier based on (1v1) and (TC) output coding for the epileptic EEG signals.

The proposed algorithm was compared with six existing methods. It was proved that the proposed scheme outperformed the other methods in terms of the accuracy. Furthermore, the proposed method had a possibility to analyse other EEG signals, which can lead to assist physicians to diagnose and treat brain disorders.

This page is intentionally left blank



# 4

## CHAPTER 4

### **A FEATURE EXTRACTION TECHNIQUE BASED ON TUNABLE Q-FACTOR WT FOR BRAIN SIGNAL CLASSIFICATION**

In chapter 2, the SRS technique combined with the SFS algorithm was presented to classify two sets of epileptic EEG data. In chapter 3, the optimum allocation (OA) based power spectral density estimation was presented to analyse and classify all sets of epileptic EEG data.

In this chapter, the details presented here is an exact copy of a published paper in journal of Neuroscience Methods (2018). It explains a new scheme based on a Tunable Q-factor wavelet transform (TQWT) and a statistical approach to analyse various EEG recordings. In the proposed method. Firstly, EEG signals are decomposed into different sub-bands using the TQWT method, which is parameterized by its Q-factor and redundancy. This method depends on the resonance of a signal, instead of frequency or scaling as in Fourier and wavelet transforms. Secondly, using a statistical method on the sub-bands to divide each sub-band into n windows, and then extract several statistical features from each window. Finally, the extracted features are forwarded to a bagging tree (BT), k nearest neighbor (k-NN), and support vector machine (SVM) as classifiers to evaluate the performance of the proposed feature extraction technique. The proposed method is tested on two different EEG databases: Bonn University database and Born University database. The experimental results demonstrate that the proposed feature extraction algorithm with the k-NN classifier produces the best performance compared with the other two classifiers.

Moreover, Appendix C provides MATLAB code for the proposed scheme.



# A feature extraction technique based on tunable Q-factor wavelet transform for brain signal classification

Hadi Ratham Al Ghayab<sup>a,b,\*</sup>, Yan Li<sup>a</sup>, S. Siuly<sup>c</sup>, Shahab Abdulla<sup>d</sup>

<sup>a</sup> Faculty of Health, Engineering and Sciences, University of Southern Queensland, QLD, 4350, Australia

<sup>b</sup> College of Computer Sciences and Mathematics, University of Thi-Qar, 64001, Iraq

<sup>c</sup> Centre for Applied Informatics, College of Engineering and Science, Victoria University, Melbourne, Australia

<sup>d</sup> Open Access College, Language Centre, University of Southern Queensland, QLD, 4350, Australia

## ARTICLE INFO

### Keywords:

Electroencephalogram (EEG) signal

Classification

Epilepsy

Tunable Q-factor wavelet transform

## ABSTRACT

**Background:** Electroencephalogram (EEG) signals are important for brain health monitoring applications. Characteristics of EEG signals are complex, being non-stationarity, aperiodic and nonlinear in nature. EEG signals are a combination of sustained oscillation and non-oscillation transients that are challenging to deal with using linear approaches.

**Method:** This research proposes a new scheme based on a tunable Q-factor wavelet transform (TQWT) and a statistical approach to analyse various EEG recordings. Firstly, the proposed method decompose EEG signals into different sub-bands using the TQWT method, which is parameterized by its Q-factor and redundancy. This method depends on the resonance of a signal, instead of frequency or scaling as in the Fourier and wavelet transforms. Secondly, using a statistical feature extraction on the sub-bands to divide each sub-band into  $n$  windows, and then extract several statistical features from each window. Finally, the extracted features are forwarded to a bagging tree (BT),  $k$  nearest neighbor ( $k$ -NN), and support vector machine (SVM) as classifiers to evaluate the performance of the proposed feature extraction technique.

**Results:** The proposed method is tested on two different EEG databases: Bonn University database and Born University database. The experimental results demonstrate that the proposed feature extraction algorithm with the  $k$ -NN classifier produces the best performance compared with the other two classifiers.

**Comparison with existing methods:** In order to further evaluate the performances, the proposed scheme is compared with the other existing methods in terms of accuracy. The results prove that the proposed TQWT based feature extraction method has great potential to extract discriminative information from brain signals.

**Conclusion:** The outcomes of the proposed technique can assist doctors and other health experts to identify diversified EEG categories.

## 1. Introduction

The human brain comprises billions of neurons that are connected with each other by sending tiny electrical signals. Electroencephalogram (EEG) signals are recorded by using electrodes placed on the scalp. EEG signals indicate the electrical activity of the brain, that is highly random in nature and contain useful information about the brain state to study brain function and neurological disorders (Siuly and Li, 2014; Siuly and Zhang, 2016; Siuly et al., 2013, 2011; Zhu et al., 2012). Analysing brain signals is a very challenging task due to their oscillatory and non-oscillatory transients, nonlinear, aperiodic, and non-stationary dynamic behaviours (Selesnick, 2011a,b; Siuly et al., 2017). It is challenging to extract the most

representative information from raw EEG data for classification. Two types of feature extraction methods, linear and nonlinear (Zhu et al., 2014; Siuly et al., 2010), are generally used for EEG classification.

A number of methods are used to identify various types of brain disorders by EEG analysis. The most commonly used nonlinear classification methods often employ Fourier transform (FT), wavelet transform (WT), or Lyapunov exponent. FT and WT, and other techniques are applied for detecting brain disorders because EEGs contain a combination of sustained oscillation and non-oscillation transients (Selesnick, 2011a, 2011b; Siuly et al., 2017; Zhu et al., 2014; Gajic et al., 2014). The Fourier transform is used to transfer EEG signals from time series into frequency domain in which the most discriminative features are extracted (Kohthoh et al., 2008;

\* Corresponding author.

E-mail addresses: [HadiRathamGhayab.AlGhayab@usq.edu.au](mailto:HadiRathamGhayab.AlGhayab@usq.edu.au) (H.R. Al Ghayab), [Yan.Li@usq.edu.au](mailto:Yan.Li@usq.edu.au) (Y. Li), [siuly.siuly@vu.edu.au](mailto:siuly.siuly@vu.edu.au) (S. Siuly), [Shahab.Abdulla@usq.edu.au](mailto:Shahab.Abdulla@usq.edu.au) (S. Abdulla).

<https://doi.org/10.1016/j.jneumeth.2018.11.014>

Received 24 March 2018; Received in revised form 19 November 2018; Accepted 19 November 2018

Available online 20 November 2018

0165-0270/ © 2018 Elsevier B.V. All rights reserved.

Polat and Güneş, 2007; Murugappan et al., 2014). In order to extract the best features from different bands of wavelets, a wavelet transform is also applied to EEG data (Subasi et al., 2005; Zhang et al., 2016; Lekshmi et al., 2014; Gajic et al., 2014). Wavelet transform is an improved version of Fourier transform that can capture transient features and localize them in both time and frequency domains. A Lyapunov exponent is applied to extract most significant features from EEG data (Pritchard et al., 1995; Adeli et al., 2007; Hosseinfard et al., 2013). To explore the representative samples, an empirical mode decomposition is also used (Acharya et al., 2015; Hassan and Subasi, 2016; Hassan and Haque, 2015; Sharma et al., 2014; Pachori et al., 2015; Alickovic et al., 2018). Several researchers have used entropies from EEG signals as features (Zhang et al., 2016; Pachori and Patidar, 2014; Broberg and Lewis, 2014; Jie et al., 2014; Xiang et al., 2015). However, most of the nonlinear approaches are slow in their implementation, which makes them difficult to use in real time (Diykh et al., 2016; Zhu et al., 2014).

Most recently, the tunable Q-factor WT (TQWT) has become popular in brain signal processing (Patidar et al., 2017; Patidar and Panigrahi, 2017; Patidar et al., 2015; Hassan et al., 2016) and in the other fields (Soleymani et al., 2018; Selesnick, 2011a, 2011b) as a flexible and discrete wavelet transform that is applicable particularly for analysing oscillatory signals (Hassan et al., 2016). Most wavelet transforms are incapable of tuning their Q-factors. The TQWT is able to adjust its Q-factor and has thus emerged as a powerful tool for oscillatory signals analysis. By changing Q value, the wavelet transform can better reflect the signal (Selesnick, 2011a, 2011b). In addition, the TQWT method is developed to process the signals through employing ideal reconstruction oversampled filter banks that is developed in expression of repetition two channel filter banks, with real-valued scaling factors. However, it depends on the resonance (its oversampling rate) of a signal, instead of the frequency, while the band pass filter depends on the frequency of a signal. This study presents a nonlinear algorithm that is sufficiently accurate to analyse and classify EEG signals. This procedure is developed based on both the TQWT and a statistical feature extraction method. Three popular classifiers: bagging trees (BT),  $k$ -nearest neighbor ( $k$ -NN), and support vector machine (SVM), are employed to evaluate the performance of the proposed scheme. This approach is explained in detail in Sections 2.2. Two epileptic databases, which are Bonn University EEG Database and Bern University EEG Database (focal and non-focal EEG signals), are used to test the effectiveness of the proposed methodology in this study.

The rest of the paper is organized as follows. Section 2 provides the description of the experimental data used in this study and the proposed framework. Section 3 presents the experimental results with discussions. Section 4 provides a comparative report with state-of-the-art methods. Finally, Section 5 provides the concluding remarks about this study.

## 2. Materials and methods

### 2.1. Experimental data

In order to test the efficiency of the proposed method, this study uses two epileptic databases: a set of epileptic signals collected by Bonn University denoted as Bonn University EEG database and a set of focal and non-focal epileptic data collected by Bern University denoted as Born University EEG database as described below:

#### 2.1.1. Bonn university EEG database

This database is publicly available, which is widely used by researchers (Siuly and Li, 2014; Zhu et al., 2014; Al Ghayab et al., 2016; Supriya et al., 2016; Al Ghayab et al., 2017; Tzimourta et al., 2018), and was collected by Bonn University, Germany. It contains five different EEG data sets (A–E). From five healthy people, sets A and B were recorded with eyes opened and closed, respectively. Sets C–E were obtained from different five patients. Sets C and D were recorded from epileptic patients free of seizures. Set E was taken from epileptic subjects during active seizures. The EEG recordings were digitized at 173.61 Hz with 12-bit resolution. Consequently, each dataset (A, B, C, D or E) contained 100 single-channels, 4096 sample length, with each set having 23.6 s of time duration to avoid continuous multichannel EEG recordings after visual inspection for artifacts, (e.g., due to muscle activity or eye movements) (Andrzejak et al., 2001).

#### 2.1.2. Bern university EEG database

The second datasets utilized in this study are the Bern Barcelona database that were collected by Bern University and this database is publicly available (The Bern University EEG database, 2012; Andrzejak et al., 2012). Multichannel EEG signals were recorded using the 10/20 system and the sampling rate of the acquisition is 512 Hz. It contains two different EEG signals, focal and non-focal and denoted  $F$  and  $N$ , respectively. Those datasets were recorded from five patients who were suffering from temporal lobe EEG epilepsy. Each recording contains 10,240 observations and 3750 pairs of focal ( $F$ ) and non-focal ( $N$ ) EEG data. In this study, to evaluate the proposed method, full size of this database, 3750 pairs ( $F$  and  $N$  simultaneously) of EEG recording, respectively, with 15,000 sets was utilized. This study processed the full size of data pairs and formed 7500 sets from each group of  $F$  and  $N$ , respectively. Each set having 20 s of time duration.

## 2.2. Methodology

A detailed description of the proposed scheme is presented in this section. The structure of the proposed TQWT based feature extraction method is shown in Fig. 1. The proposed method and three popular classifiers ( $k$ -NN, BT, and SVM) are implemented to evaluate the classification process.

### 2.2.1. Feature extraction

Generally, a huge number of data points are included in the EEG recordings. Redundant data slow down the classification process and often cause inaccurate results. To reduce the data dimensionality and for a better performance, feature extraction techniques are often used in this stage. Fig. 2 illustrates the feature extraction process used in this study.

**2.2.1.1. Tunable Q-factor WT (TQWT).** The TQWT is a newly developed signal decomposition technique. It is an analogous form of the rational-dilation wavelet transform (Bayram and Selesnick, 2009; Selesnick, 2011a, 2011b), and it has been used to analyse EEG signals (Patidar et al., 2017; Bhattacharyya et al., 2017; Al Ghayab et al., 2017). The TQWT depends on changeable parameters: Q-factor (Q), redundancy (R), and decomposition level (J). For the TQWT parameters, Q is often setting at a high value because EEG signals have more oscillations. The

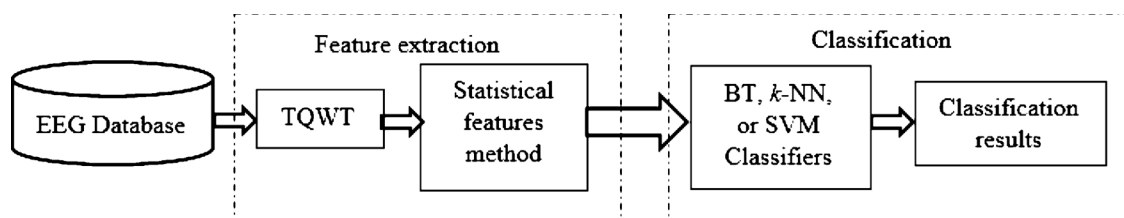


Fig. 1. Block diagram of the proposed scheme for the analysis and classification of EEGs.

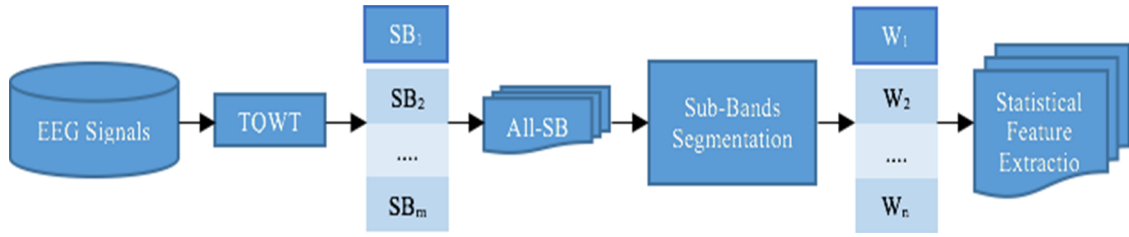


Fig. 2. Feature extraction based on Tunable Q-factor wavelet transform and statistical method. Note: SB = Sub-Bands; All-SB<sub>m</sub> = gathering each Sub-Band from all channels in one set; W<sub>n</sub> = the number of windows.

TQWT decomposes EEG signals into a number of decomposition levels called sub-bands (SB) using the input parameters (Q, R, and J).

The input signal,  $Sig[m]$ , in each level is decomposed into a low pass sub-band ( $Lsig$ ) and a high pass sub-band ( $Hsig$ ) with sampling frequencies of  $af_s$  and  $\beta f_s$ , respectively, where  $\alpha$  and  $\beta$  are low pass scaling and high pass scaling factors, respectively, and  $f_s$  refers to the sampling rate of input  $Sig[m]$ . For this process, a filter bank with two channels is repeatedly utilized and has employed this filter to  $Lsig$  by TQWT.  $Lsig$  is generated by applying a low pass filter  $h_0(\omega)$  with  $\alpha$ . Also,  $Hsig$  is generated from a high pass filter  $h_1(\omega)$  with  $\beta$  at one level of decomposing.  $Hsig$  is further decomposed into its  $Lsig$  and  $Hsig$  at  $j^{\text{th}}$  level as shown in Fig. 3. In this research, Eqs. (1) and (2) are used to calculate  $h_0(\omega)$  and  $h_1(\omega)$  (Patidar and Panigrahi, 2017; Patidar et al., 2015; Hassan et al., 2016; Selesnick, 2011a, 2011b):

$$h_0(\omega) = \begin{cases} 1, & \text{if } |\omega| \leq (1 - \beta)\pi, \\ \theta\left(\frac{\omega + (\beta - 1)\pi}{\alpha + \beta - 1}\right), & \text{if } (1 - \beta)\pi < |\omega| < \alpha\pi, \\ 0, & \text{if } \alpha\pi \leq |\omega| \leq \pi, \end{cases} \quad (1)$$

where  $h_1(\omega)$  can be mathematically expressed as (Patidar et al., 2015; Bayram and Selesnick, 2009):

$$h_1(\omega) = \begin{cases} 0, & \text{if } |\omega| \leq (1 - \beta)\pi, \\ \theta\left(\frac{\alpha\pi - \omega}{\alpha + \beta - 1}\right), & \text{if } (1 - \beta)\pi < |\omega| < \alpha\pi, \\ 1, & \text{if } \alpha\pi \leq |\omega| \leq \pi, \end{cases} \quad (2)$$

where  $\theta(\omega)$  represents Daubechies filter frequency response as (Patidar et al., 2015). R, Q, and J values in the TQWT related to the filter bank parameters,  $\alpha$  and  $\beta$ , as in the equations below (Patidar et al., 2015; Hassan et al., 2016):

$$Q = \frac{(\beta - 2)}{\beta} \quad (3)$$

$$R = \beta/(\alpha - 1) \quad (4)$$

This research used two databases as mentioned in Section 2.1, which have different oscillation transients. For instance, the Q-factor sets to high if the EEG signal has more oscillations. In this study, the Q value was set at 14 for Bonn University database and 6 for the focal and non-focal database, which are selected empirically for the both databases. The different Q values were used because the databases have different oscillatory transients. In addition, the decomposition level J was set empirically for the five sub-bands. Figs. 4(a), (b) and 5(a), (b) present a single EEG signal from the health persons and epileptic patients, and focal/non-focal epileptic EEG, and their five sub-bands which were obtained by the TQWT. From Fig. 4, the difference between F and N data are seen in the last two sub-bands of the TWQT method that means the statistical power is different in these sub-bands. However, the difference from sets A and E appeared in Fig. 5 in each sub-band of the proposed method. Clearly, the statistical power of each sub-band is different among Figs. 5 (a) and (b).

**2.2.1.2. Statistical feature extracted method.** The statistical approach is the most important part of the proposed scheme and employed to extract the representative statistical features. It influences the performance of a classifier if the features are not selected well. The statistical method includes two stages, the segmentation and the statistical feature extraction. The EEG signals are nonlinear and non-stationary in nature (Siuly et al., 2017; Al Ghayab et al., 2018a,b), which makes analysis and classification difficult.

Two methods are employed to make EEG signals quasi-stationary. Firstly, a segmentation technique is utilized to divide each sub-band into several smaller windows that are denoted as  $W_1, W_2, \dots, W_n$ . The window size is determined by the following algorithm:

Algorithm

Input the number of Sub-Bands (SB)

1. **For**  $i = 1$  to the number of SB
2.  $X = SB(i)$
3. **Initially**  $n = 0$
4. let  $p$  be the size of each window
5.  $n = n + 1$
6.  $p = X / n$
7. **Extracting** the statistical features from  $p$
8. **Put** statistical features in one set
9. **Forward** this vector to the classifier
10. **if** the classification accuracy is satisfactory then stop segmentation, go to step 12
11. **else** go to step 5
12. **end**

Output sets of statistical features

Size of each window



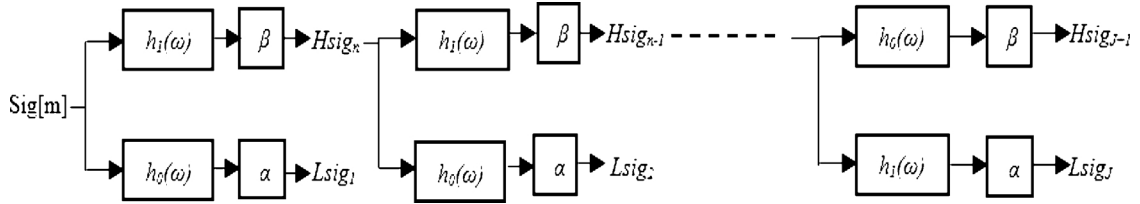


Fig. 3. The TWQT decomposes the input EEG signal ( $Sig[m]$ ) into low pass sub-band  $Lsig$  and high pass sub-band  $Hsig$  at  $J^{th}$  levels.

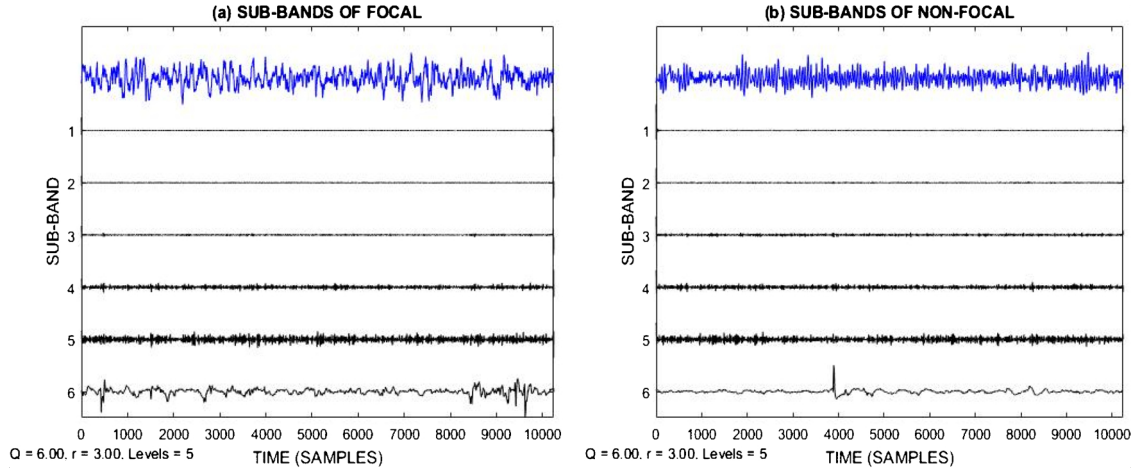


Fig. 4. Examples of Focal ( $F$ ) and Non-focal ( $N$ ) and their five Sub-Bands: (a) Focal =  $F$ ; (b) Non-focal = ( $N$ ), obtained by using the TQWT.

Secondly, ten statistical features are extracted from each window. The features are {minimum (Min), maximum (Max), mean (Mean), median (Med), first quartile (Q1), second quartile (Q2), range (Ran), standard deviation (SD), skewness (Sk), and variance (Var)}. The statistical features can represent important information included in the EEG signals (Siuly et al., 2015a, 2015b) and can be expressed mathematically as:

$$Min = \min(x_n) \tag{5}$$

$$Max = \max(x_n) \tag{6}$$

$$Mean = \frac{1}{n} \sum_1^n x_n \tag{7}$$

$$Med = \left( \frac{N+1}{2} \right)^{th} \tag{8}$$

$$Q1 = \frac{1}{4(N+1)}^{th} \tag{9}$$

$$Q2 = \frac{2}{4(N+1)}^{th} \tag{10}$$

$$Ran = Max - Min \tag{11}$$

$$SD = \sqrt{\sum_{n=1}^N (x_n - SM)^2 \frac{2}{n-1}} \tag{12}$$

$$SK = \sum_{n=1}^N (x_n - SM) \frac{3}{(N-1)SD^3} \tag{13}$$

$$Var = \sum_{n=1}^N (x_n - SM)^2 \frac{2}{N-1} \tag{14}$$

where  $x_n = 1, 2, \dots, n$ , is a time series;  $N$  refers to the number of data

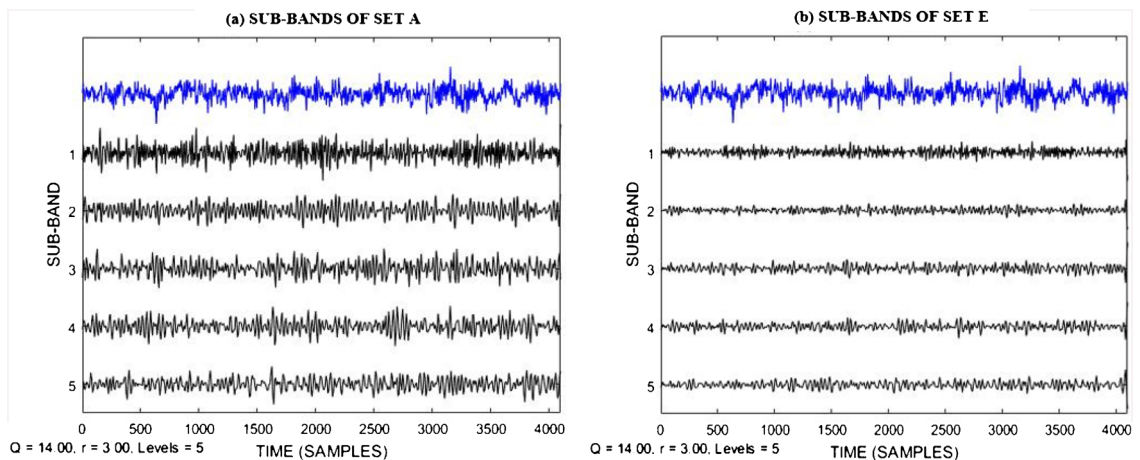


Fig. 5. Examples of five level TQWT decomposition: (a) Healthy persons with eye open (set A); (b) Patient with seizure (set E).

**Table 1**  
Results of ANOVA *f*-test of the statistic feature characteristics for each case of Bonn EEG Dataset.

Statistical Features	<i>f</i> -test							Result
	{A vs E}	{B vs E}	{C vs E}	{D vs E}	{AB vs E}	{AB vs CD vs E}	{A vs B vs C vs D vs E}	
<i>Min</i>	275.84772	348.95716	274.53135	311.2916	275.84772	268.76247	287.99356	S
<i>Max</i>	311.26012	395.90688	310.12819	352.68545	311.26012	302.2518	323.54456	S
<i>Mean</i>	0.02882	0.02576	0.0239	0.0202	0.02882	0.02536	0.02249	NS
<i>Med</i>	0.3794	0.43788	1.19521	0.25751	0.3794	0.26107	0.0568	NS
<i>Q1</i>	0.19645	0.07705	1.40824	0.33093	0.19645	0.13463	0.06509	NS
<i>Q2</i>	0.3141	0.46503	1.28864	0.28364	0.3141	0.22865	0.09816	NS
<i>Ran</i>	654.6591	826.88195	648.32584	736.28225	654.6591	634.67079	676.89236	S
<i>SD</i>	786.35536	982.60613	799.20694	879.65361	786.35536	775.18764	816.60389	S
<i>SK</i>	0.01688	0.02259	0.00003	0.02199	0.01688	0.01202	0.01589	NS
<i>Var</i>	198.36928	209.44634	198.69604	204.06362	198.36928	198.37071	202.228	S

points, *SM* is the mean of the samples.

ANOVA *f*-test is used to test the strength of the characteristics of the extracted features for classifying various brain signal cases, {A vs E}, {B vs E}, {C vs E}, {D vs E}, {AB vs E}, {AB vs CD vs E}, and {A vs B vs C vs D vs E} as shown in Table 1. The ANOVA *f*-test is tested using the one-way ANOVA that takes one random sample from each population under consideration, with a selection of a significance level ( $\alpha$ ) of 0.05.

This investigation found that five of the extracted features (*Min*, *Max*, *Ran*, *SD*, and *Var*) were significant, with  $f > 1.65$ . The features were labeled as “S”, reflecting their significance in representing epileptic activities as shown in Table 1. In Table 1, the NS label indicates that the features with  $f < 1.65$  are not significant for the classification. In this study, we considered the *f* value of 1.65 based on the hypothesis used in this paper that said that each value more than 1.65 is accepted otherwise it was rejected. Choosing value of the *f*-test is depended on the correlation among features. Based on the results, we found that those features gained less than 1.65, were not significant and noisy. As result, they were rejected. For details of how to choose the significant features based on the *f*-test, readers may refer to (Faul et al., 2007; Morgan et al., 2004; Brown and Forsythe, 1974).

At the end, the first and second sub-bands contain 2048 observations and the rest of the sub-bands have 1024 observations. Each sub-band includes 20 data points. Furthermore, each group of the focal and non-focal EEG contains 8192 observations of the first two sub-bands and other three sub-bands have 4096 observations. Next step, we will use the cross-validation measurement technique as can see in Section 2.2.3.

### 2.2.2. Classification method

Three popular machine learning classifiers, bagging trees, *k*-nearest neighbors, and support vector machine are applied to evaluate the proposed features extraction technique and select the most suitable classification method. In the sections below, we present more details about those classifiers.

**2.2.2.1. Bagged trees (BT).** The BT classifier, developed by Breiman (1996), is one of the statistical learning methods. It is employed for the EEG classification in this paper. The equation of the BT classifier is presented below (Breiman, 1996; James et al., 2014):

$$A_{avg}(x) = \frac{1}{P} \sum_{i=1}^P A^i(x) \tag{15}$$

where  $A_{avg}(x)$  is the average of the accuracy of  $x$ ;  $P$  is a separate training sets;  $A^i(x)$  is the tree’s prediction at input  $x$ . In this investigation, 80 training sets were selected from the extracted features to reduce the variance and hence increase the prediction accuracy. The main benefit of this classifier is to gather fitted values from a huge size of bootstrap samples and calculate the average of each fitted values with a low bias and high variance. This way aggregates the

fitted values based on other statistical learning developments (Breiman, 1996).

**2.2.2.2. *K* nearest neighbors (*k*-NN).** The *k*-NN is one of the most common nonparametric methods. It is considered to be the simplest method among all the machine learning algorithms (Duda et al., 2012). The classifier is applied to classify the extracted features based on the nearest training features. It can classify the unlabeled input features to its *k* nearest neighbors (Ergen, 2016). To make the *k*-NN classifier work properly, *k* should be selected carefully (Duda et al., 2012; Ergen, 2016; Cover and Hart, 1967). In this paper,  $k = 1$  is used. The advantage to use the *k*-NN classifier is that this method applies effective techniques to reduce the noise appeared in input data, which are improving the accuracy of the *k*-NN classifier (Cunningham and Delany, 2007).

**2.2.2.3. Support vector machines (SVM).** The SVM is one of the well-known classification methods that was developed by Cortes and Vapnik (1995). A variation of the SVM is a quadratic kernel free non-linear support vector machine (denoted as QSVM) that was proposed by Dagher (2008), is used in this study. This function separates the feature sets nonlinearly as can be found in (Dagher, 2008). The main reason to apply the SVM classifier in this study is that the SVM grants a better generalization solution if its parameters are well chosen. In choosing appropriate parameters, the SVM can be robust (Auria and Moro, 2008).

### 2.2.3. Performance measurements

Several measurements were used for evaluating the performance of the proposed scheme. The *f*-fold cross validation method was used. The datasets were divided into *f* subsets/folds (Siuly and Li, 2015b). In each implementation, one-fold is used as a testing set and *f*-1 folds are utilized as a training set. In each process, the number of features in the training and testing sets are shown in Table 2.

An average accuracy is obtained for the whole process in the cross-

**Table 2**  
Numbers of the training and testing sets used in this study.

Database	Case	Sub-band	Total	Training	Testing
Bonn Database	{A vs E}, {B vs E}, {C vs E}, and {D vs E}	1 <sup>st</sup> and 2 <sup>nd</sup>	4096	3278	818
		3 <sup>rd</sup> , 4 <sup>th</sup> and 5 <sup>th</sup>	2048	1640	408
	{AB vs E}	1 <sup>st</sup> and 2 <sup>nd</sup>	6144	4916	1228
		3 <sup>rd</sup> , 4 <sup>th</sup> and 5 <sup>th</sup>	3072	2459	613
	{AB vs CD vs E}	1 <sup>st</sup> and 2 <sup>nd</sup>	10,240	8195	2045
		{A vs B vs C vs D vs E}	3 <sup>rd</sup> , 4 <sup>th</sup> and 5 <sup>th</sup>	5120	4100
Born Database	{F vs N}	1 <sup>st</sup> and 2 <sup>nd</sup>	16384	13108	3276
		3 <sup>rd</sup> , 4 <sup>th</sup> and 5 <sup>th</sup>	8192	6554	1638

validation method. All the sub-sets are used for testing separately, and the average of classification accuracy is calculated by using 5-fold cross validation as shown in equation (16).

$$ACA = \frac{\sum_1^n perfo}{f}, n = 1,2, \dots, 5 \tag{16}$$

where *ACA* refers to the average of classification accuracy, *f* is the number of folds, and *perfo* refers to the performance of each fold.

The classification accuracy (Acc) was applied to evaluate the proposed method. It is defined as the percentage of the correctly classification result by the classifier for the testing dataset as depicted in Eq. (17) (Zhu et al., 2014; Breiman, 1996; Al Ghayab et al., 2018a, 2018b):

$$Acc = \frac{\sum TP + \sum TN}{\sum AS} \times 100 \tag{17}$$

where *TP* is the true positives, *TN* refers to the true negatives, *FP* is the false positives, *FN* is the false negatives, and *AS* refers to all the samples.

Another statistical measurement used in this study was sensitivity (*SE*). This measurement takes into consideration the true positive rate divided by the positive samples in the test data as shown in Eq. (20) (Al Ghayab et al., 2016):

$$SE = \frac{\sum TP}{\sum P} \times 100 \tag{18}$$

Specificity (*SP*) was applied in this study to evaluate the performance of the algorithm. *SP* considers true negatives divided by the negative samples of the test to obtain a result (Al Ghayab et al., 2016). *SP* is defined as:

$$SP = \frac{\sum TN}{\sum N} \times 100 \tag{19}$$

### 3. Experimental results and discussions

In this research, a set of experiments were implemented by using two different databases, which are described in Section 2.1. The proposed new technique was applied to analyse and extract the key features from both databases (Bonn University EEG and Bern University EEG databases). The TQWT based feature extraction technique was implemented. Two methods were used to decompose the epileptic EEG signals into five sub-bands and extract the most representative features. Firstly, the TQWT nonlinear method was applied to each dataset of EEG signals to decompose the EEG signals based on the resonance of five sub-bands. Secondly, each sub-band was divided into smaller windows, and ten statistical features were extracted from each window. The key features were obtained and forwarded to three classifiers (BT, *k*-NN and SVM), separately, to select the most suitable one. In this study, several cases from the two databases were tested. Also, the experiments were conducted using MATLAB R2017b on a computer with Intel (R) core i7-7700, 3.60 GHz CPU, RAM capacity of 16 GB.

#### 3.1. Experiment results for the Bonn University EEG database

This experiment used seven epileptic cases, which are {A vs E}, {B vs E}, {C vs E}, {D vs E}, {AB vs E}, {AB vs CD vs E}, and {A vs B vs C vs D vs E} as shown in Table 2. The proposed scheme was implemented and evaluated using a variety of evaluation methods. Table 3 shows that the average accuracies of the measurement of each epileptic case were achieved from each sub-band by using the *k*-NN, BT, and SVM classifiers. In Table 3, the TQWT based algorithm decomposed the EEG signals into five sub-bands, which have chosen empirically as mentioned in Section 2.2.1.1, to validate the proposed method in different sub-bands. The overall classification accuracy of the TQWT method with *k*-NN for Bonn University database is 100% in variety cases and difference sub-bands, which was the highest accuracy compared with

**Table 3**

Accuracy rates of each sub-band of TQWT based algorithm for different cases of Bonn University database.

No	Case	Classifier	Acc (%)				
			1 <sup>st</sup> SB	2 <sup>nd</sup> SB	3 <sup>rd</sup> SB	4 <sup>th</sup> SB	5 <sup>th</sup> SB
1	A vs E	<i>k</i> -NN	100	100	100	100	100
		BT	99.98	99.98	100	100	100
		SVM	97.6	96.3	96.3	91.1	92.3
2	B vs E	<i>k</i> -NN	100	100	100	100	100
		BT	100	99.98	100	99.97	99.98
		SVM	95.5	94.9	97.6	85.8	89.6
3	C vs E	<i>k</i> -NN	100	100	100	100	100
		BT	100	100	100	100	100
		SVM	99.1	95.8	96.8	85.8	92.3
4	D vs E	<i>k</i> -NN	100	100	100	100	100
		BT	99.97	100	100	100	100
		SVM	97.7	96.4	96.3	91.5	87.2
5	AB vs E	<i>k</i> -NN	100	100	100	100	100
		BT	99.98	100	100	100	99.98
		SVM	92.3	97.4	96.5	92.4	90.2
6	AB vs CD vs E	<i>k</i> -NN	100	100	100	100	100
		BT	99.82	99.77	99.92	99.67	99.76
		SVM	55.02	57.32	50.66	50.22	48.95
7	A vs B vs C vs D vs E	<i>k</i> -NN	100	100	100	100	100
		BT	99.85	99.83	99.82	99.86	99.82
		SVM	45.71	49.63	47.6	49.55	47.12

**Table 4**

Confusion matrix and accuracy for the case of {A vs B vs C vs D vs E} from Bonn University database using *k*-NN.

		Expert's Scoring				
		A	B	C	D	E
TQWT	A	204	0	0	0	0
	B	0	204	0	0	0
	C	0	0	204	0	0
	D	0	0	0	204	0
	E	0	0	0	0	204
Accuracy		100%	100%	100%	100%	100%

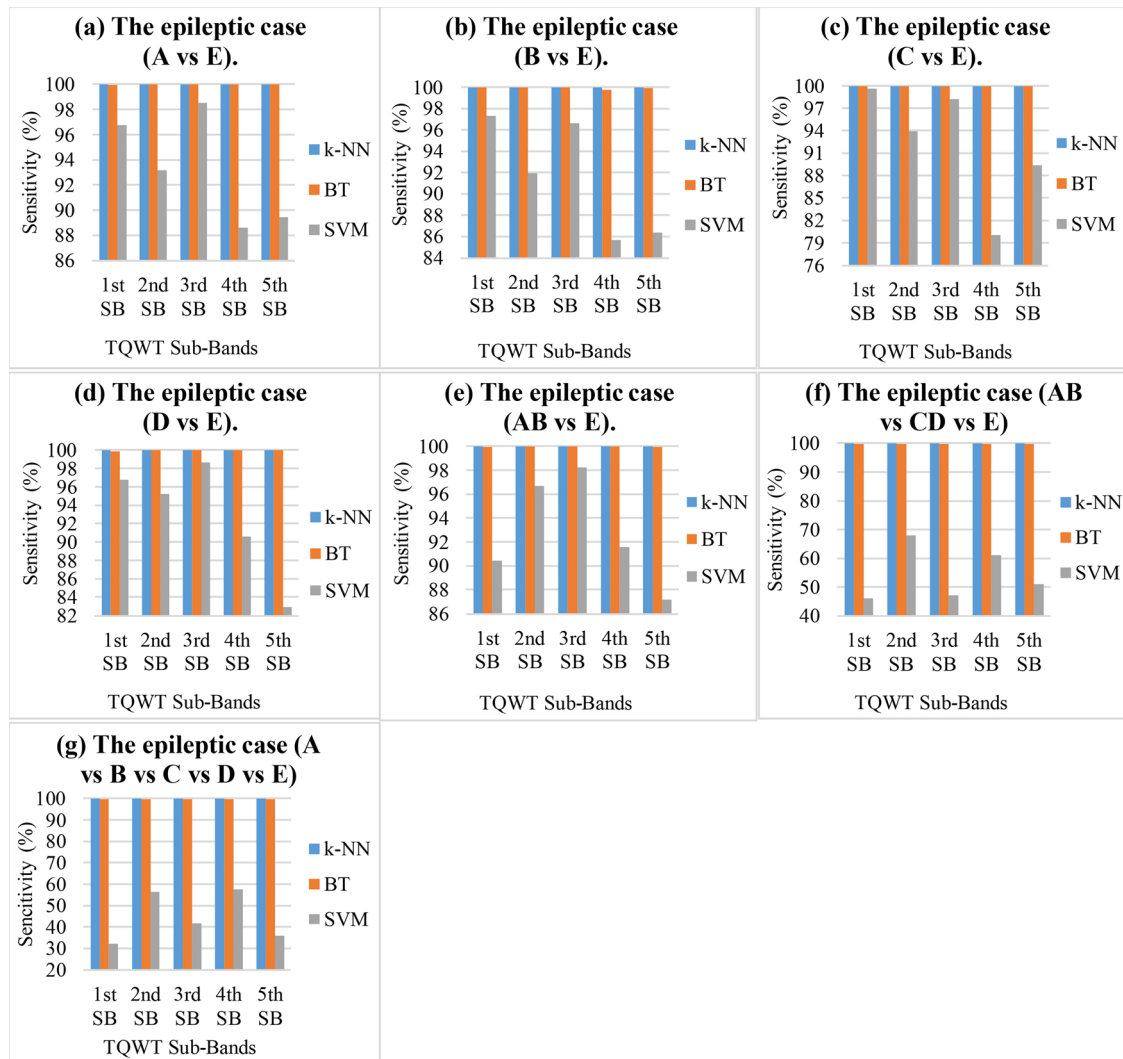
**Table 5**

Confusion matrix and accuracy for the case of {AB vs CD vs E} from Bonn University database using *k*-NN.

		Expert's Scoring		
		AB	CD	E
TQWT	AB	408	0	0
	CD	0	408	0
	E	0	0	204
Accuracy		100%	100%	100%

the BT and SVM classifiers in each sub-band from the seven experimental cases for the epileptic EEG data. In contrast, the lowest accuracy was obtained by using the TQWT with SVM as shown in Table 3.

On the other hand, Tables 4 and 5 illustrate the confusion matrix and accuracy for the cases of {A vs B vs C vs D vs E} and {AB vs CD vs E}, respectively, from Bonn University database. Based on the yielded results there are significant differences in the classification accuracies among the three classifiers (*k*-NN, BT and SVM) compared with the proposed method. The accuracy of the proposed method with *k*-NN classifier in Tables 4 and 5 is 100% overall classification rate, which is the highest accuracy compared with the BT and SVM classifiers in Table 3. The results reveal that the TQWT based algorithm combined with *k*-NN classifier is effective to classify the epileptic EEG signals. Tables 4 and 5 show the confusion matrices between the results by the proposed and the expert's scoring, and their accuracies for the relevant



**Fig. 6.** Performance comparison among the reported classifiers with the proposed feature extraction method for seven cases of Bonn University EEG database in terms of Sensitivity (SN).

pairs.

The suggested approach based on the  $k$ -NN classifier gained the highest score with 100% average of sensitivity in all the sub-bands for the seven epileptic cases. At the same time, the proposed method with the BT achieved a 100% sensitivity rate with most of the sub-bands for all epileptic cases as can be seen in Fig. 6. Fig. 6(a)-(g) provide a comparison of using the proposed feature extraction method with the three classifiers on seven epileptic states in terms of the sensitivity measurement. The lowest sensitivity yielded was 46.23% and 32.32% for the epileptic cases of {AB vs CD vs E} and {A vs B vs C vs D vs E}, respectively. It was from the SVM classifier and the results are presented in Fig. 6. The two cases of {AB vs CD vs E} and {A vs B vs C vs D vs E} that are multi-classification, were similar to each other, which led to the lowest sensitivity results, compared with the results for other cases.

Furthermore, Table 6 shows the average of the specificity rates from each sub-band by the proposed technique for different cases of the epileptic EEG signals. As can be seen in Table 6, the highest results for specificity were achieved by using the suggested approach combined with the  $k$ -NN classifier that was 100% specificity rate in all sub-bands for all cases. In contrast, the TQWT with the other two classifiers (BT and SVM) got low results in some sub-bands. The experiment results from this study demonstrated that the TQWT coupled with the  $k$ -NN classifier was the best method to analyse and classify the epileptic EEG

signals.

### 3.2. Experiment results for the born university EEG database

In this research, we used focal vs non-focal case of Born University database as seen in Table 2. The proposed scheme was implemented and evaluated using the above evaluation measures.

Table 7 presents the accuracy rates that were gained by the same proposed TQWT based scheme and classifiers ( $k$ -NN, BT, and SVM) with the full size of Born University database (refer to Section 2.1.2). In these experiments, the TQWT feature extraction method with the  $k$ -NN gained the highest overall accuracy of 100% for each sub-band, while the BT classifier with the proposed approach achieved > 99% classification accuracies from all the sub-bands. In contrast, the proposed technique with the SVM classifier achieved about 92% overall classification accuracies from all the sub-bands for the  $F$  vs  $N$  case with large datasets. It seems that they are the lowest rate among all the three classifiers for the proposed scheme as can be seen from Table 7.

The sensitivity and specificity were used to evaluate the implementation of the TQWT based algorithm with three popular classifiers for the  $F$  and  $N$  two datasets. This scheme based on the  $k$ -NN classifier obtained a 100% average of sensitivity and specificity for the five sub-bands for  $F$  vs  $N$  case with large datasets, which was the highest outcome. However, the proposed scheme combined with the BT yielded



**Table 6**  
Specificity (SP) rates from each sub-band by the TQWT based algorithm for different cases of Bonn University database.

No	Case	Classifier	SP (%)				
			1 <sup>st</sup> SB	2 <sup>nd</sup> SB	3 <sup>rd</sup> SB	4 <sup>th</sup> SB	5 <sup>th</sup> SB
1	A vs E	k-NN	100	100	100	100	100
		BT	100	100	100	100	100
		SVM	98.51	99.84	94.3	93.85	95.68
2	B vs E	k-NN	100	100	100	100	100
		BT	100	99.95	100	100	100
		SVM	93.8	99.47	98.51	85.98	93.46
3	C vs E	k-NN	100	100	100	100	100
		BT	100	100	100	100	100
		SVM	98.55	97.96	95.45	94.32	95.78
4	D vs E	k-NN	100	100	100	100	100
		BT	100	100	100	100	100
		SVM	98.61	97.64	94.22	92.41	92.71
5	AB vs E	k-NN	100	100	100	100	100
		BT	100	100	100	100	100
		SVM	97.41	98.76	93.21	94.59	100
6	AB vs CD vs E	k-NN	100	100	100	100	100
		BT	99.79	99.69	99.9	99.58	99.67
		SVM	65.37	58.17	61.28	54.04	62.82
7	A vs B vs C vs D vs E	k-NN	100	100	100	100	100
		BT	99.9	99.87	99.9	99.95	99.93
		SVM	89.71	72.99	87.6	69.22	79.85

**Table 7**  
Accuracy, Sensitivity, and Specificity rates for each sub-band by the TQWT based algorithm for Focal (F) vs Non-focal (N) with a data full size from the Bonn University database.

Case	Sub-Band	Classifier	Acc (%)	SE (%)	SP (%)
F vs N	1 <sup>st</sup> SB	k-NN	100	100	100
		BT	99.97	99.95	99.99
		SVM	92.47	92.35	92.59
	2 <sup>nd</sup> SB	k-NN	100	100	100
		BT	100	100	100
		SVM	97.0	97.59	96.45
	3 <sup>rd</sup> SB	k-NN	100	100	100
		BT	100	100	100
		SVM	97.35	97.47	97.23
	4 <sup>th</sup> SB	k-NN	100	100	100
		BT	99.99	99.98	100
		SVM	96.57	96.57	95.82
	5 <sup>th</sup> SB	k-NN	100	100	100
		BT	100	100	100
		SVM	95.65	95.79	95.55

more than 99% sensitivity and specificity rates for the same sub-bands in the F vs N case, which are the second highest results as shown in Table 7. In addition, the low results were from the TQWT and the SVM with results of 92.35% to 97.59% for sensitivity, and 92.59% to 97.23% for specificity for the sub-bands from the F vs N case in Bonn University database.

**4. Comparative study**

In this section, we formulated two types of performance comparisons to evaluate the implementation of the proposed technique. Firstly, the classification accuracies of the TQWT based feature extraction method and k-NN classifier on the epileptic EEG pairs in Table 3 were compared with five recent techniques. Secondly, the accuracy results of the focal and non-focal epileptic data in Table 7 was also compared with the existing methods. In this investigation, the comparisons were made between the proposed scheme and the existing approaches, which used the same databases and the same EEG channels.

1 The comparison among the proposed method and other epileptic

**Table 8**  
Comparative report of the proposed method for the epileptic EEG database.

Authors	Technique	Case	Acc (%)
Zhu et al. (2013)	Sample entropy + Multi-scale K-means	A vs E B vs E AB vs CDE	100 99.0 99.1
Zhu et al. (2014)	Fast weighted horizontal visibility graph + k-NN	A vs E D vs E ABCD vs E	100 93.0 95.4
Samiee et al. (2015)	Rational discrete short time Fourier transform + Multilayer perceptron classifier	A vs E B vs E C vs E D vs E	99.8 99.3 98.5 94.9
Alcin et al. (2016)	Time-frequency image + Filter vector + Extreme learning machine	Multi classification	96.4
Patidar et al. (2017)	TQWT + Kraskov entropy + least square support vector machine	CD vs E	97.75
<b>This study</b>	<b>Tunable Q-factor WT based technique + Statistical features + k-NN</b>	A vs E B vs E C vs E D vs E AB vs E AB vs CD vs E Multi classification	100 100 100 100 100 100 100

EEG classification methods: the results from the proposed method were compared with the results from other studies in (Zhu et al., 2014; Patidar and Panigrahi, 2017; Zhu et al., 2013; Samiee et al., 2015; Alcin et al., 2016). Table 8 shows a comparison of this technique with five other existing methods. From Table 8, Zhu et al. (2013) provides a new technique based on a sample entropy and multi-scale K-means method to classify the epileptic seizures. They achieved an overall classification accuracy of 100%, 99% and 99.1% for the cases of {A vs E}, {B vs E}, and {AB vs CDE}, respectively. However, the proposed scheme in this study gained a 100% accuracy rate for all cases, which means that it achieved a higher accuracy than those by (Zhu et al., 2013). Samiee et al. (2015) applied a rational discrete short time Fourier transform combined with a multilayer perceptron classifier. That paper reported a 99.8%, 99.3%, 98.5%, and 94.9% classification accuracy for the cases of {A vs E}, {B vs E}, {C vs E}, and {D vs E}, respectively. Based on those results, our proposed method outperformed Samiee et al. (2015).

- Another study was made by Zhu et al. (2014) in which fast weighting horizontal visibility graph and k-NN classifier were utilized. The authors reported a 100%, 93%, 95.3% overall accuracy for {A vs E}, {D vs E}, and {ABCD vs E}, respectively. The results in our approach were higher than those by (Zhu et al., 2014). Alcin et al. (2016) obtained a 96.4% average classification accuracy for all the data sets using time frequency image combined with a filter vector and extreme learning machine, compared with our proposed scheme and achieved 100% classification rates. Patidar et al. (2017) applied a TQWT based on Kraskov entropy and least square support vector machine to classify epileptic seizures. An average accuracy of 97.75% for seizure free classes vs epileptic seizure was reported.
- Some of the existing methods were conducted with two sets of Bonn University database and achieved an average of accuracy between 99% and 100% compared with the proposed technique that tested using the whole Bonn University database. In Table 8, the best outcome among all the methods was highlighted in bold font. Based on those results, the TQWT based algorithm with k-NN produced the highest accuracy among all the existing methods.
- The comparison between the proposed TQWT method and other focal and non-focal epileptic EEG classification methods: the outcomes of this method based on k-NN classifier were compared with those of other five existing methods, which are used the same focal and non-focal EEG database. Sharpe et al. (2015) implemented an empirical mode decomposition and least square support vector

**Table 9**  
Comparative report of the proposed method for c the Focal and Non-focal epilepsy EEG datasets.

Authors	Approach	Data size	Acc (%)
Sharma et al. (2015)	Empirical mode decomposition + least square support vector machine	50	87
Das and Bhuiyan (2016)	Empirical mode decomposition + discrete wavelet transform + k-NN	3750	89.4
Arunkumar et al. (2017)	Entropies (approximate entropy, sample entropy, and Reyni's entropy) as features + non-nested generalized exemplars classifier	50	98
Sharma et al. (2017)	Orthogonal wavelet filter banks + entropies + <i>t</i> -test + least square support vector machine	3750	94.25
Arunkumar et al. (2018)	Feature entropies (approximate entropy, sample entropy, and fuzzy entropy) + normal inverse Gaussian + non-nested generalized exemplars	50	99
<b>This study</b>	<b>Tunable Q-factor WT based technique + Statistical features + k-NN</b>	<b>3750</b>	<b>100</b>

machine to classify focal and non-focal epileptic EEG data. Totally, they achieved an accuracy of 87%. However, the proposed TQWT based algorithm gained a higher rate than those by Sharma et al. (2015). Das and Bhuiyan (2016) also applied the empirical mode decomposition combined with the discrete wavelet transform and k-NN classifier. Their approach obtained an 89.4% accuracy for {F vs N}. Based on those results, the proposed TQWT method outperformed Das and Bhuiyan (2016). Different entropies, such as approximate entropy, sample entropy, and Reyni's entropy, have been used as features by Arunkumar et al. (2017). To evaluate their features, they employed non-nested generalized exemplars classifier and achieved an average classification accuracy of 98%.

5 Another study was made by Arunkumar et al. (2018) in which a number of entropies based on normal inverse Gaussian and non-nested generalized exemplars were utilized. Overall, their method obtained 99% classification rate compared with our method, which used the same database and achieved 100% accuracy. Sharma et al. (2017) applied orthogonal wavelet filter banks and entropies. In order to select the most discriminative features, they used *t*-test method. The selected features were forwarded to the least square support vector machine. They reported a 94.25% accuracy for F vs N sets. The obtained results in our scheme were higher than those by Sharma et al. (2017). Some of the existing methods were conducted with smaller datasets excerpted from Bonn University database and achieved an average of accuracy between 87% and 99%. We highlighted in bold font the best classification accuracy among all the compared techniques in Table 9.

Obviously, based on Tables 8 and 9, the proposed scheme achieved the highest accuracy for both two EEG databases, the Bonn University and Bonn University, compared with the existing methods which used the same databases.

One of the limitations of the proposed method is that the proposed method may not work well for real time applications, due to it needs further processing to remove the artifacts. However, the proposed method was implemented with two off-line databases collected by Bonn University and Bonn University. In addition, the main advantage of the TQWT based feature extraction method is: the technique can reduce a huge amount of EEG data into small sets selecting the best representative data points from every segmentation of a dataset considering the variability of the observations. Due to the reduction of data, this approach can handle a large amount of data with less computation cost compared to the existing methods.

## 5. Conclusion

This article developed an innovative technique to extract the most significant features from two different EEG databases, Bonn University EEG and Bonn University EEG databases. The TQWT was applied to decompose the EEG signals into five sub-bands based on its parameters, *Q*, *R*, and *J* level. After signal decomposition, ten statistical features were used to segment the sub-bands and the most representative features were extracted. These features were put in one set for each sub-band and then forwarded to three popular machine-learning classifiers,

k-NN, BT, and SVM. The experimental results reveal that the proposed TQWT based feature extraction method combined with the k-NN classifier is capable to differentiate the epileptic EEG signals with an excellent performance, compared to the existing methods. In the future, we will endeavour to enhance this method and implement it online.

## Acknowledgments

The first author acknowledges the Iraqi government (Ministry of Higher Education and scientific research) for providing him a PhD scholarship to undertake the study.

## References

- Acharya, U.R., Sudarshan, V.K., Adeli, H., Santhosh, J., Koh, J.E., Puthankatti, S.D., Adeli, A., 2015. A novel depression diagnosis index using nonlinear features in EEG signals. *Eur. Neurol.* 74, 79–83.
- Adeli, H., Ghosh-Dastidar, S., Dadmehr, N., 2007. A wavelet-chaos methodology for analysis of EEGs and EEG subbands to detect seizure and epilepsy. *IEEE Trans. Biomed. Eng.* 54, 205–211.
- Al Ghayab, H.R., Li, Y., Abdulla, S., Diykh, M., Wan, X., 2016. Classification of epileptic EEG signals based on simple random sampling and sequential feature selection. *Brain Inform.* 3, 85–91.
- Al Ghayab, H.R., Li, Y., Siuly, S., Abdulla, S., Wen, P., 2017. Developing a tunable Q-factor wavelet transform based algorithm for epileptic EEG feature extraction. *International Conference on Health Information Science* 45–55.
- Al Ghayab, H.R., Li, Y., Siuly, S., Abdulla, S., 2018a. Epileptic EEG signal classification using optimum allocation based power spectral density estimation. *Iet Signal Process.* 12, 738–747.
- Al Ghayab, H.R., Li, Y., Siuly, S., Abdulla, S., 2018b. Epileptic seizures detection in EEGs blending frequency domain with information gain technique. *Soft Comput.* <https://doi.org/10.1007/s00500-018-3487-0>.
- Alçin, Ö.F., Siuly, S., Bajaj, V., Guo, Y., Şengü, A., Zhang, Y., 2016. Multi-category EEG signal classification developing time-frequency texture features based Fisher Vector encoding method. *Neurocomputing* 218, 251–258.
- Andrzejak, R.G., Lehnertz, K., Mormann, F., Rieke, C., David, P., Elger, C.E., 2001. Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: dependence on recording region and brain state. *Phys. Rev. E* 64, 061907.
- Andrzejak, R.G., Schindler, K., Rummel, C., 2012. Nonrandomness, nonlinear dependence, and nonstationarity of electroencephalographic recordings from epilepsy patients. *Phys. Rev. E* 86, 046206.
- Arunkumar, N., Kumar, K.R., Venkataraman, V., 2018. Entropy features for focal EEG and non focal EEG. *J. Comput. Sci.*
- Arunkumar, N., Ramkumar, K., Venkatraman, V., Abdulhay, E., Fernandes, S.L., Kadry, S., Segal, S., 2017. Classification of focal and non focal EEG using entropies. *Pattern Recognit. Lett.* 94, 112–117.
- Auria, L., Moro, R.A., 2008. Support Vector Machines (SVM) As a Technique for Solvency Analysis.
- Alickovic, E., Kevric, J., Subasi, A., 2018. Performance evaluation of empirical mode decomposition, discrete wavelet transform, and wavelet packet decomposition for automated epileptic seizure detection and prediction. *Biomed. Signal Process. Control* 39, 94–102.
- Brown, M.B., Forsythe, A.B., 1974. Robust tests for the equality of variances. *J. Am. Stat. Assoc.* 69 (346), 364–367.
- Bayram, I., Selesnick, I.W., 2009. Frequency-domain design of overcomplete rational-dilation wavelet transforms. *IEEE Trans. Signal Process.* 57, 2957–2972.
- Bhattacharyya, A., Pachori, R.B., Upadhyay, A., Acharya, U.R., 2017. Tunable-q wavelet transform based multiscale entropy measure for automated classification of epileptic EEG signals. *Appl. Sci.* 7, 385.
- Breiman, L., 1996. Bagging predictors. *Mach. Learn.* 24, 123–140.
- Broberg, R., Lewis, R., 2014. Classification of epileptoid oscillations in EEG using Shannon's entropy amplitude probability distribution. *International Conference on Similarity Search and Applications* 247–252.
- Cortes, C., Vapnik, V., 1995. Support-vector networks. *Mach. Learn.* 20, 273–297.
- Cover, T., Hart, P., 1967. Nearest neighbor pattern classification. *IEEE Trans. Inf. Theory*

- 13, 21–27.
- Cunningham, P., Delany, S.J., 2007. k-Nearest neighbour classifiers. *Multiple Classif. Syst.* 34, 1–17.
- Dagher, I., 2008. Quadratic kernel-free non-linear support vector machine. *J. Glob. Optim.* 41, 15–30.
- Das, A.B., Bhuiyan, M.I.H., 2016. Discrimination and classification of focal and non-focal EEG signals using entropy-based features in the EMD-DWT domain. *Biomed. Signal Process. Control* 29, 11–21.
- Diykh, M., Li, Y., Wen, P., 2016. Complex networks approach for EEG signal sleep stages classification. *Expert Syst. Appl.* 63, 241–248.
- Duda, R.O., Hart, P.E., Stork, D.G., 2012. *Pattern Classification*. John Wiley & Sons.
- Ergen, B., 2016. Scale invariant and fixed-length feature extraction by integrating discrete cosine transform and autoregressive signal modeling for palmprint identification. *Turk. J. Electr. Eng. Comput. Sci.* 24, 1768–1781.
- Faul, F., Erdfelder, E., Lang, A.G., Buchner, A., 2007. G\* Power 3: a flexible statistical power analysis program for the social, behavioral, and biomedical sciences. *Behav. Res. Methods* 39 (2), 175–191.
- Gajic, D., Djurovic, Z., Di Gennaro, S., Gustafsson, F., 2014. Classification of EEG signals for detection of epileptic seizures based on wavelets and statistical pattern recognition. *Biomed. Eng. Appl. Basis Commun.* 26, 1450021.
- Hassan, A.R., Siuly, S., Zhang, Y., 2016. Epileptic seizure detection in EEG signals using tunable-Q factor wavelet transform and bootstrap aggregating. *Comput. Methods Programs Biomed.* 137, 247–259.
- Hassan, A.R., Subasi, A., 2016. Automatic identification of epileptic seizures from EEG signals using linear programming boosting. *Comput. Methods Programs Biomed.* 136, 65–77.
- Hassan, A.R., Haque, M.A., 2015. Epilepsy and Seizure Detection Using Statistical Features in the Complete Ensemble Empirical Mode Decomposition Domain. *TENCON 2015-2015. IEEE Region 10 Conference, IEEE*.
- Hosseinifard, B., Moradi, M.H., Rostami, R., 2013. Classifying depression patients and normal subjects using machine learning techniques and nonlinear features from EEG signal. *Comput. Methods Programs Biomed.* 109, 339–345.
- James, G., Witten, D., Hastie, T., 2014. *An Introduction to Statistical Learning: With Applications in R*. An Introduction to Statistical Learning: With Applications in R.
- Jie, X., Cao, R., Li, L., 2014. Emotion Recognition Based on the Sample Entropy of EEG 24. *Bio-medical materials and engineering*, pp. 1185–1192.
- Kohtoh, S., Taguchi, Y., Matsumoto, N., Wada, M., HUANG, Z.L., Urade, Y., 2008. Algorithm for sleep scoring in experimental animals based on fast Fourier transform power spectrum analysis of the electroencephalogram. *Sleep Biol. Rhythms* 6, 163–171.
- Lekshmi, S., Selvam, V., Rajasekaran, M.P., 2014. EEG signal classification using principal component analysis and wavelet transform with neural network. *communications and signal processing (ICCSP). 2014 International Conference on. IEEE*. pp. 687–690.
- Murugappan, M., Murugappan, S., Gerard, C., 2014. Wireless EEG signals based neuro-marketing system using fast fourier transform (FFT). *Signal processing & its applications (CSPA). 2014 IEEE 10th International Colloquium on. IEEE*. pp. 25–30.
- Morgan, G.A., Leech, N.L., Gloeckner, G.W., Barrett, K.C., 2004. *SPSS For Introductory Statistics: Use and Interpretation*. Lawrence Erlbaum Associates, Mahwah, NJ.
- Pachori, R.B., Patidar, S., 2014. Epileptic seizure classification in EEG signals using second-order difference plot of intrinsic mode functions. *Comput. Methods Programs Biomed.* 113, 494–502.
- Patidar, S., Pachori, R.B., Acharya, U.R., 2015. Automated diagnosis of coronary artery disease using tunable-Q wavelet transform applied on heart rate signals. *Knowledge Based Syst.* 82, 1–10.
- Patidar, S., Pachori, R.B., Upadhyay, A., Acharya, U.R., 2017. An integrated alcoholic index using tunable-Q wavelet transform based features extracted from EEG signals for diagnosis of alcoholism. *Appl. Soft Comput.* 50, 71–78.
- Patidar, S., Panigrahi, T., 2017. Detection of epileptic seizure using Kraskov entropy applied on tunable-Q wavelet transform of EEG signals. *Biomed. Signal Process. Control* 34, 74–80.
- Polat, K., Güneş, S., 2007. Classification of epileptiform EEG using a hybrid system based on decision tree classifier and fast Fourier transform. *Appl. Math. Comput.* 187, 1017–1026.
- Pritchard, W.S., Duke, D.W., Kriebel, K.K., 1995. Dimensional analysis of resting human EEG II: surrogate-data testing indicates nonlinearity but not low-dimensional chaos. *Psychophysiology* 32, 486–491.
- Pachori, R.B., Sharma, R., Patidar, S., 2015. Classification of Normal and Epileptic Seizure EEG Signals Based on Empirical Mode Decomposition. *Complex System Modelling and Control Through Intelligent Soft Computations*. Springer, Cham, pp. 367–388.
- Sharma R, Pachori RB, Gautam S. Empirical mode decomposition based classification of focal and non-focal seizure EEG signals. *Medical Biometrics, 2014 International Conference on, IEEE. Samar VJ, Bopardikar A, Rao R, Swartz K. Wavelet analysis of neuroelectric waveforms: a conceptual tutorial. Brain and language, 1999; 66: 7-60.*
- Samiee, K., Kovacs, P., Gabbouj, M., 2015. Epileptic seizure classification of EEG time-series using rational discrete short-time Fourier transform. *IEEE Trans. Biomed. Eng.* 62, 541–552.
- Selesnick, I.W., 2011a. Resonance-based signal decomposition: a new sparsity-enabled signal analysis method. *Signal Process.* 91, 2793–2809.
- Selesnick, I.W., 2011b. Wavelet transform with tunable Q-factor. *Ieee Trans. Signal Process.* 59, 3560–3575.
- Soleymani, R., Selesnick, I.W., Landsberger, D.M., 2018. SEDA: a tunable Q-factor wavelet-based noise reduction algorithm for multi-talker babble. *Speech Commun.* 96, 102–115.
- Sharma, M., Dhere, A., Pachori, R.B., Acharya, U.R., 2017. An automatic detection of focal EEG signals using new class of time–frequency localized orthogonal wavelet filter banks. *Knowledge Based Syst.* 118, 217–227.
- Sharma, R., Pachori, R.B., Acharya, U.R., 2015. Application of entropy measures on intrinsic mode functions for the automated identification of focal electroencephalogram signals. *Entropy* 17, 669–691.
- Siuly, S., Li, Y., Wen, P., 2010. Analysis and classification of EEG signals using a hybrid clustering technique. In: *The Proceedings of 2010 IEEE/ICME. International Conference on Complex Medical Engineering*. pp. 34–39.
- Siuly, S., Kabir, E., Wang, H., Zhang, Y., 2015a. Exploring sampling in the detection of multicategory EEG signals. *Comput. Math. Methods Med.*
- Siuly, S., Li, Y., 2015b. Designing a robust feature extraction method based on optimum allocation and principal component analysis for epileptic EEG signal classification. *Comput. Methods Programs Biomed.* 119, 29–42.
- Siuly, S., Li, Y., Zhang, Y., 2017. *EEG Signal Analysis and Classification: Techniques and Applications*. Springer.
- Siuly, S., Li, Y., 2014. A novel statistical algorithm for multiclass EEG signal classification. *Eng. Appl. Artif. Intell.* 34, 154–167.
- Siuly, S., Zhang, Y., 2016. Medical big data: neurological diseases diagnosis through medical data analysis. *Data Sci. Eng.* 1, 54–64.
- Siuly, S., Li, Y., Wen, P., 2013. Identification of motor imagery tasks through CC-LR algorithm in brain computer interface. *Int. J. Bioinform. Res. Appl. (IJBRA)* 9 (2), 156–172.
- Siuly, S., Li, Y., Wen, P., 2011. Developing a logistic regression model with Cross-correlation for motor imagery Signal recognition. *The Proceedings of 2011 IEEE/ICME International Conference on Complex Medical Engineering* 22-25, 502–507.
- Subasi, A., Alkan, A., Koklukaya, E., Kiyim, M.K., 2005. Wavelet neural network classification of EEG signals by using AR model with MLE preprocessing. *Neural Netw.* 18, 985–997.
- Supriya, S., Siuly, S., Wang, H., Cao, J., Zhang, Y., 2016. Weighted visibility graph with complex network features in the detection of epilepsy. *IEEE Access* 4, 1 6554–66.
- The Bern-Barcelona EEG database, <http://ntsa.upf.edu/downloads/andrzejak-rg-schindler-k-rummel-c-2012-nonrandomness-nonlinear-dependence-and>, accessed 1 March 2012.
- Tzamourta, K.D., Tzallas, A.T., Giannakeas, N., et al., 2018. A robust methodology for classification of epileptic seizures in EEG signals. *Health Technol. (Berl)*. <https://doi.org/10.1007/s12553-018-0265-z>.
- Xiang, J., Li, C., Li, H., et al., 2015. The detection of epileptic seizure signals based on fuzzy entropy. *J. Neurosci. Methods* 243, 18–25.
- Zhang, Y., Liu, B., Ji, X., Huang, D., 2016. Classification of EEG signals based on autoregressive model and wavelet packet decomposition. *Neural Process. Lett.* 1–14.
- Zhu, G., Li, Y., Wen, P.P., 2014. Epileptic seizure detection in EEGs signals using a fast weighted horizontal visibility algorithm. *Comput. Methods Programs Biomed.* 115, 64–75.
- Zhu, G., Li, Y., Wen, P.P., Wang, S., Zhong, N., 2013. Unsupervised classification of epileptic EEG signals with multi scale K-means algorithm. *International Conference on Brain and Health Informatics* 158–167.
- Zhu, G., Li, Y., Wen, P.P., 2012. An efficient visibility graph similarity algorithm and its application on sleep stages classification. *The Proceedings of the 2012 International Conference on Brain Informatics (BI'12), Brain Informatics, Macau SAR, China, December 4-7 (7670)*, 185–195.

## **4.2 Chapter Summary**

Al Ghayab et al. (2018) implemented an innovative technique to extract the most significant features from two different EEG databases, Bonn University EEG database and Born University EEG database. The TQWT was applied to decompose the EEG signals into five sub-bands. Thereafter, statistical features extraction were used to segment these sub-bands and extract the most representative features. The key features were forwarded to three classifiers, k-NN, BT, and SVM. The experimental results reveal that the proposed feature extraction method combined with the k-NN classifier is capable to differentiate the variety of EEG categories with an excellent performance, compared to existing methods.

Al Ghayab et al. (2018) obviously demonstrated that using TQWT method has a high potential to improve the classification way and to detect an abnormal EEG signal.



This page is intentionally left blank



# 5

## CHAPTER 5

### **EPILEPTIC SEIZURES DETECTION IN EEGS BLENDING FREQUENCY DOMAIN WITH INFORMATION GAIN TECHNIQUE**

This chapter is an exact copy of a published article in *Soft Computing* (2018). It presents a new algorithm which combines the information in frequency domain with the Information Gain (InfoGain) technique for the detection of epileptic seizures from electroencephalogram (EEG) data. The process includes four main steps: Firstly, it is necessary to investigate which is most method suitable to decompose the EEG signals into frequency bands, a fast Fourier transform (FFT) or discrete wavelet transform (DWT); Secondly, each band is partitioned into  $k$  windows and a set of statistical features are extracted from each window; Thirdly, the InfoGain is used to rank the extracted features and the most important ones are selected; and lastly, these features are forwarded to a least square support vector machine (LS\_SVM) classifier to classify the EEG. This scheme is implemented and tested on a benchmark EEG database and also compared with other existing methods, based on some performance evaluation measures.

In addition, the simulation code of this method is provided in appendix D.



# Epileptic seizures detection in EEGs blending frequency domain with information gain technique

Hadi Ratham Al Ghayab<sup>1,2</sup> · Yan Li<sup>1</sup> · Siuly Siuly<sup>3</sup> · Shahab Abdulla<sup>4</sup>

Published online: 27 August 2018  
© Springer-Verlag GmbH Germany, part of Springer Nature 2018

## Abstract

This paper proposes a new algorithm which combines the information in frequency domain with the Information Gain (InfoGain) technique for the detection of epileptic seizures from electroencephalogram (EEG) data. The proposed method consists of four main steps. Firstly, in order to investigate which method is most suitable to decompose the EEG signals into frequency bands, we implement separately a fast Fourier transform (FFT) or discrete wavelet transform (DWT). Secondly, each band is partitioned into  $k$  windows and a set of statistical features are extracted from each window. Thirdly, the InfoGain is used to rank the extracted features and the most important ones are selected. Lastly, these features are forwarded to a least square support vector machine (LS-SVM) classifier to classify the EEG. This scheme is implemented and tested on a benchmark EEG database and also compared with other existing methods, based on some performance evaluation measures. The experimental results show that the proposed FFT combined with InfoGain method can generate better performance than the DWT method. This method achieves 100% accuracy for five different pairs: healthy people with eyes open ( $z$ ) versus epileptic patients with activity seizures ( $s$ ); healthy people with eyes closed ( $o$ ) versus  $s$ ; epileptic patients with free seizures ( $n$ ) versus  $s$ ; patients with free seizures epileptic ( $f$ ) versus  $s$ ; and  $z$  versus  $o$ . The accuracies obtained for two other pairs, ( $o$  vs.  $n$ ) and ( $z$  vs.  $f$ ), are 95.62 and 88.32%, respectively. These two pairs have more similarities with each other, leading to a lower level of accuracy. The proposed approach outperforms six other reported methods and achieves an 11.9% improvement. Finally, it can be concluded that the proposed FFT combined with InfoGain method has the capacity to detect epileptic seizures in EEG most effectively.

**Keywords** Electroencephalogram · Epileptic seizures · Frequency domain · Information gain technique · Least square support vector machine

Communicated by V. Loia.

✉ Hadi Ratham Al Ghayab  
HadiRathamGhayab.AlGhayab@usq.edu.au

Yan Li  
Yan.Li@usq.edu.au

Siuly Siuly  
siuly.siuly@vu.edu.au

Shahab Abdulla  
Shahab.Abdulla@usq.edu.au

<sup>1</sup> Faculty of Health, Engineering and Sciences, University of Southern Queensland, Toowoomba, QLD 4350, Australia

<sup>2</sup> Collage of Computer Sciences and Mathematics, University of Thi-Qar, Nasiriyah 64001, Iraq

<sup>3</sup> Centre for Applied Informatics, College of Engineering and Science, Victoria University, Melbourne, Australia

## 1 Introduction

An electroencephalogram (EEG) is a test commonly performed to study the brain's electrical activity. The brain cells communicate with each other producing tiny electrical impulses. An abnormal electrical activity would reflect potential brain disorders. Epileptic seizures may lead to a brief case of involuntary activity that may involve a part or all of the body and be accompanied by loss of consciousness and disturbances of movement, as well as uncomfortable sensation, mood, or mental function for a patient (Yuan et al. 2011). The World Health Organization (2011) reported that approximately 2.4 million people around the world are diagnosed with epilepsy each year. In developing countries, it is

<sup>4</sup> Open Access College, University of Southern Queensland, Toowoomba, QLD 4350, Australia

estimated that between 30 and 50 people per 100,000 people suffer from epilepsy at some stage, while in poor countries this incidence will appear to be up to twice as high (WHO 2011; Mcgrogan 1999). Although it is easy to treat epileptic seizures with inexpensive medication, it is often difficult for experts to identify people who have a brain defect through visual inspection of EEG signals (Siuly and Li 2015). Therefore, it is necessary to develop a new method of analysing and classifying EEG data. This method can help doctors detect and treat brain disorders. There are a variety of techniques being developed to analyse and classify EEG signals (Shen et al. 2013; Siuly and Li 2015; Al Ghayab et al. 2017, 2018; Tzimourta et al. 2018).

Siuly et al. (2015) detected multi-category EEG recording data. Random sampling and optimal allocation sampling were used in their research to identify the important features from each group of EEG classes. The method divided the input signals of each class into different segments. Twelve statistical features from each set (RS or OS set) were extracted, separately. In order to evaluate the performance of the proposed feature extraction method, three different classifiers were applied. Those were  $k$ -nearest neighbour, multinomial logistic regression, and a support vector machine. It was reported that the random sampling technique represented the EEG data and the  $k$ -nearest neighbour with the random sampling achieved good results for each EEG class. Wali et al. (2013) used the theta, alpha and beta bands of EEG signals from the discrete wavelet packet transform and the FFT methods to extract features. The extracted features were based on spectral centroid and power spectral. These features were evaluated by three different classifiers: subtractive fuzzy clustering probabilistic; neural network; and  $k$ -nearest neighbour, for comparing the results. The researchers achieved a 79.21% accuracy by using the subtractive fuzzy classifier. Swami et al. (2016) used three different methods to select the best method for the detection of epileptic seizures. These methods were DWT, wavelet packet transform and dual-tree complex wavelet transform. Their technique applied energy, root-mean-square values and standard deviation as the extracted features. A general regression neural network classifier was utilized in this study to evaluate the performance. A 98% overall accuracy was obtained by using the dual-tree complex wavelet transform technique, which were much more successful other methods. Chua et al. (2011) extracted features from EEG signals by using higher-order spectra. To select the best classifier, the extraction features were forwarded to two different methods to compare their performances. A Gaussian mixture model and a support vector machine classifier were used to detect epileptic EEG signals. The research achieved an average accuracy of 93.11 and 92.56% for higher-order spectra based on the Gaussian mixture model classifier and the support vector machine classifier, respectively, for dif-

ferent EEG classes, such as normal, pre-ictal and epileptic EEGs. A time–frequency technique was proposed to extract features for analysing the EEG signals by Rao and Vishwanath (2014). The DWT method was used to detect sleep disorders, such as a sleep breathing disorder, an insomnia disorder, a bruxism disorder, and one healthy subject. Four levels of decomposition were used to extract EEG features. The extracted features were evaluated by using the following evaluation measurements: energy; variance; waveform length; and standard deviation. All these evaluation measurements were used to detect different sleep disorders. Shen et al. (2013) proposed a cascade of wavelet-approximate entropies for feature selection. This approach utilized fisher scores for adaptive feature selection and support vector machine to detect epileptic seizures. The technique was applied to different epileptic EEG recordings: open source EEG data and clinical EEG data. The method achieved overall classification accuracies of 99.97 and 98.73%, respectively.

A wavelet transform method was used by Gajic et al. (2014) to extract the key features. Scatter matrices were used to reduce the dimensionality of the features. These features were used as the input to a quadratic classifier. The EEG epileptic data were classified into healthy subjects, epileptic subjects during a seizure-free period (inter-ictal), and epileptic patients during seizure activity (ictal). A 99% classification accuracy was reported. The DWT was employed by Amin et al. (2015) for analysing EEG recordings. Two datasets of the Raven's advance progressive metric test and the eyes open test were used to classify them. The research applied different classifiers, including the SVM,  $k$ -nearest neighbouring, and multilayer perceptron classifiers. An accuracy of above 98% was achieved by using those classifiers with approximations and detail coefficients from the DWT. Gajic et al. (2015) extracted different features from time, frequency, time frequency domain, and used a nonlinear analysis. The features were obtained from sub-bands with good representative characteristics compared with those from the whole band. The researchers reduced the dimension of the features by using scatter matrices. That method yielded a 98.7% accuracy. A comparative study was made by Yang and Pedersen (1997) with a different feature selection method to classify texts. Five types of selection algorithm were estimated, as follows: document frequency; information gain (InfoGain); mutual information;  $X^2$ -test (CHI); and a term strength. InfoGain yielded the highest classification accuracy of the above five methods, with 98%.

Most recently, a rational discrete short-time Fourier transform was utilized by Samiee et al. (2015). It was a novel method of extracting features from epileptic EEG signals. To classify the extracted features, they employed a multilayer neural network perceptron classifier. A 98.3% overall classification accuracy was achieved with that technique.

In 2014, Kaya et al. performed a one-dimensional local binary pattern to extract key features from epileptic EEG recordings. The key features of uniform and non-uniform were used as the input to different classifiers, such as Bayes Net, SVM, artificial neural network, logistic regression and the functional trees. 99.5 and 99.0% accuracies were achieved by uniform and non-uniform functional trees, respectively, for classes  $z$  and  $s$ . Wang et al. (2014) implemented a significant method to classify the epileptic EEG signals by using a simple random sampling combined with the J48 decision tree. This approach achieved a 97.6% average classification accuracy for classes  $z$  versus  $s$ , classes  $o$  versus  $s$  and classes  $n$  versus  $s$ . Shen et al. (2013) presented an approach based on a cascade of wavelet approximate entropies to extract features from the epileptic EEG signals. In order to classify the extracted features, three classification methods were tested: support vector machine (SVM);  $K$ -nearest neighbour; and radial basis function neural network. The overall accuracy achieved from these three classification approaches were 99.97, 99.10 and 99.82%, respectively. Also in 2013, Zhu et al. reported an important method which employed the sample entropy with multi-scale  $K$ -means. The sample entropy was used to extract features and measured the complexity time. With the aim of finding the optimum solution, the three different classifiers of  $K$ -means, multi-scale  $K$ -means and a SVM were tested. The sample entropy based on multi-scale  $K$ -means attained a 99.5% average classification accuracy for classes  $z$  versus  $s$  and classes  $o$  versus  $s$ . Nicolaou and Georgiou (2012) established a new method to detect epileptic seizures, using permutation entropy as the extracted feature. The SVM classifier was applied to classify the epileptic EEG signals based on permutation entropy.

As can be seen in the literature review, there are several challenges in the epileptic EEG classification techniques. Several demonstrated limited accuracy and effectiveness. Therefore, some approaches do not work successfully with large amounts of data.

This study introduces an approach based on the InfoGain method incorporating either FFT or DWT techniques to extract features from EEG recordings. The LS-SVM classifier is applied to evaluate the performance of this proposed method. The structure of the proposed method is discussed in Sects. 2–4. The conclusion is presented in Sect. 5.

## 2 Data and methods

### 2.1 Epileptic EEG signals

The epileptic EEG data used in this paper were recorded by the Epileptology Department, Bonn University, Germany.

They are publicly available.<sup>1</sup> The database contains five classes of EEG signals (denoted classes  $z$ ,  $o$ ,  $n$ ,  $f$ , and  $s$ ), with each class containing 100 single channel EEG signals of 23.6 s. All classes of the EEG signals were made with the same 128-channel amplifier system, using a common reference. The signals were digitized at 173.61 samples per-second using 12-bit resolution. Band-pass filter settings were 0.53–40 Hz (12B/oct). Classes  $z$  and  $o$  were recorded from five healthy people with open and closed eyes, respectively, while classes  $n$ ,  $f$ , and  $s$  were collected from five epileptic patients. The data in classes  $n$  and  $f$  were collected through the free seizure from five epileptic patients. However, class  $s$  was taken from EEG records of five pre-surgical epileptic patients during epileptic seizure activity. Figure 1 shows examples of five EEG signals, classes  $z$ ,  $o$ ,  $n$ ,  $f$  and  $s$  (Al Ghayab et al. 2016).

### 2.2 Methods

This study aims to develop a technique for detecting epileptic seizures from EEG. Figure 2 shows the structure of the proposed method. Here the frequency domain is used by performing FFT or DWT based on InfoGain to extract and select the key features. The LS-SVM is used to classify the extracted and selected features. As illustrated in Fig. 2, the proposed approach is divided into five main parts: FFT and DWT (employed separately); bands segmentation; statistical features extraction; features ranking; and selection based on the InfoGain and classification part by the LS-SVM classifier. A brief explanation of these five parts is provided below.

#### 2.2.1 Fast Fourier transform

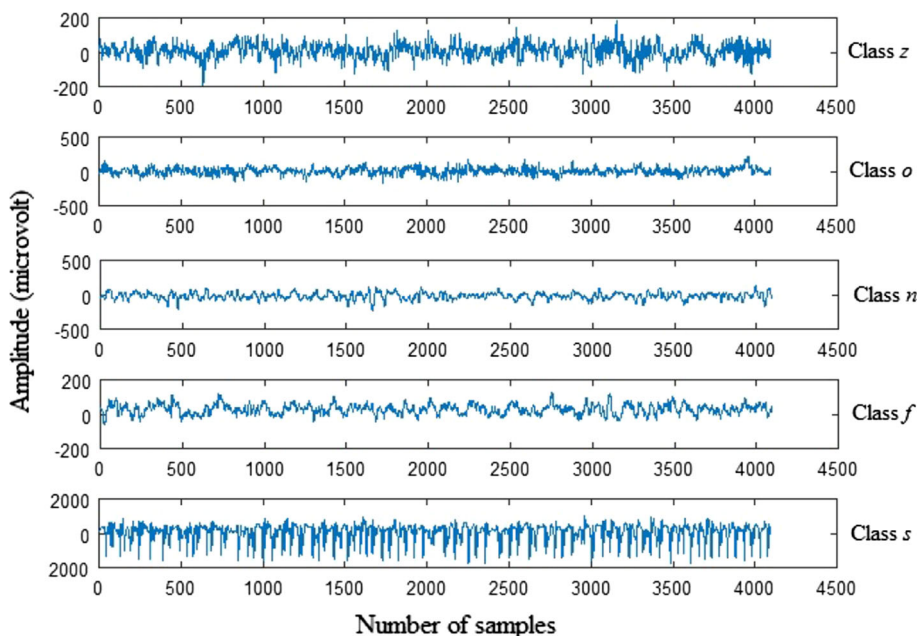
Fast Fourier transform (FFT) is an improved algorithm of discrete Fourier transform (DFT). It is used for the fast implementation of the DFT method (Heckbert 1995). According to Zonst (1995), the execution time of the DFT is greater than the execution time of the FFT. The FFT can achieve the same results much faster than that of DFT to analyse the EEG signals. In this study, we use the FFT algorithm to reduce the amount of computation required on EEG quantities data. The FFT splits EEG data into two parts and processes each part individually to extract five frequency bands.

#### 2.2.2 Discrete wavelet transform

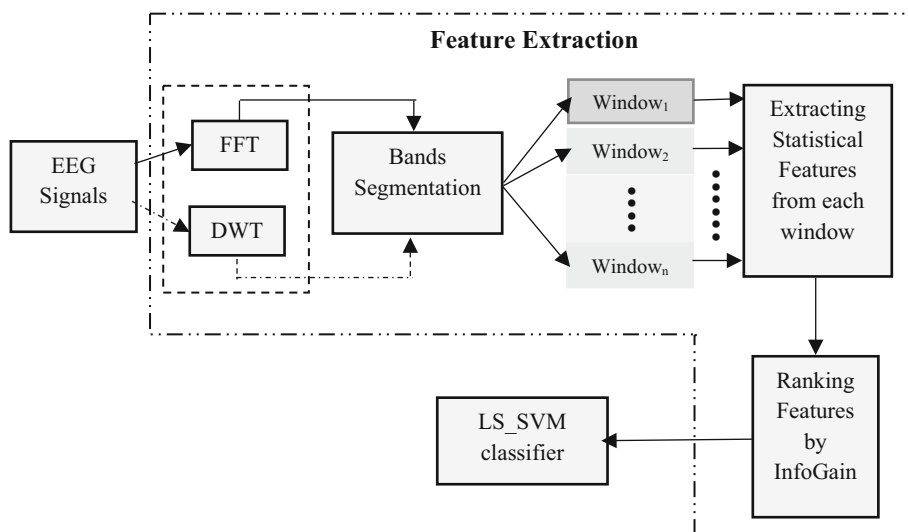
Discrete wavelet transform (DWT) transforms the EEG signals into the wavelet domain. In the DWT, the EEG signals are transformed into different levels of decomposition. In the first level, a signal is decomposed into a low pass filter (LPF)

<sup>1</sup> <http://www.meb.unibonn.de/epileptologie/science/physik/eegdata.html>.

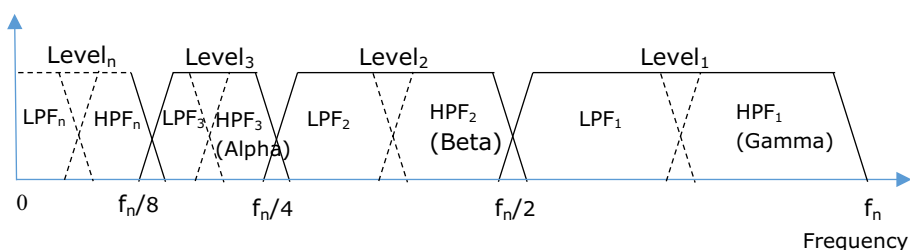
**Fig. 1** Examples of five different classes of EEG signals



**Fig. 2** The structure of the proposed system for the classification of epileptic EEG signals



**Fig. 3** Process of decomposition levels using the DWT



called ‘approximate coefficients’, and a high pass filter (HPF) called ‘detail coefficients’. This technique is used to transform the EEG signals into two different numerical vectors of the same length. In the next level of decomposition, the previous LPF is decomposed again into a HPF and a LPF. This operation is repeated to acquire the next level of DWT coef-

ficients. At the end of each level, the length of the LPF and the HPF are half of the length of the previous approximate coefficients (Acharya et al. (2015)). Figure 3 represents the process of the decomposition by using the DWT approach.

### 2.2.3 Bands segmentation

EEG recordings are non-stationary in frequency. Partitioning EEG signals into smaller segments or windows can make the EEG signals quasi-stationary. To extract the most discriminative statistical features from each band, firstly, we utilize and divide each band from the FFT or DWT into a number of small windows. The number of windows is determined empirically based on the following formula (1):

$$S(i) = \frac{B}{W}; \quad W = 1, 2, 3, 4 \quad (1)$$

where  $S(i)$  refers to the statistical features set for each window,  $i$  is the number of the statistical features,  $B$  is the band from the FFT or DWT,  $W$  is the number of windows.

Secondly, we extract the key statistical features from each window. All the extracted features from each window are gathered in one set, called *Feature\_FFT\_D<sub>n</sub>* or *Feature\_Wave\_D<sub>n</sub>*, depending on the FFT or DWT as appearing in formulas (2) and (3):

$$\begin{aligned} \text{Feature\_FFT\_D}_n &= \sum_{i=1}^W S(i); \\ (i &= 1, 2, \dots, W), (n = 1, 2, \dots, 5) \end{aligned} \quad (2)$$

$$\begin{aligned} \text{Feature\_Wave\_D}_n &= \sum_{i=1}^W S(i); \\ (i &= 1, 2, \dots, W), (n = 1, 2, \dots, 5) \end{aligned} \quad (3)$$

where *Feature\_FFT\_D<sub>n</sub>* is the set of the features from each band of the FFT, *Feature\_Wave\_D<sub>n</sub>*, is the features set from each band of the DWT,  $n$  is the number of bands, and  $W$  is the number of windows.

### 2.2.4 The statistical features

In this study, nine statistical features are extracted from each window to reduce the dimensions of each band. These statistical features are {*maximum, minimum, mean, median, mode, first quartile, second quartile, range and standard deviation*}. As the EEG signals provide important information for human behaviours, these statistical features ideally represent the important information (Siuly et al. 2015). A set of features are obtained from each band by the FFT or the DWT, having 4097 data points in 36 dimensions. These features are used in the next step of the proposed method.

### 2.2.5 Information gain

Information Gain (InfoGain) is widely applied in machine learning areas. This technique is important to optimize the feature sets used in classification. It is often used to reduce the

dimensionality of the key features by selecting the important data and removing the redundant data from the classification process (Deng et al. 2008; Koprinska 2010). In this study, the InfoGain is used to select the most important features from the huge amount of data provided by the EEG signals. It is based on the concept of entropy from information theory. The formula of the InfoGain is presented below (Han et al. 2011):

$$\text{Info}(X) = - \sum_{i=0}^n P_i \log_2(P_i) \quad (4)$$

where  $P_i$  is the prior probability for the  $i$ th discretized value of  $X$ . The entropy of  $X$  after observing another variable  $Y$  is then defined as

$$\text{Info}\left(\frac{X}{Y}\right) = - \sum_{j=1}^m \frac{|y_j|}{|X|} \times \sum_{i=0}^n P \frac{x_i}{y_i} \log_2\left(P \frac{x_i}{y_i}\right) \quad (5)$$

The InfoGain is defined as the difference between the original information and the new information after dividing by  $Y$ . It is given as

$$\text{InfoGain} = \text{Info}(X) - \text{Info}\left(\frac{X}{Y}\right) \quad (6)$$

In this study, to select the most important statistical features, each feature within the different bands of EEG signals is ranked based on an adaptive threshold in decreasing order by using the InfoGain method.

### 2.2.6 Least square support vector machines (LS-SVM)

The LS-SVM classifier is extensively used in the field of biomedical signal processing. It is updated by versions of the support vector machines (SVMs). SVMs employ supervised learning approaches that analyse information and understand patterns of data. The LS-SVMs were developed by Suykens and Vandewalle (1999). Thus, the LS-SVM classifier is applied for the classification of EEG recordings with a radial basis function kernel in this paper. The use of the LS-SVM classifier is to avoid the convex quadratic programming issue which appeared in the classical SVMs, as it uses a set of linear equations (Siuly and Li 2015). The LS-SVM has  $\gamma$  and  $\sigma^2$  parameters. These two parameters are selected empirically as shown in Sect. 3. In this paper, the classification is performed by the LS-SVMlab (version 1.8) toolbox in MATLAB (2011).<sup>2</sup>

<sup>2</sup> <http://www.esat.kuleuven.ac.be/sista/lssvmlab>.



### 3 Classification measurements

Different assessments are used in this study to evaluate the proposed technique. One of these method is a  $k$  fold cross-validation (Siuly and Li 2015; Al Ghayab et al. 2018). In order to assess the performance of this scheme, the  $k$  fold is employed to divide the datasets into  $k$  subgroups denoted as folds. In each implementation, one of these folds is used as a testing group and other folds ( $k - 1$ ) are used as the training groups. The number of the  $k$  fold used in this study is fourfold cross-validation, which is found to be the most suitable for this dataset.

In addition, three different statistical evaluation methods are applied to evaluate the performances of the proposed technique. One of these measurements is the accuracy or the recognition rate. It is the percentage of a class correctly classified by a classifier for the testing data. A second assessment measurement is sensitivity or recall. This is defined as the percentage of the positive set correctly identified. It refers to the rate of the true positives. The specificity is the proportion of the negative set correctly identified. It is the average of the true negatives as shown in formulas (7)–(9) (Al Ghayab et al. 2016, 2017):

$$\text{Accuracy} = \frac{\text{True positives} + \text{True negatives}}{\text{All samples}} \times 100 \quad (7)$$

$$\text{Sensitivity} = \frac{\text{Number of true positives}}{\text{Positive samples}} \times 100 \quad (8)$$

$$\text{Specificity} = \frac{\text{Number of true negatives}}{\text{Negative samples}} \times 100 \quad (9)$$

### 4 Experimental results and discussions

In this section, the epileptic EEG signals used in this paper include five classes, which are: healthy people with eyes open (class  $z$ ); healthy people with eyes closed (class  $o$ ); epileptic patients with two free seizure classes (class  $n$  and  $f$ ); and the epileptic patients through activity seizures (class  $s$ ) classes. The FFT and DWT approaches were utilized to analyse the five classes of epileptic EEG signals into different sub-bands based on the levels of decomposition. Here, we used five levels of decomposition for both the FFT and DWT techniques and obtained five different frequency bands, namely Gamma, Beta, Alpha, Theta, and Delta, respectively. With the purpose of extracting the most discriminative statistical features from the bands, each band was divided empirically into four windows (refer to Sect. 2.2.2 for details).

Nine statistical features were extracted from each window. A set of statistical features was gained from the segmentations of the bands. These features were processed by the InfoGain technique in the next step. The InfoGain technique ranked each feature in the statistical features set extracted

**Table 1** Reduced dimensionality of each band from FFT and DWT by using segmentation and InfoGain

Classes	Band	FFT+ InfoGain	DWT+ InfoGain
Class $z$	1st band (Gamma)	$1882 \times 36$	$2016 \times 36$
	2nd band (Beta)	$1767 \times 36$	$1934 \times 36$
	3rd band (Alpha)	$1909 \times 36$	$1942 \times 36$
	4th band (Theta)	$1964 \times 36$	$1881 \times 36$
	5th band (Delta)	$1934 \times 36$	$1832 \times 36$
Class $o$	1st band (Gamma)	$1845 \times 36$	$1913 \times 36$
	2nd band (Beta)	$1929 \times 36$	$1892 \times 36$
	3rd band (Alpha)	$1990 \times 36$	$1962 \times 36$
	4th band (Theta)	$1878 \times 36$	$2038 \times 36$
	5th band (Delta)	$2031 \times 36$	$1876 \times 36$
Class $n$	1st band (Gamma)	$1784 \times 36$	$2321 \times 36$
	2nd band (Beta)	$1805 \times 36$	$2234 \times 36$
	3rd band (Alpha)	$1972 \times 36$	$2124 \times 36$
	4th band (Theta)	$2081 \times 36$	$1837 \times 36$
	5th band (Delta)	$2088 \times 36$	$1641 \times 36$
Class $f$	1st band (Gamma)	$2943 \times 36$	$2307 \times 36$
	2nd band (Beta)	$2194 \times 36$	$2529 \times 36$
	3rd band (Alpha)	$2460 \times 36$	$2533 \times 36$
	4th band (Theta)	$2571 \times 36$	$2261 \times 36$
	5th band (Delta)	$2263 \times 36$	$2378 \times 36$
Class $s$	1st band (Gamma)	$2101 \times 36$	$2821 \times 36$
	2nd band (Beta)	$2000 \times 36$	$2799 \times 36$
	3rd band (Alpha)	$2161 \times 36$	$2321 \times 36$
	4th band (Theta)	$2496 \times 36$	$2044 \times 36$
	5th band (Delta)	$2752 \times 36$	$2060 \times 36$

from the previous step. The InfoGain selects the most important features from each band and omits the less important ones. The goal of this technique is to reduce the dimensionality of the extracted features. The comparison performance results are shown in Table 1. After reducing the dimensions of the statistical features by the InfoGain, several experiments were conducted with the epileptic EEG signals. In each experiment, we gathered two similar bands of two different classes as pairs. For instance, we grouped the extracted features from the Delta band of class  $z$  with the Delta band of class  $s$  as the first pair, and continuing, as summarized in Table 2.

To assess the performance of each pair, EEG data was divided into two groups of training and testing by applying the  $k$ -fold cross-validation technique. The training group was used to train a classifier. The testing group was used

**Table 2** The pairs gained from different classes with similar bands

Methods	Pairs	Gamma	Beta	Alpha	Theta	Delta
FFT	<i>z</i> versus <i>s</i>	3983 × 36	3767 × 36	4070 × 36	4460 × 36	4686 × 36
	<i>z</i> versus <i>o</i>	3727 × 36	3696 × 36	3962 × 36	3842 × 36	3965 × 36
	<i>z</i> versus <i>f</i>	4825 × 36	3961 × 36	4369 × 36	4535 × 36	4197 × 36
	<i>o</i> versus <i>n</i>	3629 × 36	3734 × 36	3973 × 36	3959 × 36	4119 × 36
	<i>o</i> versus <i>s</i>	3946 × 36	3929 × 36	4151 × 36	4374 × 36	4783 × 36
	<i>n</i> versus <i>s</i>	3885 × 36	3805 × 36	4133 × 36	4577 × 36	4840 × 36
	<i>f</i> versus <i>s</i>	5044 × 36	4194 × 36	4621 × 36	5067 × 36	5015 × 36
DWT	<i>z</i> versus <i>s</i>	4837 × 36	4733 × 36	4263 × 36	3925 × 36	3892 × 36
	<i>z</i> versus <i>o</i>	3929 × 36	3826 × 36	3904 × 36	3919 × 36	3708 × 36
	<i>z</i> versus <i>f</i>	4323 × 36	4463 × 36	4475 × 36	4142 × 36	4210 × 36
	<i>o</i> versus <i>n</i>	4234 × 36	4126 × 36	4086 × 36	3875 × 36	3517 × 36
	<i>o</i> versus <i>s</i>	4734 × 36	4691 × 36	4283 × 36	3920 × 36	3936 × 36
	<i>n</i> versus <i>s</i>	5142 × 36	5033 × 36	4445 × 36	3881 × 36	3701 × 36
	<i>f</i> versus <i>s</i>	5128 × 36	5328 × 36	4854 × 36	4305 × 36	4438 × 36

**Table 3** Experimental results for the classes of (*z* and *s*) pair in the epileptic EEG database with different groups of parameters,  $\gamma$  and  $\sigma^2$ 

Methods	Bands	Sensitivity (%)	Specificity (%)	Accuracy (%)	Parameter groups
FFT + InfoGain with LS-SVM	Gamma	97.19	100	98.60	$\gamma = 1, \sigma^2 = 10^5$
	Beta	100	100	100	$\gamma = 1, \sigma^2 = 10^3$
	Alpha	100	100	100	$\gamma = 1, \sigma^2 = 10^3$
	Theta	100	100	100	$\gamma = 1, \sigma^2 = 10^3$
	Delta	98.57	99.47	99.02	$\gamma = 1, \sigma^2 = 10^3$
Averages	N/A	99.05	99.99	99.52	N/A
DWT + InfoGain with LS-SVM	Gamma	34.16	99.34	66.78	$\gamma = 10^2, \sigma^2 = 10^3$
	Beta	27.99	96.63	62.34	$\gamma = 10^2, \sigma^2 = 10$
	Alpha	64.67	100	82.35	$\gamma = 1, \sigma^2 = 10$
	Theta	83.54	100	91.78	$\gamma = 1, \sigma^2 = 1$
	Delta	76.84	100	88.42	$\gamma = 1, \sigma^2 = 10^2$
Averages	N/A	57.44	99.19	78.33	N/A

to evaluate the performance of the classifier. Thereafter, the LS-SVM classifier was employed to categorize each dataset. The LS-SVM has two fundamental parameters, which are  $\gamma$  and  $\sigma^2$ . These parameters were chosen empirically to achieve the best results. The changing of these parameters affected the execution of the LS-SVM classifier. With the intention of gaining effective results, this study used various groups of parameters  $\gamma$  and  $\sigma^2$  to train the LS-SVM. We performed the proposed method with seven different pairs of the five epileptic EEG signals, which were classes *z* versus *o*, classes *z* versus *f*, classes *z* versus *s*, classes *o* versus *n*, classes *o* versus *s*, classes *n* versus *s*, and classes *f* versus *s* from the both FFT and DWT methods with different frequency bands as shown in Table 2.

The highest classification accuracies yielded 100% accuracy from the FFT and InfoGain (FFT+InfoGain) for the pairs of classes *z* versus *s*, classes *o* versus *s*, classes *n* versus *s*, classes *f* versus *s*, and classes *z* versus *o*, when  $\gamma = 1$  and  $\sigma^2 = 10^3$ , as shown in Tables 3, 4, 5, 6, 7, 8 and 9.

Tables 3, 4, 5, 6, 7, 8 and 9 illustrate the experimental results for the seven pairs of the epileptic EEG recordings in five different bands from the FFT and DWT combined with the InfoGain technique. By applying the FFT + InfoGain, a 100% classification accuracy was achieved with the following five pairs:

- Healthy people with eyes open (class *z*) versus the epileptic patients through activity seizures (class *s*).

**Table 4** Experimental results for the classes of (*o* and *s*) pair in the epileptic EEG database with different groups of parameters,  $\gamma$  and  $\sigma^2$ 

Methods	Bands	Sensitivity (%)	Specificity (%)	Accuracy (%)	Parameter groups
FFT + InfoGain with LS-SVM	Gamma	94.53	100	97.27	$\gamma = 1, \sigma^2 = 10^5$
	Beta	100	100	100	$\gamma = 1, \sigma^2 = 10^3$
	Alpha	100	100	100	$\gamma = 1, \sigma^2 = 10^3$
	Theta	100	100	100	$\gamma = 1, \sigma^2 = 10^3$
	Delta	99.66	98.28	98.97	$\gamma = 1, \sigma^2 = 10^4$
Averages	N/A	98.84	99.66	99.25	N/A
DWT + InfoGain with LS-SVM	Gamma	23.74	99.83	61.78	$\gamma = 10^4, \sigma^2 = 10^3$
	Beta	23.68	97.28	60.51	$\gamma = 10^2, \sigma^2 = 10$
	Alpha	66.23	98.32	82.29	$\gamma = 10^2, \sigma^2 = 10^2$
	Theta	82.82	100	91.42	$\gamma = 1, \sigma^2 = 1$
	Delta	81.50	100	90.75	$\gamma = 1, \sigma^2 = 10^2$
Averages	N/A	55.59	99.09	77.35	N/A

**Table 5** Experimental results for the classes of (*n* and *s*) pair in the epileptic EEG database with different groups of parameters,  $\gamma$  and  $\sigma^2$ 

Methods	Bands	Sensitivity (%)	Specificity (%)	Accuracy (%)	Parameter groups
FFT + InfoGain with LS-SVM	Gamma	92.66	100	96.34	$\gamma = 1, \sigma^2 = 10^3$
	Beta	100	100	100	$\gamma = 1, \sigma^2 = 10^3$
	Alpha	100	100	100	$\gamma = 1, \sigma^2 = 10^3$
	Theta	100	100	100	$\gamma = 1, \sigma^2 = 10^3$
	Delta	100	100	100	$\gamma = 1, \sigma^2 = 10^3$
Averages	N/A	98.53	100	99.27	N/A
DWT + InfoGain with LS-SVM	Gamma	60.78	100	80.39	$\gamma = 10^2, \sigma^2 = 10^2$
	Beta	55.80	94.92	75.38	$\gamma = 10^2, \sigma^2 = 10$
	Alpha	82.41	100	91.21	$\gamma = 10, \sigma^2 = 1$
	Theta	78.76	100	89.39	$\gamma = 1, \sigma^2 = 1$
	Delta	55.17	100	77.61	$\gamma = 1, \sigma^2 = 1$
Averages	N/A	66.58	98.98	82.80	N/A

- Healthy people with eyes closed (class *o*) versus class *s*.
- Epileptic patients with free seizures (class *n*) versus class *s*.
- Another group of epileptic patients with free seizures (class *f*) versus class *s*.
- Classes *z* versus *o*.

Epileptic seizures are normally shown in the EEGs with high amplitudes. There are significant differences between healthy EEG signals such as classes *z*, *o* and *n*, and unhealthy EEG signals, such as class *s*. The differences make it easier to classify them as demonstrated by a 100% classification

accuracy by implementing the FFT with Beta, Alpha, Theta, and Delta frequency bands based on InfoGain on five pairs.

For the pairs of *o* versus *n* and *z* versus *f*, a 95.62 and 88.32% accuracy was obtained, respectively, for Gamma and Beta bands, when  $\gamma = 1$  and  $\sigma^2 = 10^2$ . The two groups were more analogous to each other, which led to the lowest classification accuracy compared with the classification results for other epileptic EEG pairs.

The highest accuracy (100%) was achieved by using DWT and InfoGain (DWT+InfoGain) on pair *f* versus *s*, when the parameters  $\gamma$  and  $\sigma^2$  ( $10^2, 10^2$ ). Also, with the DWT+InfoGain approach, the results obtained from the remain-

**Table 6** Experimental results for the classes of ( $f$  and  $s$ ) pair in the epileptic EEG database with different groups of parameters,  $\gamma$  and  $\sigma^2$ 

Methods	Bands	Sensitivity (%)	Specificity (%)	Accuracy (%)	Parameter groups
FFT + InfoGain with LS-SVM	Gamma	97.36	100	98.68	$\gamma = 1, \sigma^2 = 10^5$
	Beta	100	100	100	$\gamma = 1, \sigma^2 = 10^3$
	Alpha	100	100	100	$\gamma = 1, \sigma^2 = 10^3$
	Theta	100	100	100	$\gamma = 1, \sigma^2 = 10^3$
	Delta	98.17	100	99.08	$\gamma = 10^4, \sigma^2 = 10^3$
Averages	N/A	99.11	100	99.55	N/A
DWT + InfoGain with LS-SVM	Gamma	60.69	99.22	79.95	$\gamma = 10^4, \sigma^2 = 10^2$
	Beta	78.83	94.29	86.56	$\gamma = 10^2, \sigma^2 = 10$
	Alpha	100	100	100	$\gamma = 10^2, \sigma^2 = 10^2$
	Theta	100	98.70	99.35	$\gamma = 10, \sigma^2 = 10^2$
	Delta	100	98.74	99.37	$\gamma = 10^3, \sigma^2 = 10^3$
Averages	N/A	87.90	98.19	93.05	N/A

**Table 7** Experimental results for the classes of ( $o$  and  $n$ ) pair in the epileptic EEG database with different groups of parameters,  $\gamma$  and  $\sigma^2$ 

Methods	Bands	Sensitivity (%)	Specificity (%)	Accuracy (%)	Parameter groups
FFT + InfoGain with LS-SVM	Gamma	100	91.26	95.62	$\gamma = 1, \sigma^2 = 10^2$
	Beta	65.09	73.98	69.54	$\gamma = 10^3, \sigma^2 = 10^3$
	Alpha	82.93	100	91.47	$\gamma = 1, \sigma^2 = 10$
	Theta	67.68	100	83.86	$\gamma = 1, \sigma^2 = 10^2$
	Delta	84.34	98.18	91.26	$\gamma = 10, \sigma^2 = 10$
Averages	N/A	80.01	92.68	86.35	N/A
DWT + InfoGain with LS-SVM	Gamma	61.32	97.55	79.43	$\gamma = 10^4, \sigma^2 = 10^3$
	Beta	69.69	95.37	82.53	$\gamma = 10, \sigma^2 = 10$
	Alpha	82.85	93.37	88.11	$\gamma = 10, \sigma^2 = 10$
	Theta	70.72	91.56	81.15	$\gamma = 10, \sigma^2 = 10^2$
	Delta	84.35	83.94	84.14	$\gamma = 1, \sigma^2 = 10$
Averages	N/A	73.79	92.36	83.07	N/A

ing six pairs were 91.78, 91.42, 91.21, 88.11, 80.98 and 99.69% classification accuracies, respectively. The results were achieved using various groupings of the parameters  $\gamma$  and  $\sigma^2$ , which were (1, 1), (1, 1), (10, 1), (10, 10), (10<sup>2</sup>, 10<sup>2</sup>) and (10<sup>2</sup>, 10<sup>3</sup>), respectively, as provided in Tables 3, 4, 5, 6, 7, 8 and 9.

Additionally, this study compared the experimental results and time complexity by the FFT and DWT. Even though the FFT approach took 9.07 s to decompose the epileptic EEG signals, this method achieved 100% classification accuracy with five different groups by using the LS-SVM classifier. The DWT tackled the same groups in 1.16 s and yielded the

**Table 8** Experimental results for the classes of ( $z$  and  $f$ ) pair in the epileptic EEG database with different groups of parameters,  $\gamma$  and  $\sigma^2$ 

Methods	Bands	Sensitivity (%)	Specificity (%)	Accuracy (%)	Parameter groups
FFT + InfoGain with LS-SVM	Gamma	59.89	96.30	78.10	$\gamma = 1, \sigma^2 = 10^3$
	Beta	81.05	95.58	88.32	$\gamma = 1, \sigma^2 = 10^2$
	Alpha	57.55	98.87	78.23	$\gamma = 10, \sigma^2 = 10^2$
	Theta	52.32	99.47	75.91	$\gamma = 10^2, \sigma^2 = 10^2$
	Delta	64.00	97.14	80.57	$\gamma = 10, \sigma^2 = 10$
Averages	N/A	62.96	97.47	80.23	N/A
DWT + InfoGain with LS-SVM	Gamma	73.20	88.75	80.98	$\gamma = 10^2, \sigma^2 = 10^2$
	Beta	48.93	88.77	68.87	$\gamma = 10, \sigma^2 = 10$
	Alpha	48.93	88.95	68.96	$\gamma = 10, \sigma^2 = 10^2$
	Theta	67.69	75.77	71.73	$\gamma = 10^2, \sigma^2 = 10^4$
	Delta	51.04	93.74	72.39	$\gamma = 10, \sigma^2 = 10$
Averages	N/A	57.96	87.20	72.59	N/A

**Table 9** Experimental results for the classes of ( $z$  and  $o$ ) pair in the epileptic EEG database with different groups of parameters,  $\gamma$  and  $\sigma^2$ 

Methods	Bands	Sensitivity (%)	Specificity (%)	Accuracy (%)	Parameter groups
FFT + InfoGain with LS-SVM	Gamma	61.21	81.72	71.47	$\gamma = 1, \sigma^2 = 10^3$
	Beta	90.50	99.59	95.05	$\gamma = 1, \sigma^2 = 10^3$
	Alpha	100	100	100	$\gamma = 1, \sigma^2 = 10^3$
	Theta	79.59	94.70	87.16	$\gamma = 1, \sigma^2 = 10^2$
	Delta	99.43	100.00	99.71	$\gamma = 10, \sigma^2 = 10^3$
Averages	N/A	86.15	95.20	90.68	N/A
DWT + InfoGain with LS-SVM	Gamma	100	99.39	99.69	$\gamma = 10^2, \sigma^2 = 10^3$
	Beta	82.05	83.92	82.99	$\gamma = 10, \sigma^2 = 10$
	Alpha	77.66	88.11	82.89	$\gamma = 0.5, \sigma^2 = 10$
	Theta	75.41	74.24	74.82	$\gamma = 1, \sigma^2 = 10^2$
	Delta	77.42	69.25	73.33	$\gamma = 10, \sigma^2 = 10^2$
Averages	N/A	82.51	82.98	82.74	N/A

lowest accuracies by utilizing the same classifier as presented in Table 10.

The comparison of several approaches was listed in terms of the accuracy for the epileptic EEG signals classification as shown in Table 11. In this table, the performance results of the proposed method were compared with the six exist-

ing methods (Nicolaou and Georgiou 2012; Zhu et al. 2013; Shen et al. 2013; Wang et al. 2014; Kaya et al. 2014; Samiee et al. 2015) by using the same datasets. We highlighted in bold font the highest overall classification accuracy between the techniques. It can certainly be seen from the results in Table 11 that the results yielded from the FFT based on

**Table 10** Comparison of the results and time complexity for the FFT and DWT approaches

Different pairs	Bands	The accuracy of the FFT + InfoGain	The accuracy of the DWT + InfoGain
<i>z</i> versus <i>s</i>	Theta	100%	91.78%
<i>o</i> versus <i>s</i>	Theta	100%	91.42%
<i>n</i> versus <i>s</i>	Alpha	100%	91.21%
<i>f</i> versus <i>s</i>	Alpha	100%	100%
<i>o</i> versus <i>n</i>	Gamma and Alpha, respectively	95.62%	88.11%
<i>z</i> versus <i>f</i>	Beta and Gamma, respectively	88.32%	80.98%
<i>z</i> versus <i>o</i>	Alpha and Gamma, respectively	100%	99.69%
Time (s)	N/A	9.07 s	1.16 s

**Table 11** Comparison with the existing techniques with the epileptic EEG data

Authors	Methods	Classification test (Classes)	Overall accuracy (%)
Al Ghayab et al. (the proposed method)	FFT + InfoGain + LS-SVM	<i>z</i> versus <i>s</i> , <i>o</i> versus <i>s</i> , <i>n</i> versus <i>s</i> , <i>f</i> versus <i>s</i> , and <i>z</i> versus <i>o</i>	100.0
	DWT + InfoGain + LS-SVM	<i>z</i> versus <i>s</i> , <i>o</i> versus <i>s</i> , <i>n</i> versus <i>s</i> , <i>f</i> versus <i>s</i> , and <i>z</i> versus <i>o</i>	94.8
Samiee et al. (2015)	Rational discrete short-time Fourier transform + multilayer perceptron	<i>z</i> versus <i>s</i> , <i>o</i> versus <i>s</i> and <i>n</i> versus <i>s</i>	99.2
Kaya et al. (2014)	One-dimensional local binary (non-uniform) + functional tree	<i>z</i> versus <i>s</i>	99.5
	One-dimensional local binary (uniform) + functional tree	<i>z</i> versus <i>s</i>	99.0
Wang et al. (2014)	Simple random sampling + J48 decision tree	<i>z</i> versus <i>s</i> , <i>o</i> versus <i>s</i> and <i>n</i> versus <i>s</i>	97.6
Shen et al. (2013)	Wavelet + approximate entropy + SVM	<i>z</i> versus <i>s</i> , and <i>o</i> versus <i>s</i>	99.97
	Wavelet + approximate entropy + <i>k</i> -nearest neighbours	<i>z</i> versus <i>s</i> , and <i>o</i> versus <i>s</i>	99.1
	Wavelet + approximate entropy + radial basis function neural network	<i>z</i> versus <i>s</i> , and <i>o</i> versus <i>s</i>	99.8
Zhu et al. (2013)	Sample entropy + multi-scale <i>K</i> -means clustering	<i>z</i> versus <i>s</i> , and <i>o</i> versus <i>s</i>	99.5
Nicolaou and Georgiou (2012)	Permutation entropy + SVM	<i>z</i> versus <i>s</i> , <i>o</i> versus <i>s</i> and <i>n</i> versus <i>s</i>	88.1

the InfoGain and LS-SVM were the best compared with the recent reported methods. The results show that the proposed method enhanced the overall classification accuracy up to 11.9% more than the existing approaches for the classification of the epileptic EEG recordings with the same datasets.

## 5 Conclusion

This research presents a new method to detect the epileptic seizures from EEG signals. Two frequency domain techniques FFT and DWT were used individually to investigate which method was the most suitable for EEG feature extrac-

tion. Thereafter, the extracted features were utilized as the input to the information gain algorithm (InfoGain) for ranking and selecting the most discriminative statistical features. The LS-SVM with a RBF kernel function was used for classifying two classes of EEG recordings. In this study, different pairs of two class epileptic EEG signals were tested. The FFT combined with InfoGain yielded a 100% average classification accuracy for five different pairs of epileptic EEG data, which were: healthy people with eyes open (class  $z$ ) versus the epileptic patients through activity seizures (class  $s$ ); healthy people with eyes closed (class  $o$ ) versus class  $s$ ; epileptic patients with free seizures (class  $n$ ) versus class  $s$ ; patients with free seizures epileptic (class  $f$ ) versus class  $s$ ; and class  $z$  versus  $o$ . However, for two other groups of ( $o$  vs.  $n$ ) and ( $z$  vs.  $f$ ), the classification accuracies were 95.62 and 88.32%, respectively, because these two pairs were more similar to each other.

Overall, the results demonstrate that the proposed approach was promising for the extraction of the features from the raw EEG data and the selection of the most important features. Moreover, our method was compared with six existing approaches and showed that the proposed technique outperformed the six compared methods. This method was capable of detecting EEG epileptic seizures with the highest recognition rate. The proposed method will also be useful for analysing and classifying other biomedical data.

**Acknowledgements** The first author acknowledges the Iraqi government (Ministry of Higher Education and scientific research) for providing PhD scholarship.

## Compliance with ethical standards

**Conflict of interest** The authors declare that they have no conflict of interest.

**Human and animal rights** This article does not contain any studies with human participants or animals performed by any of the authors.

## References

- Acharya UR, Vidya KS, Ghista DN et al (2015) Computer-aided diagnosis of diabetic subjects by heart rate variability signals using discrete wavelet transform method. *Knowl Based Syst* 81:56–64
- Al Ghayab HR, Li Y, Abdulla S et al (2016) Classification of epileptic EEG signals based on simple random sampling and sequential feature selection. *Brain Inform* 3(2):85–91
- Al Ghayab HR, Li Y, Siuly S et al (2017) Developing a tunable Q-factor wavelet transform based algorithm for epileptic EEG feature extraction. In: *International conference on health information science*. Springer, Cham, pp 45–55
- Al Ghayab HR, Li Y, Siuly S et al (2018) Epileptic EEG signal classification using optimum allocation based power spectral density estimation. *IET Signal Process* 12(6):738–747
- Amin HU, Malik AS, Ahmad RF et al (2015) Feature extraction and classification for EEG signals using wavelet transform and machine learning techniques. *Australas Phys Eng Sci Med* 38:139–149
- Chua KC, Chandran V, Acharya UR et al (2011) Application of higher order spectra to identify epileptic EEG. *J Med Syst* 35:1563–1571
- Deng JD, Simmermacher C, Cranefield S (2008) A study on feature analysis for musical instrument classification. *IEEE Trans Syst Man Cybern B Cybern* 38:429–438
- Gajic D, Djurovic Z, Di Gennaro S et al (2014) Classification of EEG signals for detection of epileptic seizures based on wavelets and statistical pattern recognition. *Biomed Eng Appl Basis Commun* 26:1450021
- Gajic D, Djurovic Z, Gligorijevic J et al (2015) Detection of epileptiform activity in EEG signals based on time-frequency and non-linear analysis. *Front Comput Neurosci* 9:38
- Han J, Kamber M, Pei J (2011) *Data mining: concepts and techniques*, 3rd edn. Elsevier, Amsterdam
- Heckbert P (1995) Fourier transforms and the fast Fourier transform (FFT) algorithm. *Comput Graph* 2:15–463
- Kaya Y, Uyar M, Tekin R et al (2014) 1D-local binary pattern based feature extraction for classification of epileptic EEG signals. *Appl Math Comput* 243:209–219
- Koprinska I (2010) Feature selection for brain-computer interfaces. In: Theeramunkong T et al (eds) *New Frontiers in Applied Data Mining, PAKDD 2009*. Lecture Notes in Computer Science, vol 5669. Springer, Berlin, Heidelberg, pp 106–117
- LS-SVMlab toolbox (version 1.8) (2011) <http://www.esat.kuleuven.ac.be/sista/lssvmlab/>. Accessed Nov 2016
- Mcgrogan N (1999) *Neural network detection of epileptic seizures in the electroencephalogram*. Dissertation, University of Oxford
- Nicolaou N, Georgiou J (2012) Detection of epileptic electroencephalogram based on permutation entropy and support vector machines. *Expert Syst Appl* 39:202–209
- Rao T, Vishwanath DD (2014) Detecting sleep disorders based on EEG signals by using discrete wavelet transform. In: *2014 International conference on green computing communication and electrical engineering (ICGCCEE)*. IEEE, pp 1–5
- Samiee K, Kovacs P, Gabbouj M (2015) Epileptic seizure classification of eeg time-series using rational discrete short-time fourier transform. *IEEE Trans Biomed Eng* 62:541–552
- Shen C-P, Chen C-C, Hsieh S-L et al (2013) High-performance seizure detection system using a wavelet-approximate entropy-fSVM cascade with clinical validation. *Clin EEG Neurosci* 44:247–256
- Siuly S, Li Y (2015) Designing a robust feature extraction method based on optimum allocation and principal component analysis for epileptic EEG signal classification. *Comput Methods Program Biomed* 119(1):29–42
- Siuly S, Kabir E, Wang H et al (2015) Exploring sampling in the detection of multicategory EEG signals. *Comput Math Methods Med* 2015:576437
- Suykens JA, Vandewalle J (1999) Least squares support vector machine classifiers. *Neural Process Lett* 9(3):293–300
- Swami P, Gandhi TK, Panigrahi BK et al (2016) A comparative account of modelling seizure detection system using wavelet techniques. *Int J Syst Sci Oper Logist* 4:41–52
- Tzamourta KD, Tzallas AT, Giannakeas N, Astrakas LG, Tsalikakis DG, Tspouras MG (2018) Epileptic seizures classification based on long-term EEG signal wavelet analysis. In: Maglaveras N, Chouvarda I, de Carvalho P (eds) *Precision Medicine Powered by pHealth and Connected Health*. IFMBE Proceedings, vol 66. Springer, Singapore, pp 165–169
- Wali MK, Murugappan M, Ahmmad B (2013) Wavelet packet transform based driver distraction level classification using EEG. *Math Probl Eng* 2013:10

- Wang S, Zhu G, Li Y et al (2014) Analysis of epileptic EEG signals with simple random sampling J48 algorithm. *Int J Biosci Biochem Bioinform* 4:78
- World Health Organization (WHO) (2011) Report: WHO. <http://www.who.int/mediacentre/factsheets/fs999/en/index.html>. Accessed Dec 2015
- Yang Y, Pedersen JO (1997) A comparative study on feature selection in text categorization. In: ICML, pp 412–420
- Yuan Q, Zhou W, Li S et al (2011) Epileptic EEG classification based on extreme learning machine and nonlinear features. *Epilepsy Res* 96:29–38
- Zhu G, Li Y, Wen PP et al (2013) Unsupervised classification of epileptic EEG signals with multi scale K-means algorithm. In: *Brain and health informatics*. Springer, Berlin, p 158–167
- Zonst AE (1995) *Understanding the FFT: a tutorial on the algorithm & software for laymen, students, technicians & working engineers*. Citrus Press, Florida

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations



## **5.2 Chapter summary**

Al Ghayab et al. (2018) developed a new method to detect the epileptic seizures from EEG signals. They used two frequency domain techniques FFT and DWT individually to extract features from the epileptic EEG signals. The extracted features from FFT or DWT were transmitted to the InfoGain algorithm for ranking and selecting the most discriminative statistical features. The LS\_SVM with an RBF kernel function was used for classifying EEG recordings. The proposed technique tested seven cases of epileptic EEG signals. This study proved that the FFT combined with InfoGain achieved higher accuracy than DWT and InfoGain.

Al Ghayab et al. (2018) demonstrated that the proposed approach was promising for the extraction of the features from the EEG data and the selection of the most important features. Moreover, this method was compared with recent studies and showed that the proposed technique outperformed those methods. Al Ghayab et al. (2018) clearly demonstrate that the proposed method is capable of detecting EEG epileptic seizures with the highest recognition rate.

This page is intentionally left blank

# 6

## CHAPTER 6

### CONCLUSIONS AND DIRECTIONS FOR FUTURE WORK

EEG signals are affected by brain diseases, such as epileptic seizures, sleep problems, Alzheimer, etc. Analysis and classification methods of EEG signals are used to detect and identify brain disorders. These methods are grouped into five domains: time domain, frequency domain, time-frequency domain, traditional nonlinear method, and graph theory. Practical and effective approaches are therefore in demand to achieve accurate results for EEG classification. This chapter summarises the analysis and classification techniques presented in this thesis, and also some work limitations are discussed for future research guidance.

#### 6.1 Summary and Conclusions of the Thesis

In this research, EEG analysis and classification methods are developed in four domains to identify different categorises of EEG recordings, with four new techniques:

1. To develop a new technique to analyse and classify epileptic EEG signals based on the simple random sampling (SRS) and sequential feature selection (SFS). The SRS\_SFS was utilized to reduce the redundant EEG epileptic data and also minimized the execution time as shown in Table 6.1. This method achieved a high classification accuracy compared with other techniques, as seen in Chapter 2.
2. To develop a novel technique to classify the EEG multi-channels by blending optimum allocation (OA) technique based power spectral density estimation

methods. The proposed OA\_AR technique was implemented to extract the best features from big quantities of EEG signals with a small execution time. This scheme yielded a 100% of classification accuracy. The comparisons were made between the proposed scheme and the existing approaches. These comparisons show that the proposed technique outperforms other methods as seen in Chapter 3.

3. To introduce an efficient feature extraction method from different EEG databases based on a Tunable Q-factor wavelet transform for EEG signal classification. In this technique, two different databases were used to validate the performances. Additionally, the comparison with other methods was conducted to evaluate the effectiveness of the proposed method as seen in Chapter 4. As a result, the method has a great potential to extract representative features from the EEG recordings.
4. To detect the epileptic seizures in EEGs by developing a frequency domain and information gain method. This approach took into consideration that the reduction of dimensions of the data used and consumption time are paramount for best EEG classification. While, this method achieved a high classification accuracy. To evaluate the performance of this scheme, the comparisons were made between the proposed technique and other approaches, which used the same database and the same EEG channels as discussed in Chapter 5.

These four techniques can help doctors and other health experts to analyze EEG signals and diagnose the different brain disorders effectively. In addition, the proposed methods made contributions by providing significant information to the researchers in this filed.

In order to accomplish these techniques, a simple random sampling (SRS) method, combined with sequential feature selection (SFS) was built for feature extraction and selection in the time domain as seen in Chapter 2. Firstly, an SRS technique was applied to extract the most representative features from the EEG data. To extract a set of features, EEG recording was divided into two levels. In the first level, ten samples were selected randomly from each set of EEGs. In the next level, five subsamples were selected randomly from each first phase sample. In order to determine a sample size for each level, a sample size calculator (Creative Research System) was used. Ultimately, nine statistical features were extracted from each subsample and located in one set. Secondly, the SFS algorithm was employed to select

the most discriminating feature and to reduce the dimensionality of EEG data. The SFS was utilized to select fewer numbers of uncorrelated variables were used as the input of a classifier for better EEG classification. Finally, a least square support vector machine (LS\_SVM) classifier was applied to classify the extracted features. The SRS and SFS method was implemented on the EEG epileptic data to differentiate between the EEG categories. The results show that this method achieved 99.9% classification accuracy.

To improve the classification performance, an optimum allocation (OA) technique based power spectral density estimation method was developed for epileptic EEG classification in the time-frequency domain as discussed in Chapter 3. In this technique, the OA method to determine representative sample points was introduced in two stages from the EEG data. In the first stage, a sample size was determined for each EEG class by using a sample size calculator. In the next stage, the OA technique was employed to partition each EEG class and then choose the sample points from each class of epileptic EEG data depending on sample size determination from the previous stage. In order to reduce the dimensionality of the chosen sample points and to extract the most relevant features from each OA sample group, two power spectral density estimation were used methods: periodogram and autoregressive. The extracted feature sets from this technique were fed to three popular machine learning classifiers: support vector machine (SVM), quadratic discriminant analysis (QDA) and k-nearest neighbour (k-NN). Furthermore, several output coding schemes were used for the SVM classifier, which included one vs one (1vs1), all pairs (AP), one vs all (1vsA), binary complete (BC), ternary complete (TC), ordinal (OR), sparse random (SR), and dense random (DR), were tested. In addition, this technique was implemented on a benchmark epileptic EEG database for classification and evaluation. The experimental results showed that the OA\_AR method yielded 100% overall accuracy with SVM for all epileptic EEG sets.

For the nonlinear methods, a Tunable Q-factor wavelet transform (TQWT) and statistical approach was developed for EEG signal analysis and classification as explained in Chapter 4. In this framework, the multi-channel EEG signals were decomposed into a number of sub-bands using the TQWT method. The TQWT method was parameterized by its Q factor (Q), redundancy (R) and decomposition level ( $j^{\text{th}}$ ). This method also depends on the resonance of the EEG signals, rather than on

frequency or scaling as in other wavelet transform methods. Thereafter, a statistical method was applied on the sub-bands. The statistical method was used to divide each sub-band into  $n$  windows, and then extracted ten statistical features from each window. Lastly, the extracted features were used as the input to a bagging tree,  $k$ -NN and SVM as classifiers to evaluate the performance of this technique. Further, this technique was conducted on two different EEG databases. The results presented indicated that the TQWT with  $k$ -NN classifier achieved 100% average classification accuracy for two databases.

In the frequency domain, a new algorithm that combines the methods from the frequency domain with an information Gain (InfoGain) technique was presented to detect the epileptic seizures in EEGs as shown in Chapter 5. Firstly, two frequency domain methods, namely fast Fourier transform (FFT) and discrete wavelet transform (DWT), were applied to decomposed EEG recording signals. In order to extract the representative features, each frequency band was segmented into a number of segmentations and key statistical features were extracted from each segment. Hereinafter, the InfoGain was employed to rank the extracted features and then to select the most important features. Finally, to evaluate the performance of this algorithm, the LS-SVM classifier was used. The experimental results demonstrated that the FFT + InfoGain obtained 100% average accuracy with some epileptic EEG cases, which were tested on a benchmark EEG database.

Table 6.1 Total of the dimension reduction and the time consuming from each studies and the whole database.

Method \ Dataset	Raw Datasets	SRS_SFS with LS_SVM	OA_AR with SVM	TQWT with $k$ -NN	FFT_InfoGain with LS_SVM
Set (A)	4097×100	1000×35	512×100	1024×20	1909×36
Set (B)	4097×100	--	512×100	1024×20	1990×36
Set (C)	4097×100	--	512×100	1024×20	1972×36
Set (D)	4097×100	--	512×100	1024×20	2460×36
Set (E)	4097×100	1000×35	512×100	1024×20	2161×36
Total Dimension	20,485×100	2000×35	2580×100	5120×20	10,492×36
Execution time	25.93s	0.16s	3.08s	0.28s	1.16s

It was found that the techniques could reduce the implementation time, as can be seen in Table 6.1. Table 6.1 shows a comparison for the four proposed techniques in terms of data reduction and time consuming. As in Table 6.1 the data points to the algorithms have been reduced through the four proposed techniques as well as the execution time. This table demonstrates that the proposed approaches have a good potential to extract the most significant features. These extracted features from the developed methods led to decrease the execution time to classify the EEG signals.

In terms of accuracy, I have compared the proposed techniques with the recent methods as can be seen in each chapter in this thesis. I found that, the developed techniques outperformed the state-of-the-art works with a 14.1% improvement. The proposed approaches will assist experts to automatically analyse a large volume of EEG data and benefit epilepsy research.

To sum up, it can be concluded that this research project has established new and successful algorithms and techniques for reliable classification of epileptic EEG recordings in the biomedical field. These approaches will assist experts to detect and diagnose brain related disorders and the outcomes will help epileptic patients to improve the quality of their lives.

### **6.2 Future Work**

This study develops analysis and feature extraction techniques from EEG recordings for classification in the following different domains: time domain; frequency domain; time-frequency domain; and nonlinear method. It is believed that the techniques discussed in this thesis will provide great potential to analyse EEG signals in the biomedical field. To enable additional improvements to the methods presented in this thesis, a few key issues are highlighted which are addressed below.

In regards to the SRS and SFS technique, it was tested on two classes of EEG signals: health class (A); vs patient during epileptic seizures class (E). Technically, the SRS method was applied to divide the EEG signals into two levels (samples and subsamples) through the distribution of population. In future, the SRS method could be used to segment the EEG data into three or more levels. At the last subsamples, more than nine statistical features with entropies features will extract and study the characteristics of each feature to find the harmonize

among these features. The SFS algorithm will test these features and select the most proportional features from all EEG classes. This study will extract the most representative features of raw EEG data.

With regard to the OA\_AR method, this was developed to extract features from the time-frequency domain from an offline database collected by Bonn University, Germany. This approach works well with this database, but it may increase the delay time when implemented with a real time application. In the future, the OA\_AR scheme might be executed on a real time database by using a big data framework.

In addition, the TQWT based feature extraction method was developed to analyse and classify two different epileptic EEG databases. The TQWT is a parametric method that depends on Q factor (Q) and redundancy (R), which were set empirically, to decompose EEG signals into a number of sub-bands. In the future, the technique could be used to investigate the relation between two parameters (Q and R) to obtain the best decomposition level and to extract key features from each level.

Finally, two frequency domain methods (FFT and DWT) with InfoGain were tested using offline Epileptic EEG signals. This method was used to classify binary EEG classes. This scheme can be extended in the future to test multi-channel EEG signals, and also to be implemented on real time databases.

Additionally, the raw EEG recordings contain a variety of noises and artifacts, both from the subject and from equipment interferences. In this study, we did not develop techniques which might deal with removing noise and artifacts from raw EEG data. Further study is needed to successfully remove noise without composing EEG signals for the proposed algorithms. In the future, these algorithms would be developed so that there could be a significant improvement in signal classification, after removal of these kind of unrelated signals.

In summary, these proposed techniques can successfully classify EEG signals and efficiently get accurate results. However, room for improvement still exists and therefore, further study needs to be undertaken in the future.



This page is intentionally left blank



## REFERENCES

- Acharya, U. R., Sree, S. V., Swapna, G., Martis, R. J., & Suri, J. S. (2013). Automated EEG analysis of epilepsy: a review. *Knowledge-Based Systems, 45*, 147-165.
- Acharya, U. R., Sree, S. V., Alvin, A. P. C., & Suri, J. S. (2012). Use of principal component analysis for automatic classification of epileptic EEG activities in wavelet framework. *Expert Systems with Applications, 39*(10), 9072-9078.
- Altunay, S., Telatar, Z., & Eroglu, O. (2010). Epileptic EEG detection using the linear prediction error energy. *Expert Systems with Applications, 37*(8), 5661-5665.
- Ackerman, S. (1992). *Discovering the brain*: National Academies Press.
- Arunkumar, N., Kumar, K. R., & Venkataraman, V. (2016). Automatic detection of epileptic seizures using new entropy measures. *Journal of Medical Imaging and Health Informatics, 6*(3), 724-730.
- Abhang, P. A., & Gawali, B. W. (2015). Correlation of EEG images and speech signals for emotion analysis. *British Journal of Applied Science & Technology, 10*(5), 1-13.
- Amin, H. U., Malik, A. S., Ahmad, R. F., Badruddin, N., Kamel, N., Hussain, M., & Chooi, W. T. (2015). Feature extraction and classification for EEG signals using wavelet transform and machine learning techniques. *Australasian physical & engineering sciences in medicine, 38*(1), 139-149.
- Acharya, U. R., Sree, S. V., Chattopadhyay, S., Yu, W., & Ang, P. C. A. (2011). Application of recurrence quantification analysis for the automated identification of epileptic EEG signals. *International journal of neural systems, 21*(03), 199-211.
- Acharya, U. R., Molinari, F., Sree, S. V., Chattopadhyay, S., Ng, K. H., & Suri, J. S. (2012). Automated diagnosis of epileptic EEG using entropies. *Biomedical Signal Processing and Control, 7*(4), 401-408.
- Artameeyanant, P., Sultornsanee, S., & Chamnongthai, K. (2017). Electroencephalography-based feature extraction using complex network for automated epileptic seizure detection. *Expert Systems, 34*(3), e12211.
- Alickovic, E., Kevric, J., & Subasi, A. (2018). Performance evaluation of empirical mode decomposition, discrete wavelet transform, and wavelet packed

## References

---

- decomposition for automated epileptic seizure detection and prediction. *Biomedical Signal Processing and Control*, 39, 94-102.
- Abibullaev, B., Seo, H. D., & Kim, M. S. (2010). Epileptic spike detection using continuous wavelet transforms and artificial neural networks. *International journal of wavelets, multiresolution and information processing*, 8(01), 33-48.
- Ames, F. R. (1971). "Self-induction" in photosensitive epilepsy. *Brain*, 94(4), 781-798.
- Buck, D., Baker, G. A., Jacoby, A., Smith, D. F., & Chadwick, D. W. (1997). Patients' experiences of injury as a result of epilepsy. *Epilepsia*, 38(4), 439-444.
- Brain waves, (<http://www.crystalblueent.com/brain-waves.html>) (Accessed April 2018).
- Bajaj, V., Rai, K., Kumar, A., & Sharma, D. (2017). Time-frequency image based features for classification of epileptic seizures from EEG signals. *Biomedical Physics & Engineering Express*, 3(1), 015012.
- Berg, A. T. (2008). Risk of recurrence after a first unprovoked seizure. *Epilepsia*, 49, 13-18.
- Başar, E., & Schürmann, M. (1999). Brain functioning: integrative models. In *Brain function and oscillations* (pp. 393-406). Springer, Berlin, Heidelberg.
- Bhattacharyya, A., Sharma, M., Pachori, R. B., Sircar, P., & Acharya, U. R. (2018). A novel approach for automated detection of focal EEG signals using empirical wavelet transform. *Neural Computing and Applications*, 29(8), 47-57.
- Coenen, A., Fine, E., & Zayachkivska, O. (2014). Adolf Beck: A forgotten pioneer in electroencephalography. *Journal of the History of the Neurosciences*, 23(3), 276-286.
- Chen, L., Zhao, E., Wang, D., Han, Z., Zhang, S., & Xu, C. (2010). *Feature extraction of EEG signals from epilepsy patients based on Gabor transform and EMD decomposition*. Paper presented at the Natural Computation (ICNC), 2010 Sixth International Conference on.
- Cahn, B. R., & Polich, J. (2013). Meditation states and traits: EEG, ERP, and neuroimaging studies.
- Choi, K. S., Zeng, Y., & Qin, J. (2012, October). Using sequential floating forward selection algorithm to detect epileptic seizure in EEG signals. In *Signal Processing (ICSP), 2012 IEEE 11th International Conference on* (Vol. 3, pp. 1637-1640). IEEE.

- Chua, K. C., Chandran, V., Acharya, R., & Lim, C. M. (2007). Higher order spectral (HOS) analysis of epileptic EEG signals.
- Chua, K. C., Chandran, V., Acharya, U. R., & Lim, C. M. (2011). Application of higher order spectra to identify epileptic EEG. *Journal of medical systems*, 35(6), 1563-1571.
- Chen, S., Zhang, X., & Yang, Z. (2017). Epileptic seizure detection by combining robust-principal component analysis and least square-support vector machine. *International Journal of Imaging Systems and Technology*, 27(4), 368-375.
- Chen, D., Wan, S., & Bao, F. S. (2017). Epileptic Focus Localization Using Discrete Wavelet Transform Based on Interictal Intracranial EEG. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(5), 413-425.
- Diykh, M., & Li, Y. (2016). Complex networks approach for EEG signal sleep stages classification. *Expert Systems with Applications*, 63, 241-248.
- Diykh, M., Li, Y., & Wen, P. (2017). Classify epileptic EEG signals using weighted complex networks based community structure detection. *Expert Systems with Applications*, 90, 87-100.
- Davey, G. (2011). *Applied Psychology*. John Wiley & Sons. p. 978. ISBN 1444331219.
- Faust, O., Acharya, U. R., Adeli, H., & Adeli, A. (2015). Wavelet-based EEG processing for computer-aided seizure detection and epilepsy diagnosis. *Seizure-European Journal of Epilepsy*, 26, 56-64.
- Fisher, R. S., Boas, W. V. E., Blume, W., Elger, C., Genton, P., Lee, P., & Engel Jr, J. (2005). Epileptic seizures and epilepsy: definitions proposed by the International League Against Epilepsy (ILAE) and the International Bureau for Epilepsy (IBE). *Epilepsia*, 46(4), 470-472.
- Frohlich, J., Senturk, D., Saravanapandian, V., Golshani, P., Reiter, L. T., Sankar, R., ... & Jeste, S. S. (2016). A quantitative electrophysiological biomarker of duplication 15q11. 2-q13. 1 syndrome. *PloS one*, 11(12), e0167179.
- Gerrard, P., & Malcolm, R. (2007). Mechanisms of modafinil: a review of current research. *Neuropsychiatric disease and treatment*, 3(3), 349.
- Gray, F. J. (2002). *Anatomy for the medical clinician*, first edition, Shannon Books Pty Ltd, Victoria, Australia.

- Gajic, D., Djurovic, Z., Di Gennaro, S., & Gustafsson, F. (2014). Classification of EEG signals for detection of epileptic seizures based on wavelets and statistical pattern recognition. *Biomedical Engineering: Applications, Basis and Communications*, 26(02), 1450021.
- Gajic, D., Djurovic, Z., Gligorijevic, J., Di Gennaro, S., & Savic-Gajic, I. (2015). Detection of epileptiform activity in EEG signals based on time-frequency and non-linear analysis. *Frontiers in computational neuroscience*, 9, 38.
- Guo, L., Rivero, D., Dorado, J., Munteanu, C. R., & Pazos, A. (2011). Automatic feature extraction using genetic programming: An application to epileptic EEG classification. *Expert Systems with Applications*, 38(8), 10425-10436.
- Hall, J. E. (2015). *Guyton and Hall textbook of medical physiology e-Book*: Elsevier Health Sciences.
- Herculano-Houzel, S. (2009). The human brain in numbers: a linearly scaled-up primate brain. *Frontiers in human neuroscience*, 3, 31.
- Haas, L. F. (2003). Hans berger (1873–1941), richard caton (1842–1926), and electroencephalography. *Journal of Neurology, Neurosurgery & Psychiatry*, 74(1), 9-9.
- Huang, G., Song, S., Gupta, J. N., & Wu, C. (2014). Semi-supervised and unsupervised extreme learning machines. *IEEE Trans. Cybernetics*, 44(12), 2405-2417.
- Husain, S. J., & Rao, K. S. (2014). An artificial neural network model for classification of epileptic seizures using Huang-Hilbert Transform. *International Journal on Soft Computing*, 5(3), 23.
- Iasemidis, L. D. (2003). Epileptic seizure prediction and control. *IEEE Transactions on Biomedical Engineering*, 50(5), 549-558.
- Joshi, V., Pachori, R. B., & Vijesh, A. (2014). Classification of ictal and seizure-free EEG signals using fractional linear prediction. *Biomedical Signal Processing and Control*, 9, 1-5.
- Kumar, T. S., Kanhangad, V., & Pachori, R. B. (2015). Classification of seizure and seizure-free EEG signals using local binary patterns. *Biomedical Signal Processing and Control*, 15, 33-40.
- Kutlu, Y., Kuntalp, M., & Kuntalp, D. (2009). Optimizing the performance of an MLP classifier for the automatic detection of epileptic spikes. *Expert Systems with Applications*, 36(4), 7567-7575.

## References

---

- Kandel, E. R., Schwartz, J. H., Jessell, T. M., Siegelbaum, S. A., & Hudspeth, A. J. (2000). *Principles of neural science* (Vol. 4): McGraw-hill New York.
- Kiyimik, M. K., Subasi, A., & Ozcalik, H. R. (2004). Neural networks with periodogram and autoregressive spectral analysis methods in detection of epileptic seizure. *Journal of Medical Systems*, 28(6), 511-522.
- Klem, G. H., Lüders, H. O., Jasper, H. H., & Elger, C. (1999). The ten-twenty electrode system of the International Federation. *Electroencephalogr Clin Neurophysiol*, 52(3), 3-6.
- Kumar, J. S., & Bhuvaneshwari, P. (2012). Analysis of Electroencephalography (EEG) signals and its categorization—a study. *signal*, 25, 26.
- Lindsay, P. H., & Norman, D. A. (2013). *Human information processing: An introduction to psychology*: Academic press.
- McGrogan, N. (1999). Neural network detection of epileptic seizures in the electroencephalogram.
- Millet, D., (2002). The origins of EEG. In *7th Annual Meeting of the International Society for the History of the Neurosciences (ISHN)*.
- McPhee, S. J., & Hammer, G. D. (1995). *Pathophysiology of Disease: An Introduction to Clinical Medicine*, (Lange Medical Books). USA.
- Martis, R. J., Acharya, U. R., Tan, J. H., Petznick, A., Yanti, R., Chua, C. K., ... & Tong, L. (2012). Application of empirical mode decomposition (EMD) for automated detection of epilepsy using EEG signals. *International journal of neural systems*, 22(06), 1250027.
- Martis, R. J., Tan, J. H., Chua, C. K., Loon, T. C., YEO, S. W. J., & Tong, L. (2015). Epileptic EEG classification using nonlinear parameters on different frequency bands. *Journal of Mechanics in Medicine and Biology*, 15(03), 1550040.
- Niedermeyer, E., & Silva, F. L. (Eds.). (2005). *Electroencephalography: basic principles, clinical applications, and related fields*. Lippincott Williams & Wilkins
- Nunez, P. L., & Pilgreen, K. L. (1991). The spline-Laplacian in clinical neurophysiology: a method to improve EEG spatial resolution. *Journal of clinical neurophysiology: official publication of the American Electroencephalographic Society*, 8(4), 397-413.
- National, C. G. C. U. (2012). The Epilepsies: The Diagnosis and Management of the Epilepsies in Adults and Children in Primary and Secondary Care: Pharmacological Update of Clinical Guideline 20.

- Pfurtscheller, G., & Da Silva, F. L. (1999). Event-related EEG/MEG synchronization and desynchronization: basic principles. *Clinical neurophysiology*, *110*(11), 1842-1857.
- Radocy, R. E., & Boyle, J. D. (2012). *Psychological foundations of musical behavior*: Charles C Thomas Publisher.
- Salem, O., Naseem, A., & Mehaoua, A. (2014). *Epileptic seizure detection from EEG signal using Discrete Wavelet Transform and Ant Colony classifier*. Paper presented at the Communications (ICC), 2014 IEEE International Conference on.
- Sharma, P., Khan, Y. U., Farooq, O., Tripathi, M., & Adeli, H. (2014). A wavelet-statistical features approach for nonconvulsive seizure detection. *Clinical EEG and neuroscience*, *45*(4), 274-284.
- Sanei, S., & Chambers, J. A. (2013). *EEG signal processing*: John Wiley & Sons.
- Sejdic, E., Djurović, I., & Jiang, J. (2009). Time--frequency feature representation using energy concentration: An overview of recent advances. *Digital Signal Processing*, *19*(1), 153-183.
- Selesnick, I. W. (2011). Resonance-based signal decomposition: A new sparsity-enabled signal analysis method. *Signal Processing*, *91*(12), 2793-2809.
- Sharma, R., & Pachori, R. B. (2015). Classification of epileptic seizures in EEG signals based on phase space representation of intrinsic mode functions. *Expert Systems with Applications*, *42*(3), 1106-1117.
- Shearer, P., Park, D., Bowman, A. J., & Huff, S. J. (2006). Seizures and status epilepticus: Diagnosis and management in the emergency department. *Emergency Medicine Practice+ Em Practice Guidelines Update*, *8*(8), 1-31.
- Shen, C. P., Chen, C. C., Hsieh, S. L., Chen, W. H., Chen, J. M., Chen, C. M., ... & Chiu, M. J. (2013). High-performance seizure detection system using a wavelet-approximate entropy-fSVM cascade with clinical validation. *Clinical EEG and neuroscience*, *44*(4), 247-256.
- Sharma, M., Pachori, R. B., & Acharya, U. R. (2017). A new approach to characterize epileptic seizures using analytic time-frequency flexible wavelet transform and fractal dimension. *Pattern Recognition Letters*, *94*, 172-179.
- Siuly, S., & Li, Y. (2015). Designing a robust feature extraction method based on optimum allocation and principal component analysis for epileptic EEG signal classification. *Computer methods and programs in biomedicine*, *119*(1), 29-42.



- Siuly, S., & Zhang, Y. (2016). Medical big data: neurological diseases diagnosis through medical data analysis. *Data Science and Engineering*, 1(2), 54-64.
- Siuly, S., Li, Y., & Wen, P. (2009, July). Classification of EEG signals using sampling techniques and least square support vector machines. In *International Conference on Rough Sets and Knowledge Technology* (pp. 375-382). Springer, Berlin, Heidelberg.
- Siuly, S., Li, Y., & Zhang, Y. (2017). EEG Signal Analysis and Classification: Techniques and Applications. *Health Information Science*, Springer Nature, US.
- Siuly, S., Li, Y., & Zhang, Y. (2016). *EEG Signal Analysis and Classification*. Springer International Publishing.
- Standing, S. (2015). *Gray's anatomy e-book: the anatomical basis of clinical practice*: Elsevier Health Sciences.
- Subasi, A., & Ercelebi, E. (2005a). Classification of EEG signals using neural network and logistic regression. *Computer methods and programs in biomedicine*, 78(2), 87-99.
- Subasi, A., & Gursoy, M. I. (2010). EEG signal classification using PCA, ICA, LDA and support vector machines. *Expert systems with applications*, 37(12), 8659-8666.
- Supriya, S., Siuly, S., & Zhang, Y. (2016). Automatic epilepsy detection from EEG introducing a new edge weight method in the complex network. *Electronics Letters*, 52(17), 1430-1432.
- Swartz, B. E. (1998). The advantages of digital over analog recording techniques. *Electroencephalography and clinical neurophysiology*, 106(2), 113-117.
- Tatum IV, W. O. (2014). *Handbook of EEG interpretation*: Demos Medical Publishing.
- Teplan, M. (2002). Fundamentals of EEG measurement. *Measurement science review*, 2(2), 1-11.
- Towle, V. L., Bolaños, J., Suarez, D., Tan, K., Grzeszczuk, R., Levin, D. N., . . . Spire, J.-P. (1993). The spatial location of EEG electrodes: locating the best-fitting sphere relative to cortical anatomy. *Electroencephalography and clinical neurophysiology*, 86(1), 1-6.
- Tzamourta, K. D., Tzallas, A. T., Giannakeas, N., Astrakas, L. G., Tsalikakis, D. G., & Tsipouras, M. G. (2018). Epileptic Seizures Classification Based on Long-Term

## References

---

- EEG Signal Wavelet Analysis. In *Precision Medicine Powered by pHealth and Connected Health* (pp. 165-169). Springer, Singapore.
- Usman, S. M., & Hassan, A. (2018). Efficient Prediction and Classification of Epileptic Seizures Using EEG Data Based on Univariate Linear Features. *Journal of Computers*, 13(6), 616-622.
- Wang, L., Long, X., Arends, J. B., & Aarts, R. M. (2017). EEG analysis of seizure patterns using visibility graphs for detection of generalized seizures. *Journal of neuroscience methods*, 290, 85-94.
- WHO (World Health Organization) Report (2018). (<http://www.who.int/mediacentre/factsheets/fs999/en/>) (Accessed April 2018).
- Wilden, J. A., & Cohen-Gadol, A. A. (2012). Evaluation of first nonfebrile seizures. *American family physician*, 86(4).
- Wolpaw, J. R., Birbaumer, N., McFarland, D. J., Pfurtscheller, G., & Vaughan, T. M. (2002). Brain-computer interfaces for communication and control. *Clinical neurophysiology*, 113(6), 767-791.
- Zhang, Z., & Parhi, K. K. (2014). *Seizure detection using wavelet decomposition of the prediction error signal from a single channel of intra-cranial EEG*. Paper presented at the Engineering in Medicine and Biology Society (EMBC), 2014 36th Annual International Conference of the IEEE.
- Zhu, G., Li, Y., & Wen, P. P. (2014). Analysis and classification of sleep stages based on difference visibility graphs from a single-channel EEG signal. *IEEE journal of biomedical and health informatics*, 18(6), 1813-1821.
- Zhu, G., Li, Y., & Wen, P. P. (2014). Epileptic seizure detection in EEGs signals using a fast weighted horizontal visibility algorithm. *Computer methods and programs in biomedicine*, 115(2), 64-75.

# **APPENDICES**



# A

## **Matlab simulation code for Simple Random Sampling and Sequential Feature Selection**

A simulation code for classification of epileptic EEG signals is presented.

A.I Matlab code for simple random sampling (SRS) and sequential feature selection (SFS).

## Appendices

---

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
  /\%%%%%%%%%% Read Epileptic EEG database %%%%%%%%%%/\
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clear all
clc

for i=1:9
    zds=sprintf('Z0%d.txt',i);
    ods=sprintf('O0%d.txt',i);
    nds=sprintf('N0%d.txt',i);
    fds=sprintf('F0%d.txt',i);
    sds=sprintf('S0%d.txt',i);
    z(:,i)=load(zds);
    o(:,i)=load(ods);
    n(:,i)=load(nds);
    f(:,i)=load(fds);
    s(:,i)=load(sds);
end
for i=10:99
    zds=sprintf('Z0%d.txt',i);
    ods=sprintf('O0%d.txt',i);
    nds=sprintf('N0%d.txt',i);
    fds=sprintf('F0%d.txt',i);
    sds=sprintf('S0%d.txt',i);
    z(:,i)=load(zds);
    o(:,i)=load(ods);
    n(:,i)=load(nds);
    f(:,i)=load(fds);
    s(:,i)=load(sds);
end
for i=100
    zds=sprintf('Z%d.txt',i);
    ods=sprintf('O%d.txt',i);
    nds=sprintf('N%d.txt',i);
    fds=sprintf('F%d.txt',i);
    sds=sprintf('S%d.txt',i);

    z(:,i)=load(zds);
    o(:,i)=load(ods);
    n(:,i)=load(nds);
    f(:,i)=load(fds);
    s(:,i)=load(sds);
end

x=[z s];          % Combine set A (z) and set E (s)
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                %%%%%%%%%%% Function of SRS Technique %%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
m=1;s=[];ss=[];
for ii=1:200

    for jj=1:10

        s(:,jj)=randsample(x(:,ii),3287);

        for k=1:5
            ss(:,k)=randsample(s(:,jj),2745);
```

```

fs1(:,k)=[min(ss(:,k)),max(ss(:,k)),mean(ss(:,k)),median(ss(:,k)),mo
de(ss(:,k)),median(ss(:,k))/2,3*median(ss(:,k))/2,(3*median(ss(:,k))
/2-median(ss(:,k))/2), std(ss(:,k))];
    end
fs2(m,:)=fs1(:);      % Extracted features
m=m+1;

    end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                %%%%%%%%% Function of SFS technique %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear all
clc
load fs2 % load the features were extracted from Simple Random
        Sampling method (SRST2).
P=fs2;
[z1 z2]=size(P);
    for i=1:z1
        N=P(i,:);
        N1=sqfeat1(N); % calls the sqfeat1 function to select the key
                    features.
        [M M1]=size(N1);
        SFS_feature(i,1:M1)=N1(1,1:M1);
    end
%%
tic
X=SFS_feature;
x=[X(1:500,:);X(1001:1500,:)];
y=[1*ones(500,1);-1*ones(500,1)];

Xtest= [X(501:1000,:);X(1501:2000,:)];
Ytest=[1*ones(500,1);-1*ones(500,1)];

size(x)
size(y)
%%
                %%%%%%%%% LS_SVM classifier %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
gam=10;
sig2=1;
type='classification';

[alpha,b] = trainlssvm({x,y,type,gam,sig2,'RBF_kernel'});
disp(b)

Yh=simlssvm({x,y,type,gam,sig2,'RBF_kernel'},{alpha,b},Xtest);
plotlssvm({x,y,type,gam,sig2,'RBF_kernel'},{alpha,b});
%%

[perc,n,which]=misclass(Ytest,Yh); % Which: contains the indices of
                                the misclassified instances(the
                                first column gives the row, the
                                second the column index)
n; % is the number of misclassifications
perc; % is the rate of misclassifications (between 0 and 1)
[C,order] = confusionmat(Ytest,Yh);

```

## Appendices

---

C

order

```
%%  
Y_latent=latentlssvm({x,y,type,gam,sig2,'RBF_kernel'},{alpha,b},x);  
[area,se,thresholds,oneMinusspec,sens,TN,TP,FN,FP]=roc({x,y,type,gam  
,sig2,'RBF_kernel'});  
%[thresholds oneMinusspec sens ];
```

```
%%  
Sensitivity =TP/(TP+FN)*100  
Specificity =TN/(TN+FP)*100  
Accuracy =(TP+TN)/(TP+TN+FP+FN)*100  
toc
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
                %%%%%%%%% Sub-Function of SFS technique %%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
function [ Y ] = Untitled2( X )  
%UNTITLED2 Summary of this function goes here  
% Detailed explanation goes here
```

```
Y=[]  
[L1 L2]=size(X)  
for j=1:L1  
    k=0  
ave= mean(X(j,:)); % select different types of features such as  
                    % {min(X(j,:)),max(X(j,:)),median(X(j,:)),mode(X(j  
                    ,:)),median(X(j,:))/2,3*median(X(j,:))/2,(3*media  
                    n(X(j,:))/2-median(X(j,:))/2), std(X(j,:))}  
  
    for i=1:L2  
        fnumber=X(j,i);  
        if fnumber<=ave % check the feature depend on (<= or >=) and  
                        % then bet the selected features in vector which  
                        % is fnumber.  
            k=k+1;  
            Y1(k)=X(j,i);  
        end  
    end  
    Y(j,:)=Y1(1,:);
```

```
end
```



# B

## **Matlab simulation code for Optimum Allocation and Power Spectral Density Estimation**

A simulation code for classification of epileptic EEG signals is presented.

B.I Matlab code for optimum allocation (OA) and power spectrum density estimation.

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
  /\%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Read Epileptic EEG %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%/\
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

### Reading epileptic EEG data is same as in the appendix A

```

%%
sign1=[z];
sign2=[o];
sign3=[n];
sign4=[f];
sign5=[s];
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Segmentation %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
sign11=sign1(1:1024,:);
sign12=sign1(1025:2048,:);
sign13=sign1(2049:3072,:);
sign14=sign1(3073:4097,:);

sign21=sign2(1:1024,:);
sign22=sign2(1025:2048,:);
sign23=sign2(2049:3072,:);
sign24=sign2(3073:4097,:);

sign31=sign3(1:1024,:);
sign32=sign3(1025:2048,:);
sign33=sign3(2049:3072,:);
sign34=sign3(3073:4097,:);

sign41=sign4(1:1024,:);
sign42=sign4(1025:2048,:);
sign43=sign4(2049:3072,:);
sign44=sign4(3073:4097,:);

sign51=sign5(1:1024,:);
sign52=sign5(1025:2048,:);
sign53=sign5(2049:3072,:);
sign54=sign5(3073:4097,:);
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Apply OA with PSDE %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% class A %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

sum(var(sign11(:,1:100)));
sum(var(sign12(:,1:100)));
sum(var(sign13(:,1:100)));
sum(var(sign14(:,1:100)));

sign11_n=
    round(length(sign11)*sqrt(sum(var(sign11(:,1:100))))*3288/(1
    ength(sign11)*sqrt(sum(var(sign11(:,1:100))))+length(sign12)
    *sqrt(sum(var(sign12(:,1:100))))+length(sign13)*sqrt(sum(var
    (sign13(:,1:100))))+length(sign14)*sqrt(sum(var(sign14(:,1:1
    00))))));

```

## Appendices

---

```
%%% sign11_n= sample size from x11 by optimum allocation formula
sign12_n=
    round(length(sign12)*sqrt(sum(var(sign12(:,1:100))))*3288/(length(sign11)*sqrt(sum(var(sign11(:,1:100))))+length(sign12)*sqrt(sum(var(sign12(:,1:100))))+length(sign13)*sqrt(sum(var(sign13(:,1:100))))+length(sign14)*sqrt(sum(var(sign14(:,1:100))))));
sign13_n=
    round(length(sign13)*sqrt(sum(var(sign13(:,1:100))))*3288/(length(sign11)*sqrt(sum(var(sign11(:,1:100))))+length(sign12)*sqrt(sum(var(sign12(:,1:100))))+length(sign13)*sqrt(sum(var(sign13(:,1:100))))+length(sign14)*sqrt(sum(var(sign14(:,1:100))))));
sign14_n=
    round(length(sign14)*sqrt(sum(var(sign14(:,1:100))))*3288/(length(sign11)*sqrt(sum(var(sign11(:,1:100))))+length(sign12)*sqrt(sum(var(sign12(:,1:100))))+length(sign13)*sqrt(sum(var(sign13(:,1:100))))+length(sign14)*sqrt(sum(var(sign14(:,1:100))))));
%%

posi_sign11_n=randsample([1:size(sign11,1)],sign11_n);
sample_sign11_n=sign11(posi_sign11_n,:);
pxx11 = periodogram(sample_sign11_n); %Extract periodogram from the
                                     1st window for set A
AuR11 = pcov(sample_sign11_n,5);      %Extract Autoregressive from
                                     the 1st window for set A with
                                     order time is 5

posi_sign12_n=randsample([1:size(sign12,1)],sign12_n);
sample_sign12_n=sign12(posi_sign12_n,:);
pxx12 = periodogram(sample_sign12_n); %Extract periodogram from the
                                     2nd window for set A
AuR12 = pcov(sample_sign12_n,5);      %Extract Autoregressive from
                                     the 2nd window for set A with
                                     order time is 5

posi_sign13_n=randsample([1:size(sign13,1)],sign13_n);
sample_sign13_n=sign13(posi_sign13_n,:);
pxx13 = periodogram(sample_sign13_n); %Extract periodogram from the
                                     3rd window for set A
AuR13 = pcov(sample_sign13_n,5);      %Extract Autoregressive from
                                     the 3rd window for set A with
                                     order time is 5

posi_sign14_n=randsample([1:size(sign14,1)],sign14_n);
sample_sign14_n=sign14(posi_sign14_n,:);
pxx14 = periodogram(sample_sign14_n); %Extract periodogram from the
                                     4th window for set A
AuR14 = pcov(sample_sign14_n,5);      %Extract Autoregressive from
                                     the 4th window for set A with
                                     order time is 5

new_sign1=[sample_sign11_n; sample_sign12_n; sample_sign13_n;
sample_sign14_n];
pxx1=[pxx11;pxx12;pxx13;pxx14];      %All periodogram from four
                                     windows of set A
AuR_A=[AuR11;AuR12;AuR13;AuR14];    %All Autoregressive from four
                                     windows of set A
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% class B %%%%%%%%%
sum(var(sign21(:,1:100)));
```

## Appendices

---

```
sum(var(sign22(:,1:100)));
sum(var(sign23(:,1:100)));
sum(var(sign24(:,1:100)));

sign21_n=
    round(length(sign21)*sqrt(sum(var(sign21(:,1:100))))*3288/(1
    ength(sign21)*sqrt(sum(var(sign21(:,1:100))))+length(sign22)
    *sqrt(sum(var(sign22(:,1:100))))+length(sign23)*sqrt(sum(var
    (sign23(:,1:100))))+length(sign24)*sqrt(sum(var(sign24(:,1:1
    00))))));
%% sign21_n= sample size from x21
sign22_n=
    round(length(sign22)*sqrt(sum(var(sign22(:,1:100))))*3288/(1
    ength(sign21)*sqrt(sum(var(sign21(:,1:100))))+length(sign22)
    *sqrt(sum(var(sign22(:,1:100))))+length(sign23)*sqrt(sum(var
    (sign23(:,1:100))))+length(sign24)*sqrt(sum(var(sign24(:,1:1
    00))))));
sign23_n=
    round(length(sign23)*sqrt(sum(var(sign23(:,1:100))))*3288/(1
    ength(sign21)*sqrt(sum(var(sign21(:,1:100))))+length(sign22)
    *sqrt(sum(var(sign22(:,1:100))))+length(sign23)*sqrt(sum(var
    (sign23(:,1:100))))+length(sign24)*sqrt(sum(var(sign24(:,1:1
    00))))));
sign24_n=
    round(length(sign24)*sqrt(sum(var(sign24(:,1:100))))*3288/(1
    ength(sign21)*sqrt(sum(var(sign21(:,1:100))))+length(sign22)
    *sqrt(sum(var(sign22(:,1:100))))+length(sign23)*sqrt(sum(var
    (sign23(:,1:100))))+length(sign24)*sqrt(sum(var(sign24(:,1:1
    00))))));
%%

posi_sign21_n=randsample([1:size(sign21,1)],sign21_n);
sample_sign21_n=sign21(posi_sign21_n,:);
pxx21 = periodogram(sample_sign21_n); %Extract periodogram from the
    1st window for set B
AuR21 = pcov(sample_sign21_n,5); %Extract Autoregressive from
    the 1st window for set B with
    order time is 5
posi_sign22_n=randsample([1:size(sign22,1)],sign22_n);
sample_sign22_n=sign22(posi_sign22_n,:);
pxx22 = periodogram(sample_sign22_n); %Extract periodogram from the
    2nd window for set B
AuR22 = pcov(sample_sign22_n,5); %Extract Autoregressive from
    the 2nd window for set B with
    order time is 5
posi_sign23_n=randsample([1:size(sign23,1)],sign23_n);
sample_sign23_n=sign23(posi_sign23_n,:);
pxx23 = periodogram(sample_sign23_n); %Extract periodogram from the
    3rd window for set B
AuR23 = pcov(sample_sign23_n,5); %Extract Autoregressive from
    the 3rd window for set B with
    order time is 5
posi_sign24_n=randsample([1:size(sign24,1)],sign24_n);
sample_sign24_n=sign24(posi_sign24_n,:);
pxx24 = periodogram(sample_sign24_n); %Extract periodogram from the
    4th window for set B
AuR24 = pcov(sample_sign24_n,5); %Extract Autoregressive from
    the 4th window for set B with
    order time is 5
new_sign2=[sample_sign21_n; sample_sign22_n; sample_sign23_n;
sample_sign24_n];
```

## Appendices

---

```
pxx2=[pxx21;pxx22;pxx23;pxx24];           %All periodogram form four
                                           windows of set B
AuR_B=[AuR21;AuR22;AuR23;AuR24];       %All Autoregressive from four
                                           windows of set B

%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Class C %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

sum(var(sign31(:,1:100)));
sum(var(sign32(:,1:100)));
sum(var(sign33(:,1:100)));
sum(var(sign34(:,1:100)));

sign31_n=
    round(length(sign31)*sqrt(sum(var(sign31(:,1:100))))*3288/(1
    ength(sign31)*sqrt(sum(var(sign31(:,1:100))))+length(sign32)
    *sqrt(sum(var(sign32(:,1:100))))+length(sign33)*sqrt(sum(var
    (sign33(:,1:100))))+length(sign34)*sqrt(sum(var(sign34(:,1:1
    00))))));
%% sign31_n= sample size from x31
sign32_n=
    round(length(sign32)*sqrt(sum(var(sign32(:,1:100))))*3288/(1
    ength(sign31)*sqrt(sum(var(sign31(:,1:100))))+length(sign32)
    *sqrt(sum(var(sign32(:,1:100))))+length(sign33)*sqrt(sum(var
    (sign33(:,1:100))))+length(sign34)*sqrt(sum(var(sign34(:,1:1
    00))))));
sign33_n=
    round(length(sign33)*sqrt(sum(var(sign33(:,1:100))))*3288/(1
    ength(sign31)*sqrt(sum(var(sign31(:,1:100))))+length(sign32)
    *sqrt(sum(var(sign32(:,1:100))))+length(sign33)*sqrt(sum(var
    (sign33(:,1:100))))+length(sign34)*sqrt(sum(var(sign34(:,1:1
    00))))));
sign34_n=
    round(length(sign34)*sqrt(sum(var(sign34(:,1:100))))*3288/(1
    ength(sign31)*sqrt(sum(var(sign31(:,1:100))))+length(sign32)
    *sqrt(sum(var(sign32(:,1:100))))+length(sign33)*sqrt(sum(var
    (sign33(:,1:100))))+length(sign34)*sqrt(sum(var(sign34(:,1:1
    00))))));

%%

posi_sign31_n=randsample([1:size(sign31,1)],sign31_n);
sample_sign31_n=sign31(posi_sign31_n,:);
pxx31 = periodogram(sample_sign31_n); %Extract periodogram from the
                                           1st window for set C
AuR31 = pcov(sample_sign31_n,5);         %Extract Autoregressive from
                                           the 1st window for set C with
                                           order time is 5
posi_sign32_n=randsample([1:size(sign32,1)],sign32_n);
sample_sign32_n=sign32(posi_sign32_n,:);
pxx32 = periodogram(sample_sign32_n); %Extract periodogram from the
                                           2nd window for set C
AuR32 = pcov(sample_sign32_n,5);         %Extract Autoregressive from
                                           the 2nd window for set C with
                                           order time is 5
posi_sign33_n=randsample([1:size(sign33,1)],sign33_n);
sample_sign33_n=sign33(posi_sign33_n,:);
pxx33 = periodogram(sample_sign33_n); %Extract periodogram from the
                                           3rd window for set C
AuR33 = pcov(sample_sign33_n,5);         %Extract Autoregressive from
                                           the 3rd window for set C with
                                           order time is 5
posi_sign34_n=randsample([1:size(sign34,1)],sign34_n);
```

## Appendices

---

```
sample_sign34_n=sign34(posi_sign34_n,:);
pxx34 = periodogram(sample_sign34_n); %Extract periodogram from the
                                        4th window for set C
AuR34 = pcov(sample_sign34_n,5);        %Extract Autoregressive from
                                        the 4th window for set C with
                                        order time is 5
new_sign3=[sample_sign31_n; sample_sign32_n; sample_sign33_n;
sample_sign34_n];
pxx3=[pxx31;pxx32;pxx33;pxx34];        %All periodogram form four
                                        windows for set C
AuR_C=[AuR31;AuR32;AuR33;AuR34];
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% class D %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

sum(var(sign41(:,1:100)));
sum(var(sign42(:,1:100)));
sum(var(sign43(:,1:100)));
sum(var(sign44(:,1:100)));

sign41_n=
    round(length(sign41)*sqrt(sum(var(sign41(:,1:100))))*3288/(1
length(sign41)*sqrt(sum(var(sign41(:,1:100))))+length(sign42)
*sqrt(sum(var(sign42(:,1:100))))+length(sign43)*sqrt(sum(var
(sign43(:,1:100))))+length(sign44)*sqrt(sum(var(sign44(:,1:1
00))))));
    %% x41_n= sample size from x41
sign42_n=
    round(length(sign42)*sqrt(sum(var(sign42(:,1:100))))*3288/(1
length(sign41)*sqrt(sum(var(sign41(:,1:100))))+length(sign42)
*sqrt(sum(var(sign42(:,1:100))))+length(sign43)*sqrt(sum(var
(sign43(:,1:100))))+length(sign44)*sqrt(sum(var(sign44(:,1:1
00))))));
sign43_n=
    round(length(sign43)*sqrt(sum(var(sign43(:,1:100))))*3288/(1
length(sign41)*sqrt(sum(var(sign41(:,1:100))))+length(sign42)
*sqrt(sum(var(sign42(:,1:100))))+length(sign43)*sqrt(sum(var
(sign43(:,1:100))))+length(sign44)*sqrt(sum(var(sign44(:,1:1
00))))));
sign44_n=
    round(length(sign44)*sqrt(sum(var(sign44(:,1:100))))*3288/(1
length(sign41)*sqrt(sum(var(sign41(:,1:100))))+length(sign42)
*sqrt(sum(var(sign42(:,1:100))))+length(sign43)*sqrt(sum(var
(sign43(:,1:100))))+length(sign44)*sqrt(sum(var(sign44(:,1:1
00))))));
%%
posi_sign41_n=randsample([1:size(sign41,1)],sign41_n);
sample_sign41_n=sign41(posi_sign41_n,:);
pxx41 = periodogram(sample_sign41_n); %Extract periodogram from the
                                        1st window for set D
AuR41 = pcov(sample_sign41_n,5);        %Extract Autoregressive from
                                        the 1st window for set D with
                                        order time is 5
posi_sign42_n=randsample([1:size(sign42,1)],sign42_n);
sample_sign42_n=sign42(posi_sign42_n,:);
pxx42 = periodogram(sample_sign42_n); %Extract periodogram from the
                                        2nd window for set D
AuR42 = pcov(sample_sign42_n,5);        %Extract Autoregressive from
                                        the 2nd window for set D with
                                        order time is 5
posi_sign43_n=randsample([1:size(sign43,1)],sign43_n);
sample_sign43_n=sign43(posi_sign43_n,:);
```

## Appendices

---

```
pxx43 = periodogram(sample_sign43_n); %Extract periodogram from the
                                         3rd window for set D
AuR43 = pcov(sample_sign43_n,5);         %Extract Autoregressive from
                                         the 3rd window for set D with
                                         order time is 5
posi_sign44_n=randsample([1:size(sign44,1)],sign44_n);
sample_sign44_n=sign44(posi_sign44_n,:);
pxx44 = periodogram(sample_sign44_n); %Extract periodogram from the
                                         4th window for set D
AuR44 = pcov(sample_sign44_n,5);         %Extract Autoregressive from
                                         the 4th window for set D with
                                         order time is 5
new_sign4=[sample_sign41_n; sample_sign42_n; sample_sign43_n;
sample_sign44_n];
pxx4=[pxx41;pxx42;pxx43;pxx44];         %All periodogram form four
                                         windows for set D
AuR_D=[AuR41;AuR42;AuR43;AuR44];
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% class E %%%%%%%%%
sum(var(sign51(:,1:100)));
sum(var(sign52(:,1:100)));
sum(var(sign53(:,1:100)));
sum(var(sign54(:,1:100)));
sign51_n=
    round(length(sign51)*sqrt(sum(var(sign51(:,1:100))))*3288/(1
length(sign51)*sqrt(sum(var(sign51(:,1:100))))+length(sign52)
*sqrt(sum(var(sign52(:,1:100))))+length(sign53)*sqrt(sum(var
(sign53(:,1:100))))+length(sign54)*sqrt(sum(var(sign54(:,1:1
00))))));
    %% sign51_n= sample size from x51
sign52_n=
    round(length(sign52)*sqrt(sum(var(sign52(:,1:100))))*3288/(1
length(sign51)*sqrt(sum(var(sign51(:,1:100))))+length(sign52)
*sqrt(sum(var(sign52(:,1:100))))+length(sign53)*sqrt(sum(var
(sign53(:,1:100))))+length(sign54)*sqrt(sum(var(sign54(:,1:1
00))))));
sign53_n=
    round(length(sign53)*sqrt(sum(var(sign53(:,1:100))))*3288/(1
length(sign51)*sqrt(sum(var(sign51(:,1:100))))+length(sign52)
*sqrt(sum(var(sign52(:,1:100))))+length(sign53)*sqrt(sum(var
(sign53(:,1:100))))+length(sign54)*sqrt(sum(var(sign54(:,1:1
00))))));
sign54_n=
    round(length(sign54)*sqrt(sum(var(sign54(:,1:100))))*3288/(1
length(sign51)*sqrt(sum(var(sign51(:,1:100))))+length(sign52)
*sqrt(sum(var(sign52(:,1:100))))+length(sign53)*sqrt(sum(var
(sign53(:,1:100))))+length(sign54)*sqrt(sum(var(sign54(:,1:1
00))))));
%%
posi_sign51_n=randsample([1:size(sign51,1)],sign51_n);
sample_sign51_n=sign51(posi_sign51_n,:);
pxx51 = periodogram(sample_sign51_n); %Extract periodogram from the
                                         1st window for set E
AuR51 = pcov(sample_sign51_n,5);         %Extract Autoregressive from
                                         the 1st window for set E with
                                         order time is 5
posi_sign52_n=randsample([1:size(sign52,1)],sign52_n);
sample_sign52_n=sign52(posi_sign52_n,:);
```

## Appendices

---

```
pxx52 = periodogram(sample_sign52_n); %Extract periodogram from the
                                         2nd window for set E
AuR52 = pcov(sample_sign52_n,5);         %Extract Autoregressive from
                                         the 2nd window for set E with
                                         order time is 5
posi_sign53_n=randsample([1:size(sign53,1)],sign53_n);
sample_sign53_n=sign53(posi_sign53_n,:);
pxx53 = periodogram(sample_sign53_n); %Extract periodogram from the
                                         3rd window for set E
AuR53 = pcov(sample_sign53_n,5);         %Extract Autoregressive from
                                         the 3rd window for set E with
                                         order time is 5
posi_sign54_n=randsample([1:size(sign54,1)],sign54_n);
sample_sign54_n=sign54(posi_sign54_n,:);
pxx54 = periodogram(sample_sign54_n); %Extract periodogram from the
                                         4th window for set E
AuR54 = pcov(sample_sign54_n,5);         %Extract Autoregressive from
                                         the 4th window for set E with
                                         order time is 5
new_sign5=[sample_sign51_n; sample_sign52_n; sample_sign53_n;
sample_sign54_n];
pxx5=[pxx51;pxx52;pxx53;pxx54];         %All periodogram form four
                                         windows for set E
AuR_E=[AuR51;AuR52;AuR53;AuR54];
%%
X_all=[new_sign1; new_sign2; new_sign3; new_sign4; new_sign5];
pxx_all=[pxx1;pxx2;pxx3;pxx4;pxx5];     %All predictor
AuR_all=[AuR_A;AuR_B;AuR_C;AuR_D;AuR_E];
%%
%%%%%%%%%% Response classes of OA_PD %%%%%%%%%%
x1(1:2052,1)=[1]; % set A
x2(1:2052,1)=[2]; % set B
x3(1:2052,1)=[3]; % set C
x4(1:2052,1)=[4]; % set D
x5(1:2052,1)=[5]; % set E

x_all=[x1;x2;x3;x4;x5]; %All response
EEG=[pxx_all,x_all]; %Using the EEG in different
                                         classifiers(SVM; KNN; QDA)
%%%%%%%%%% Response classes of OA_AR %%%%%%%%%%
h_A(1:516,1)=[1];
h_B(1:516,1)=[2];
h_C(1:516,1)=[3];
h_D(1:516,1)=[4];
h_E(1:516,1)=[5];

h_all=[h_A;h_B;h_C;h_D;h_E]; %All response
EEG_AuR=[AuR_all,h_all]; %Using the EEG_AuR in different
                                         classifiers(SVM; KNN; QDA)
%%
%%%%%%%%%% Response Classes of OA_PD_AR %%%%%%%%%%
z1(1:2568,1)=[1];
z2(1:2568,1)=[2];
z3(1:2568,1)=[3];
z4(1:2568,1)=[4];
z5(1:2568,1)=[5];

Z_all=[z1;z2;z3;z4;z5]; %All response
OA_PD_AR=[Hybrid_all,Z_all]; %Using the OA_PD_AR set in different
classifiers(SVM; KNN; QDA)
```



## Appendices

---

```
%%%%%%%%%% Recall the classifiers %%%%%%%%%%
```

```
[Acc_KNN_PD, Acc_SVM_PD, Acc_QDA_PD]=Classifiers (OA_PD);  
[Acc_KNN_AR, Acc_SVM_AR, Acc_QDA_AR]=Classifiers (OA_AR);  
[Acc_KNN_PD_AR, Acc_SVM_PD_AR, Acc_QDA_PD_AR]=Classifiers  
(OA_PD_AR);
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%% Functions of three classifiers %%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
function [Accuracy_KNN, Accuracy_SVM, Accuracy_QDA] =  
trainClassifier(trainingData)  
% trainClassifier(trainingData)  
% returns a trained classifier and its accuracy.  
%  
% Input:  
% trainingData: the training data is 100 channels which are  
% predictors and one channel response (table or matrix).  
%  
% Output:  
% trainedClassifier: a struct containing the trained  
% classifier.  
% The struct contains various fields with information about  
the  
% trained classifier.  
%  
% trainedClassifier.predictFcn: a function to make predictions  
% on new data. It takes an input of the same form as this  
training  
% code (table or matrix) and returns predictions for the  
response.  
% If you supply a matrix, include only the predictors columns  
(or rows).  
%  
% Accuracy: a double containing the validation accuracy *100  
  
% Convert input to table  
inputTable = table(trainingData);  
inputTable.Properties.VariableNames = {'column'};  
  
% Split matrices in the input table into vectors  
inputTable =  
[inputTable(:,setdiff(inputTable.Properties.VariableNames,  
{'column'})), array2table(table2array(inputTable(:,{'column'})),  
'VariableNames', {'column_1', 'column_2', 'column_3', 'column_4',  
'column_5', 'column_6', 'column_7', 'column_8', 'column_9',  
'column_10', 'column_11', 'column_12', 'column_13', 'column_14',  
'column_15', 'column_16', 'column_17', 'column_18', 'column_19',  
'column_20', 'column_21', 'column_22', 'column_23', 'column_24',  
'column_25', 'column_26', 'column_27', 'column_28', 'column_29',  
'column_30', 'column_31', 'column_32', 'column_33', 'column_34',  
'column_35', 'column_36', 'column_37', 'column_38', 'column_39',
```

```

'column_40', 'column_41', 'column_42', 'column_43', 'column_44',
'column_45', 'column_46', 'column_47', 'column_48', 'column_49',
'column_50', 'column_51', 'column_52', 'column_53', 'column_54',
'column_55', 'column_56', 'column_57', 'column_58', 'column_59',
'column_60', 'column_61', 'column_62', 'column_63', 'column_64',
'column_65', 'column_66', 'column_67', 'column_68', 'column_69',
'column_70', 'column_71', 'column_72', 'column_73', 'column_74',
'column_75', 'column_76', 'column_77', 'column_78', 'column_79',
'column_80', 'column_81', 'column_82', 'column_83', 'column_84',
'column_85', 'column_86', 'column_87', 'column_88', 'column_89',
'column_90', 'column_91', 'column_92', 'column_93', 'column_94',
'column_95', 'column_96', 'column_97', 'column_98', 'column_99',
'column_100', 'column_101'}]);

% Extract predictors and response
% This code processes the data into the right shape for training the
% classifier.
predictorNames = {'column_1', 'column_2', 'column_3', 'column_4',
'column_5', 'column_6', 'column_7', 'column_8', 'column_9',
'column_10', 'column_11', 'column_12', 'column_13', 'column_14',
'column_15', 'column_16', 'column_17', 'column_18', 'column_19',
'column_20', 'column_21', 'column_22', 'column_23', 'column_24',
'column_25', 'column_26', 'column_27', 'column_28', 'column_29',
'column_30', 'column_31', 'column_32', 'column_33', 'column_34',
'column_35', 'column_36', 'column_37', 'column_38', 'column_39',
'column_40', 'column_41', 'column_42', 'column_43', 'column_44',
'column_45', 'column_46', 'column_47', 'column_48', 'column_49',
'column_50', 'column_51', 'column_52', 'column_53', 'column_54',
'column_55', 'column_56', 'column_57', 'column_58', 'column_59',
'column_60', 'column_61', 'column_62', 'column_63', 'column_64',
'column_65', 'column_66', 'column_67', 'column_68', 'column_69',
'column_70', 'column_71', 'column_72', 'column_73', 'column_74',
'column_75', 'column_76', 'column_77', 'column_78', 'column_79',
'column_80', 'column_81', 'column_82', 'column_83', 'column_84',
'column_85', 'column_86', 'column_87', 'column_88', 'column_89',
'column_90', 'column_91', 'column_92', 'column_93', 'column_94',
'column_95', 'column_96', 'column_97', 'column_98', 'column_99',
'column_100'};
predictors = inputTable(:, predictorNames);
response = inputTable.column_101;

% Train a classifier
% This code specifies all the classifier options and trains the
classifier.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Function of KNN %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
classificationKNN = fitcknn(...
predictors, ...
response, ...
'Distance', 'Euclidean', ...
'Exponent', [], ...
'NumNeighbors', 1, ...
'DistanceWeight', 'equal', ...
'Standardize', true, ...
'ClassNames', [1; 2; 3; 4; 5]);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Function of SVM classifier %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

template = templateSVM(...
    'KernelFunction', 'polynomial', ...
    'PolynomialOrder', 2, ...
    'KernelScale', 'auto', ...
    'BoxConstraint', 1, ...
    'Standardize', true);
classificationSVM = fitcecoc(...
    predictors, ...
    response, ...
    'Learners', template, ...
    'Coding', 'onevsone', ...
    'ClassNames', [1; 2; 3; 4; 5]);

%Using one of the Coding techniques:'onevsone', 'allpairs',
'onevsall', 'binarycomplete', 'ternarycomplete', 'ordinal',
'sparserandom', or 'denserandom'

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                    %%%%%%%%%%%%%%% Function of QDA classifier %%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
classificationDiscriminant = fitcdiscr(...
    predictors, ...
    response, ...
    'DiscrimType', 'diagQuadratic', ...
    'FillCoeffs', 'off', ...
    'SaveMemory', 'on', ...
    'ClassNames', [1; 2; 3; 4; 5]);

%%%%%%%%%% Results of three classifiers %%%%%%%%%%%
trainedClassifier1.ClassificationDiscriminant =
classificationDiscriminant;

trainedClassifier2.ClassificationKNN = classificationKNN;

trainedClassifier3.ClassificationSVM = classificationSVM;

convertMatrixToTableFcn = @(x) table(x, 'VariableNames',
{'column'});
splitMatricesInTableFcn = @(t)
[t(:,setdiff(t.Properties.VariableNames, {'column'})),
array2table(table2array(t(:,{'column'})), 'VariableNames',
{'column_1', 'column_2', 'column_3', 'column_4', 'column_5',
'column_6', 'column_7', 'column_8', 'column_9', 'column_10',
'column_11', 'column_12', 'column_13', 'column_14', 'column_15',
'column_16', 'column_17', 'column_18', 'column_19', 'column_20',
'column_21', 'column_22', 'column_23', 'column_24', 'column_25',
'column_26', 'column_27', 'column_28', 'column_29', 'column_30',
'column_31', 'column_32', 'column_33', 'column_34', 'column_35',
'column_36', 'column_37', 'column_38', 'column_39', 'column_40',
'column_41', 'column_42', 'column_43', 'column_44', 'column_45',
'column_46', 'column_47', 'column_48', 'column_49', 'column_50',
'column_51', 'column_52', 'column_53', 'column_54', 'column_55',
'column_56', 'column_57', 'column_58', 'column_59', 'column_60',
'column_61', 'column_62', 'column_63', 'column_64', 'column_65',
'column_66', 'column_67', 'column_68', 'column_69', 'column_70',
'column_71', 'column_72', 'column_73', 'column_74', 'column_75',
'column_76', 'column_77', 'column_78', 'column_79', 'column_80',
'column_81', 'column_82', 'column_83', 'column_84', 'column_85',
'column_86', 'column_87', 'column_88', 'column_89', 'column_90',

```

## Appendices

---

```
'column_91', 'column_92', 'column_93', 'column_94', 'column_95',
'column_96', 'column_97', 'column_98', 'column_99', 'column_100'}]);
extractPredictorsFromTableFcn = @(t) t(:, predictorNames);
predictorExtractionFcn = @(x)
extractPredictorsFromTableFcn(splitMatricesInTableFcn(convertMatrixT
oTableFcn(x)));
discriminantPredictFcn = @(x) predict(classificationDiscriminant,
x);
trainedClassifier1.predictFcn = @(x)
discriminantPredictFcn(predictorExtractionFcn(x));

knnPredictFcn = @(x) predict(classificationKNN, x);
trainedClassifier2.predictFcn = @(x)
knnPredictFcn(predictorExtractionFcn(x));

svmPredictFcn = @(x) predict(classificationSVM, x);
trainedClassifier3.predictFcn = @(x)
svmPredictFcn(predictorExtractionFcn(x));

% Convert input to table
inputTable = table(trainingData);
inputTable.Properties.VariableNames = {'column'};

% Split matrices in the input table into vectors
inputTable =
[inputTable(:,setdiff(inputTable.Properties.VariableNames,
{'column'})), array2table(table2array(inputTable(:,{'column'})),
'VariableNames', {'column_1', 'column_2', 'column_3', 'column_4',
'column_5', 'column_6', 'column_7', 'column_8', 'column_9',
'column_10', 'column_11', 'column_12', 'column_13', 'column_14',
'column_15', 'column_16', 'column_17', 'column_18', 'column_19',
'column_20', 'column_21', 'column_22', 'column_23', 'column_24',
'column_25', 'column_26', 'column_27', 'column_28', 'column_29',
'column_30', 'column_31', 'column_32', 'column_33', 'column_34',
'column_35', 'column_36', 'column_37', 'column_38', 'column_39',
'column_40', 'column_41', 'column_42', 'column_43', 'column_44',
'column_45', 'column_46', 'column_47', 'column_48', 'column_49',
'column_50', 'column_51', 'column_52', 'column_53', 'column_54',
'column_55', 'column_56', 'column_57', 'column_58', 'column_59',
'column_60', 'column_61', 'column_62', 'column_63', 'column_64',
'column_65', 'column_66', 'column_67', 'column_68', 'column_69',
'column_70', 'column_71', 'column_72', 'column_73', 'column_74',
'column_75', 'column_76', 'column_77', 'column_78', 'column_79',
'column_80', 'column_81', 'column_82', 'column_83', 'column_84',
'column_85', 'column_86', 'column_87', 'column_88', 'column_89',
'column_90', 'column_91', 'column_92', 'column_93', 'column_94',
'column_95', 'column_96', 'column_97', 'column_98', 'column_99',
'column_100', 'column_101'})]];

% Extract predictors and response
% This code processes the data into the right shape for training the
% classifier.
predictorNames = {'column_1', 'column_2', 'column_3', 'column_4',
'column_5', 'column_6', 'column_7', 'column_8', 'column_9',
'column_10', 'column_11', 'column_12', 'column_13', 'column_14',
'column_15', 'column_16', 'column_17', 'column_18', 'column_19',
'column_20', 'column_21', 'column_22', 'column_23', 'column_24',
'column_25', 'column_26', 'column_27', 'column_28', 'column_29',
'column_30', 'column_31', 'column_32', 'column_33', 'column_34',
'column_35', 'column_36', 'column_37', 'column_38', 'column_39',
```

```
'column_40', 'column_41', 'column_42', 'column_43', 'column_44',
'column_45', 'column_46', 'column_47', 'column_48', 'column_49',
'column_50', 'column_51', 'column_52', 'column_53', 'column_54',
'column_55', 'column_56', 'column_57', 'column_58', 'column_59',
'column_60', 'column_61', 'column_62', 'column_63', 'column_64',
'column_65', 'column_66', 'column_67', 'column_68', 'column_69',
'column_70', 'column_71', 'column_72', 'column_73', 'column_74',
'column_75', 'column_76', 'column_77', 'column_78', 'column_79',
'column_80', 'column_81', 'column_82', 'column_83', 'column_84',
'column_85', 'column_86', 'column_87', 'column_88', 'column_89',
'column_90', 'column_91', 'column_92', 'column_93', 'column_94',
'column_95', 'column_96', 'column_97', 'column_98', 'column_99',
'column_100'};
predictors = inputTable(:, predictorNames);
response = inputTable.column_101;

% Perform cross-validation
partitionedModel =
crossval(trainedClassifier1.ClassificationDiscriminant, 'KFold', 5);
partitionedMode2 = crossval(trainedClassifier2.ClassificationKNN,
'KFold', 5);
partitionedMode3 = crossval(trainedClassifier3.ClassificationSVM,
'KFold', 5);

% Compute accuracy of QDA, KNN, and SVM
validationAccuracy = 1 - kfoldLoss(partitionedModel, 'LossFun',
'ClassifError');
Accuracy_QDA=validationAccuracy*100

validationAccuracy = 1 - kfoldLoss(partitionedMode2, 'LossFun',
'ClassifError');
Accuracy_KNN=validationAccuracy*100

validationAccuracy = 1 - kfoldLoss(partitionedMode3, 'LossFun',
'ClassifError');
Accuracy_SVM=validationAccuracy*100
```



# C

## **Matlab simulation code for Tunable Q-Factor WT**

A simulation code for classification two databases, epileptic EEG signals and focal and non-focal EEG signals, is presented.

C.I Matlab code for Tunable Q-Factor WT (TQWT).

## 1- Implementation of the first EEG database

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
/\%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Read Epileptic EEG %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%\
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

### Reading epileptic EEG data is same as in the appendices A and B

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% TQWT to classify the Epileptic EEG data %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
clear all
clc
close all
load Epilpeic;

%% Set example parameters
% High Q-factor wavelet transform parameters
    Q2 = 14;
    r2 = 3;
    J2 = 5;

%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Set A %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
x1=A;
[n m ]=size (x1);
for i=1:m
w1=tqwt_radix2(x1(:,i),Q2,r2,J2);
p1{i}.o=w1;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Plot class A %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%
fs = 1;
figure(1), clf
PlotSubbands(x1(:,1),w1,Q2,r2,1,J2,fs);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Extract feature using TQWT %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for i=1:100
    M1=p1{i};
    T11=M1.o{1,1}; T12=M1.o{1,2}; T13=M1.o{1,3}; T14=M1.o{1,4};
    T15=M1.o{1,5};

%%
    Fea_new11{i}=T11; Fea_new12{i}=T12; Fea_new13{i}=T13;
    Fea_new14{i}=T14; Fea_new15{i}=T15;
end

%%
for j=1:100
    TT=Fea_new11{j}; WQ11(:,j)=TT; TT=[];
    TT1=Fea_new12{j}; WQ12(:,j)=TT1; TT1=[];
    TT2=Fea_new13{j}; WQ13(:,j)=TT2; TT2=[];
    TT3=Fea_new14{j}; WQ14(:,j)=TT3; TT3=[];
    TT4=Fea_new15{j}; WQ15(:,j)=TT4; TT4=[];
end

%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Feature extraction from A set %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```



## Appendices

---

```
%%%%%%%%%% Sub-band 1 %%%%%%%%%%%
v1=WQ11(:,1:25);
v2=WQ11(:,26:50);
v3=WQ11(:,51:75);
v4=WQ11(:,76:100);

for k=1:2048
    fs11(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,:)),
    median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1(k,:)),
    skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:2048

    fs12(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,:)),
    median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2(k,:)),
    skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:2048

    fs13(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,:)),
    median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3(k,:)),
    skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:2048

    fs14(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,:)),
    median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4(k,:)),
    skewness(v4(k,:)),var(v4(k,:))];
end

F_A_Band1=[fs11 fs12 fs13 fs14];
v1=[]; v2=[]; v3=[]; v4=[];
fs11=[]; fs12=[]; fs13=[]; fs14=[];
%%
%%%%%%%%%% Sub-band 2 %%%%%%%%%%%

v1=WQ12(:,1:25);
v2=WQ12(:,26:50);
v3=WQ12(:,51:75);
v4=WQ12(:,76:100);

for k=1:2048
    fs21(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,:)),
    median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1(k,:)),
    skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:2048
    fs22(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,:)),
    median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2(k,:)),
    skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:2048
    fs23(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,:)),
    median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3(k,:)),
    skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:2048
    fs24(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,:)),
    median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4(k,:)),
    skewness(v4(k,:)),var(v4(k,:))];
end
```

## Appendices

---

```
F_A_Band2=[fs21 fs22 fs23 fs24];
v1=[]; v2=[]; v3=[]; v4=[];
fs21=[]; fs22=[]; fs23=[]; fs24=[];
%%
%%%%%%%%%% Sub-band 3 %%%%%%%%%%%

v1=WQ13(:,1:25);
v2=WQ13(:,26:50);
v3=WQ13(:,51:75);
v4=WQ13(:,76:100);

for k=1:1024
    fs31(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:1024
    fs32(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
    (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:1024
    fs33(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:1024
    fs34(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_A_Band3=[fs31 fs32 fs33 fs34];
v1=[]; v2=[]; v3=[]; v4=[];
fs31=[]; fs32=[]; fs33=[]; fs34=[];
%%
%%%%%%%%%% Sub-band 4 %%%%%%%%%%%

v1=WQ14(:,1:25);
v2=WQ14(:,26:50);
v3=WQ14(:,51:75);
v4=WQ14(:,76:100);

for k=1:1024
    fs41(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:1024
    fs42(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
    (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:1024
    fs43(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:1024
```

## Appendices

---

```
        fs44(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
        :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
        (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_A_Band4=[fs41 fs42 fs43 fs44];
v1=[]; v2=[]; v3=[]; v4=[];
fs41=[]; fs42=[]; fs43=[]; fs44=[];
%%
%%%%%%%%%% Sub-band 5 %%%%%%%%%%%

v1=WQ15(:,1:25);
v2=WQ15(:,26:50);
v3=WQ15(:,51:75);
v4=WQ15(:,76:100);

for k=1:1024
    fs51(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:1024
    fs52(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
    (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:1024
    fs53(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:1024
    fs54(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_A_Band5=[fs51 fs52 fs53 fs54];
v1=[]; v2=[]; v3=[]; v4=[];
fs51=[]; fs52=[]; fs53=[]; fs54=[];

%%
%%%%%%%%%% Set B %%%%%%%%%%%
x2=B;
[n m ]=size (x2);
for i=1:m
    w2=tqwt_radix2(x2(:,i),Q2,r2,J2);
    p2{i}.o=w2;
end
%%%%%%%% Plot class B %%%%%%%%%
fs = 1;
figure(2), clf
PlotSubbands(x2(:,1),w2,Q2,r2,1,J2,fs);

%%
for i=1:100
    M2=p2{i};
    T21=M2.o{1,1}; T22=M2.o{1,2}; T23=M2.o{1,3}; T24=M2.o{1,4};
    T25=M2.o{1,5};
```

## Appendices

---

```
    Fea_new21{i}=T21; Fea_new22{i}=T22; Fea_new23{i}=T23;
Fea_new24{i}=T24; Fea_new25{i}=T25;
end
for j=1:100
    TT=Fea_new21{j}; WQ21(:,j)=TT; TT=[];
    TT1=Fea_new22{j}; WQ22(:,j)=TT1; TT1=[];
    TT2=Fea_new23{j}; WQ23(:,j)=TT2; TT2=[];
    TT3=Fea_new24{j}; WQ24(:,j)=TT3; TT3=[];
    TT4=Fea_new25{j}; WQ25(:,j)=TT4; TT4=[];
end
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                %%%%%%%%%%%%%%% Feature extraction from B set %%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%% Sub-band 1 %%%%%%%%%%%%%%%

v1=WQ21(:,1:25);
v2=WQ21(:,26:50);
v3=WQ21(:,51:75);
v4=WQ21(:,76:100);

for k=1:2048
    fs11(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:2048
    fs12(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
    (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:2048
    fs13(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:2048
    fs14(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_B_Band1=[fs11 fs12 fs13 fs14];
v1=[]; v2=[]; v3=[]; v4=[];
fs11=[]; fs12=[]; fs13=[]; fs14=[];
%%
%%%%%%%%%%%%%% Sub-band 2 %%%%%%%%%%%%%%%

v1=WQ22(:,1:25);
v2=WQ22(:,26:50);
v3=WQ22(:,51:75);
v4=WQ22(:,76:100);

for k=1:2048
    fs21(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:2048
```

## Appendices

---

```
        fs22(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
        :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
        (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:2048
    fs23(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:2048
    fs24(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_B_Band2=[fs21 fs22 fs23 fs24];
v1=[]; v2=[]; v3=[]; v4=[];
fs21=[]; fs22=[]; fs23=[]; fs24=[];
%%
%%%%%%%%%% Sub-band 3 %%%%%%%%%%%

v1=WQ23(:,1:25);
v2=WQ23(:,26:50);
v3=WQ23(:,51:75);
v4=WQ23(:,76:100);

for k=1:1024
    fs31(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:1024
    fs32(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
    (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:1024
    fs33(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:1024
    fs34(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_B_Band3=[fs31 fs32 fs33 fs34];
v1=[]; v2=[]; v3=[]; v4=[];
fs31=[]; fs32=[]; fs33=[]; fs34=[];
%%
%%%%%%%%%% Sub-band 4 %%%%%%%%%%%

v1=WQ24(:,1:25);
v2=WQ24(:,26:50);
v3=WQ24(:,51:75);
v4=WQ24(:,76:100);

for k=1:1024
```

## Appendices

---

```
        fs41(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
        :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
        (k,:)),skewness(v1(k,:)),var(v1(k,:))];
    end
    for k=1:1024
        fs42(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
        :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
        (k,:)),skewness(v2(k,:)),var(v2(k,:))];
    end
    for k=1:1024
        fs43(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
        :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
        (k,:)),skewness(v3(k,:)),var(v3(k,:))];
    end
    for k=1:1024
        fs44(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
        :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
        (k,:)),skewness(v4(k,:)),var(v4(k,:))];
    end

    F_B_Band4=[fs41 fs42 fs43 fs44];
    v1=[]; v2=[]; v3=[]; v4=[];
    fs41=[]; fs42=[]; fs43=[]; fs44=[];
    %%
    %%%%%%%%%%% Sub-band 5 %%%%%%%%%%%

    v1=WQ25(:,1:25);
    v2=WQ25(:,26:50);
    v3=WQ25(:,51:75);
    v4=WQ25(:,76:100);

    for k=1:1024
        fs51(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
        :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
        (k,:)),skewness(v1(k,:)),var(v1(k,:))];
    end
    for k=1:1024
        fs52(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
        :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
        (k,:)),skewness(v2(k,:)),var(v2(k,:))];
    end
    for k=1:1024
        fs53(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
        :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
        (k,:)),skewness(v3(k,:)),var(v3(k,:))];
    end
    for k=1:1024
        fs54(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
        :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
        (k,:)),skewness(v4(k,:)),var(v4(k,:))];
    end

    F_B_Band5=[fs51 fs52 fs53 fs54];
    v1=[]; v2=[]; v3=[]; v4=[];
    fs51=[]; fs52=[]; fs53=[]; fs54=[];

    %%
    %%%%%%%%%%% Set C %%%%%%%%%%%
    x3=C;
    [n m ]=size (x3);
```

## Appendices

---

```
for i=1:m
    w3=tqwt_radix2(x3(:,i),Q2,r2,J2);
    p3{i}.o=w3;
end
%%%%%%%% Plot class C %%%%%%%%%
fs = 1;
figure(3), clf
PlotSubbands(x3(:,1),w3,Q2,r2,1,J2,fs);

%%
for i=1:100
    M3=p3{i};
    T31=M3.o{1,1}; T32=M3.o{1,2}; T33=M3.o{1,3}; T34=M3.o{1,4};
    T35=M3.o{1,5};
    Fea_new31{i}=T31; Fea_new32{i}=T32; Fea_new33{i}=T33;
    Fea_new34{i}=T34; Fea_new35{i}=T35;
end
for j=1:100
    TT=Fea_new31{j}; WQ31(:,j)=TT; TT=[];
    TT1=Fea_new32{j}; WQ32(:,j)=TT1; TT1=[];
    TT2=Fea_new33{j}; WQ33(:,j)=TT2; TT2=[];
    TT3=Fea_new34{j}; WQ34(:,j)=TT3; TT3=[];
    TT4=Fea_new35{j}; WQ35(:,j)=TT4; TT4=[];
end
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                %%%%%%%%% Feature extraction from C set %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%% Sub-band 1 %%%%%%%%%

v1=WQ31(:,1:25);
v2=WQ31(:,26:50);
v3=WQ31(:,51:75);
v4=WQ31(:,76:100);

for k=1:2048
    fs11(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:2048
    fs12(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
    (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:2048
    fs13(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:2048
    fs14(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_C_Band1=[fs11 fs12 fs13 fs14];
v1=[]; v2=[]; v3=[]; v4=[];
fs11=[]; fs12=[]; fs13=[]; fs14=[];
%%
```

## Appendices

---

%%%%%%%% Sub-band 2 %%%%%%%%%

```
v1=WQ32(:,1:25);
v2=WQ32(:,26:50);
v3=WQ32(:,51:75);
v4=WQ32(:,76:100);

for k=1:2048
    fs21(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:2048
    fs22(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
    (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:2048
    fs23(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:2048
    fs24(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end
```

```
F_C_Band2=[fs21 fs22 fs23 fs24];
v1=[]; v2=[]; v3=[]; v4=[];
fs21=[]; fs22=[]; fs23=[]; fs24=[];
%%
```

%%%%%%%% Sub-band 3 %%%%%%%%%

```
v1=WQ33(:,1:25);
v2=WQ33(:,26:50);
v3=WQ33(:,51:75);
v4=WQ33(:,76:100);

for k=1:1024
    fs31(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:1024
    fs32(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
    (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:1024
    fs33(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:1024
    fs34(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end
```



## Appendices

---

```
F_C_Band3=[fs31 fs32 fs33 fs34];
v1=[]; v2=[]; v3=[]; v4=[];
fs31=[]; fs32=[]; fs33=[]; fs34=[];
%%
%%%%%%%%%% Sub-band 4 %%%%%%%%%%%

v1=WQ34(:,1:25);
v2=WQ34(:,26:50);
v3=WQ34(:,51:75);
v4=WQ34(:,76:100);

for k=1:1024
    fs41(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:1024
    fs42(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
    (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:1024
    fs43(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:1024
    fs44(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_C_Band4=[fs41 fs42 fs43 fs44];
v1=[]; v2=[]; v3=[]; v4=[];
fs41=[]; fs42=[]; fs43=[]; fs44=[];
%%
%%%%%%%%%% Sub-band 5 %%%%%%%%%%%

v1=WQ35(:,1:25);
v2=WQ35(:,26:50);
v3=WQ35(:,51:75);
v4=WQ35(:,76:100);

for k=1:1024
    fs51(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:1024
    fs52(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
    (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:1024
    fs53(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:1024
```

## Appendices

---

```
        fs54(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
        :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
        (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_C_Band5=[fs51 fs52 fs53 fs54];
v1=[]; v2=[]; v3=[]; v4=[];
fs51=[]; fs52=[]; fs53=[]; fs54=[];
%%

%%%%%%%%%% set D %%%%%%%%%%%
x4=D;
[n m ]=size (x4);
for i=1:m
w4=tqwt_radix2(x4(:,i),Q2,r2,J2);
p4{i}.o=w4;
end
%%%%%%%%%% Plot class D %%%%%%%%%%%
fs = 1;
figure(4), clf
PlotSubbands(x4(:,1),w4,Q2,r2,1,J2,fs);
%%
for i=1:100
    M4=p4{i};
    T41=M4.o{1,1}; T42=M4.o{1,2}; T43=M4.o{1,3}; T44=M4.o{1,4};
    T45=M4.o{1,5};
    Fea_new41{i}=T41; Fea_new42{i}=T42; Fea_new43{i}=T43;
    Fea_new44{i}=T44; Fea_new45{i}=T45;
end
for j=1:100
    TT=Fea_new41{j}; WQ41(:,j)=TT; TT=[];
    TT1=Fea_new42{j}; WQ42(:,j)=TT1; TT1=[];
    TT2=Fea_new43{j}; WQ43(:,j)=TT2; TT2=[];
    TT3=Fea_new44{j}; WQ44(:,j)=TT3; TT3=[];
    TT4=Fea_new45{j}; WQ45(:,j)=TT4; TT4=[];
end
%%
%%%%%%%%%% Feature extraction from D set %%%%%%%%%%%
%%%%%%%%%% Sub-band 1 %%%%%%%%%%%

v1=WQ41(:,1:25);
v2=WQ41(:,26:50);
v3=WQ41(:,51:75);
v4=WQ41(:,76:100);

for k=1:2048
    fs11(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:2048
    fs12(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
    (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:2048
```

## Appendices

---

```
        fs13(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
        :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
        (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:2048
    fs14(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end
F_D_Band1=[fs11 fs12 fs13 fs14];
v1=[]; v2=[]; v3=[]; v4=[];
fs11=[]; fs12=[]; fs13=[]; fs14=[];
%%
%%%%%%%%%% Sub-band 2 %%%%%%%%%%%

v1=WQ42(:,1:25);
v2=WQ42(:,26:50);
v3=WQ42(:,51:75);
v4=WQ42(:,76:100);

for k=1:2048
    fs21(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:2048
    fs22(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
    (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:2048
    fs23(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:2048
    fs24(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_D_Band2=[fs21 fs22 fs23 fs24];
v1=[]; v2=[]; v3=[]; v4=[];
fs21=[]; fs22=[]; fs23=[]; fs24=[];
%%
%%%%%%%%%% Sub-band 3 %%%%%%%%%%%

v1=WQ43(:,1:25);
v2=WQ43(:,26:50);
v3=WQ43(:,51:75);
v4=WQ43(:,76:100);

for k=1:1024
    fs31(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:1024
```

## Appendices

---

```
        fs32(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
        :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
        (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:1024
    fs33(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:1024
    fs34(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_D_Band3=[fs31 fs32 fs33 fs34];
v1=[]; v2=[]; v3=[]; v4=[];
fs31=[]; fs32=[]; fs33=[]; fs34=[];
%%
%%%%%%%%%% Sub-band 4 %%%%%%%%%%%

v1=WQ44(:,1:25);
v2=WQ44(:,26:50);
v3=WQ44(:,51:75);
v4=WQ44(:,76:100);

for k=1:1024
    fs41(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:1024
    fs42(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
    (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:1024
    fs43(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:1024
    fs44(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_D_Band4=[fs41 fs42 fs43 fs44];
v1=[]; v2=[]; v3=[]; v4=[];
fs41=[]; fs42=[]; fs43=[]; fs44=[];
%%
%%%%%%%%%% Sub-band 5 %%%%%%%%%%%

v1=WQ45(:,1:25);
v2=WQ45(:,26:50);
v3=WQ45(:,51:75);
v4=WQ45(:,76:100);

for k=1:1024
```

## Appendices

---

```
fs51(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
:)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
(k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:1024
fs52(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
:)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
(k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:1024
fs53(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
:)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
(k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:1024
fs54(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
:)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
(k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_D_Band5=[fs51 fs52 fs53 fs54];
v1=[]; v2=[]; v3=[]; v4=[];
fs51=[]; fs52=[]; fs53=[]; fs54=[];
%%
%%%%%%%%%% set E %%%%%%%%%%%
x5=E; % one set form the datasets (A, B, C, D, E)
[n m ]=size (x5);
for i=1:m
w5=tqwt_radix2(x5(:,i),Q2,r2,J2);
p5{i}.o=w5;
end
%%%%%%%%%% Plot class E %%%%%%%%%%%
fs = 1;
figure(5), clf
PlotSubbands(x1(:,1),w5,Q2,r2,1,J2,fs);
%%
for i=1:100
M5=p5{i};
T51=M5.o{1,1}; T52=M5.o{1,2}; T53=M5.o{1,3}; T54=M5.o{1,4};
T55=M5.o{1,5};
Fea_new51{i}=T51; Fea_new52{i}=T52; Fea_new53{i}=T53;
Fea_new54{i}=T54; Fea_new55{i}=T55;
end
for j=1:100
TT=Fea_new51{j}; WQ51(:,j)=TT; TT=[];
TT1=Fea_new52{j}; WQ52(:,j)=TT1; TT1=[];
TT2=Fea_new53{j}; WQ53(:,j)=TT2; TT2=[];
TT3=Fea_new54{j}; WQ54(:,j)=TT3; TT3=[];
TT4=Fea_new55{j}; WQ55(:,j)=TT4; TT4=[];
end
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%% Feature extraction from E set %%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%% Sub-band 1 %%%%%%%%%%%

v1=WQ51(:,1:25);
v2=WQ51(:,26:50);
v3=WQ51(:,51:75);
v4=WQ51(:,76:100);
```

```

for k=1:2048
    fs11(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,:)),
    median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1(k,
    :)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:2048
    fs12(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,:)),
    median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2(k,
    :)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:2048
    fs13(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:2048
    fs14(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_E_Band1=[fs11 fs12 fs13 fs14];
v1=[]; v2=[]; v3=[]; v4=[];
fs11=[]; fs12=[]; fs13=[]; fs14=[];
%%
%%%%%%%%%% Sub-band 2 %%%%%%%%%%%

v1=WQ52(:,1:25);
v2=WQ52(:,26:50);
v3=WQ52(:,51:75);
v4=WQ52(:,76:100);

for k=1:2048
    fs21(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:2048
    fs22(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
    (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:2048
    fs23(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:2048
    fs24(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_E_Band2=[fs21 fs22 fs23 fs24];
v1=[]; v2=[]; v3=[]; v4=[];
fs21=[]; fs22=[]; fs23=[]; fs24=[];
%%
%%%%%%%%%% Sub-band 3 %%%%%%%%%%%

```

## Appendices

---

```
v1=WQ53(:,1:25);
v2=WQ53(:,26:50);
v3=WQ53(:,51:75);
v4=WQ53(:,76:100);

for k=1:1024
    fs31(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
        :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
        (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:1024
    fs32(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
        :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
        (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:1024
    fs33(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
        :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
        (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:1024
    fs34(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
        :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
        (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_E_Band3=[fs31 fs32 fs33 fs34];
v1=[]; v2=[]; v3=[]; v4=[];
fs31=[]; fs32=[]; fs33=[]; fs34=[];
%%
%%%%%%%%%% Sub-band 4 %%%%%%%%%%%

v1=WQ54(:,1:25);
v2=WQ54(:,26:50);
v3=WQ54(:,51:75);
v4=WQ54(:,76:100);

for k=1:1024
    fs41(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
        :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
        (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:1024
    fs42(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
        :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
        (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:1024
    fs43(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
        :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
        (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:1024
    fs44(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
        :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
        (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_E_Band4=[fs41 fs42 fs43 fs44];
v1=[]; v2=[]; v3=[]; v4=[];
```

## Appendices

---

```
fs41=[]; fs42=[]; fs43=[]; fs44=[];
%%
%%%%%%%%%% Sub-band 5 %%%%%%%%%%%

v1=WQ55(:,1:25);
v2=WQ55(:,26:50);
v3=WQ55(:,51:75);
v4=WQ55(:,76:100);

for k=1:1024
    fs51(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:1024
    fs52(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
    (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:1024
    fs53(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:1024
    fs54(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_E_Band5=[fs51 fs52 fs53 fs54];
v1=[]; v2=[]; v3=[]; v4=[];
fs51=[]; fs52=[]; fs53=[]; fs54=[];

%%
%%%%%%%%%% class label %%%%%%%%%%%
f11(1:2048,1)=[1]; f12(1:1024,1)=[1];
f21(1:2048,1)=[2]; f22(1:1024,1)=[2];
f31(1:2048,1)=[3]; f32(1:1024,1)=[3];
f41(1:2048,1)=[4]; f42(1:1024,1)=[4];
f51(1:2048,1)=[5]; f52(1:1024,1)=[5];

A_SBL1=[F_A_Band1,f11]; A_SBL2=[F_A_Band2,f11];
A_SBL3=[F_A_Band3,f12]; A_SBL4=[F_A_Band4,f12];
A_SBL5=[F_A_Band5,f12];
B_SBL1=[F_B_Band1,f21]; B_SBL2=[F_B_Band2,f21];
B_SBL3=[F_B_Band3,f22]; B_SBL4=[F_B_Band4,f22];
B_SBL5=[F_B_Band5,f22];
B_SBL11=[F_B_Band1,f11]; B_SBL22=[F_B_Band2,f11];
B_SBL33=[F_B_Band3,f12]; B_SBL44=[F_B_Band4,f12];
B_SBL55=[F_B_Band5,f12];
C_SBL1=[F_C_Band1,f31]; C_SBL2=[F_C_Band2,f31];
C_SBL3=[F_C_Band3,f32]; C_SBL4=[F_C_Band4,f32];
C_SBL5=[F_C_Band5,f32];
C_SBL11=[F_C_Band1,f51]; C_SBL22=[F_C_Band2,f51];
C_SBL33=[F_C_Band3,f52]; C_SBL44=[F_C_Band4,f52];
C_SBL55=[F_C_Band5,f52];
D_SBL1=[F_D_Band1,f41]; D_SBL2=[F_D_Band2,f41];
D_SBL3=[F_D_Band3,f42]; D_SBL4=[F_D_Band4,f42];
D_SBL5=[F_D_Band5,f42];
```



## Appendices

---

```
D_SBL11=[F_D_Band1,f51]; D_SBL22=[F_D_Band2,f51];
D_SBL33=[F_D_Band3,f52]; D_SBL44=[F_D_Band4,f52];
D_SBL55=[F_D_Band5,f52];
E_SBL1=[F_E_Band1,f51]; E_SBL2=[F_E_Band2,f51];
E_SBL3=[F_E_Band3,f52]; E_SBL4=[F_E_Band4,f52];
E_SBL5=[F_E_Band5,f52];

%%
%%%%%%%%%% cases used in this scheme %%%%%%%%%%%
case11=[A_SBL1;E_SBL1]; case12=[A_SBL2;E_SBL2];
case13=[A_SBL3;E_SBL3]; case14=[A_SBL4;E_SBL4];
case15=[A_SBL5;E_SBL5];
case21=[A_SBL1;C_SBL1]; case22=[A_SBL2;C_SBL2];
case23=[A_SBL3;C_SBL3]; case24=[A_SBL4;C_SBL4];
case25=[A_SBL5;C_SBL5];
case31=[B_SBL1;E_SBL1]; case32=[B_SBL2;E_SBL2];
case33=[B_SBL3;E_SBL3]; case34=[B_SBL4;E_SBL4];
case35=[B_SBL5;E_SBL5];
case41=[C_SBL1;E_SBL1]; case42=[C_SBL2;E_SBL2];
case43=[C_SBL3;E_SBL3]; case44=[C_SBL4;E_SBL4];
case45=[C_SBL5;E_SBL5];
case51=[D_SBL1;E_SBL1]; case52=[D_SBL2;E_SBL2];
case53=[D_SBL3;E_SBL3]; case54=[D_SBL4;E_SBL4];
case55=[D_SBL5;E_SBL5];
case61=[A_SBL1;B_SBL11;E_SBL1]; case62=[A_SBL2;B_SBL22;E_SBL2];
case63=[A_SBL3;B_SBL33;E_SBL3]; case64=[A_SBL4;B_SBL44;E_SBL4];
case65=[A_SBL5;B_SBL55;E_SBL5];
case71=[A_SBL1;C_SBL11;D_SBL11;E_SBL1];
case72=[A_SBL2;C_SBL22;D_SBL22;E_SBL2];
case73=[A_SBL3;C_SBL33;D_SBL33;E_SBL3];
case74=[A_SBL4;C_SBL44;D_SBL44;E_SBL4];
case75=[A_SBL5;C_SBL55;D_SBL55;E_SBL5];

%%
%%%%%%%%%% 5 fold cross validation %%%%%%%%%%%
%% for cases 1,2,3,4, and 5

% X=[case11]; % cross validation for subcase 1 and 2
%
% a1=[X(1:409,:);X(2049:2457,:)];
% a2=[X(410:818,:);X(2458:2866,:)];
% a3=[X(819:1227,:);X(2867:3275,:)];
% a4=[X(1228:1636,:);X(3276:3684,:)];
% a5=[X(1637:2048,:);X(3685:4096,:)];
% x=[a1;a3;a4;a5];
% Xtest=[a2];

% X=[case13]; % cross validation for subcase 3, 4 and 5
%
% a1=[X(1:204,:);X(1025:1228,:)];
% a2=[X(205:408,:);X(1229:1432,:)];
% a3=[X(409:612,:);X(1433:1636,:)];
% a4=[X(613:816,:);X(1637:1840,:)];
% a5=[X(817:1024,:);X(1841:2048,:)];
% x=[a1;a3;a4;a5];
% Xtest=[a2];
```

## Appendices

---

```
%%
% % for case 6

% X=[case61]; % cross validation for subcase 1 and 2
%
% a1=[X(1:819,:);X(4097:4505,:)];
% a2=[X(820:1638,:);X(4506:4914,:)];
% a3=[X(1639:2457,:);X(4915:5323,:)];
% a4=[X(2458:3276,:);X(5324:5732,:)];
% a5=[X(3277:4096,:);X(5733:6144,:)];
% x=[a2;a3;a4;a5];
% Xtest=[a1];

% X=[case63]; % cross validation for subcase 3, 4 and 5
%
% a1=[X(1:409,:);X(2049:2252,:)];
% a2=[X(410:818,:);X(2253:2456,:)];
% a3=[X(819:1227,:);X(2457:2660,:)];
% a4=[X(1228:1636,:);X(2661:2864,:)];
% a5=[X(1637:2048,:);X(2865:3072,:)];
% x=[a1;a3;a4;a5];
% Xtest=[a2];

%%
% for case 7
%
% X=[case71]; % cross validation for subcase 1 and 2
%
% a1=[X(1:1228,:);X(6145:6553,:)];
% a2=[X(1229:2456,:);X(6554:6962,:)];
% a3=[X(2457:3684,:);X(6963:7371,:)];
% a4=[X(3685:4912,:);X(7372:7780,:)];
% a5=[X(4913:6144,:);X(7781:8192,:)];
% x=[a2;a3;a4;a5];
% Xtest=[a1];

X=[case73]; % cross validation for subcase 3, 4 and 5

a1=[X(1:614,:);X(3073:3276,:)];
a2=[X(615:1228,:);X(3277:3480,:)];
a3=[X(1229:1842,:);X(3481:3684,:)];
a4=[X(1843:2456,:);X(3685:3888,:)];
a5=[X(2457:3072,:);X(3889:4096,:)];
x=[a1;a2;a4;a5];
Xtest=[a3];

%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%% Classification %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

inputTable = array2table(Xtest, 'VariableNames', {'column_1',
'column_2', 'column_3', 'column_4', 'column_5', 'column_6',
'column_7', 'column_8', 'column_9', 'column_10', 'column_11',
'column_12', 'column_13', 'column_14', 'column_15', 'column_16',
'column_17', 'column_18', 'column_19', 'column_20', 'column_21',
'column_22', 'column_23', 'column_24', 'column_25', 'column_26',
'column_27', 'column_28', 'column_29', 'column_30', 'column_31',
'column_32', 'column_33', 'column_34', 'column_35', 'column_36',
'column_37', 'column_38', 'column_39', 'column_40', 'column_41'});
```

```

predictorNames = {'column_1', 'column_2', 'column_3', 'column_4',
'column_5', 'column_6', 'column_7', 'column_8', 'column_9',
'column_10', 'column_11', 'column_12', 'column_13', 'column_14',
'column_15', 'column_16', 'column_17', 'column_18', 'column_19',
'column_20', 'column_21', 'column_22', 'column_23', 'column_24',
'column_25', 'column_26', 'column_27', 'column_28', 'column_29',
'column_30', 'column_31', 'column_32', 'column_33', 'column_34',
'column_35', 'column_36', 'column_37', 'column_38', 'column_39',
'column_40'};
predictors = inputTable(:, predictorNames);
response = inputTable.column_41;
isCategoricalPredictor = [false, false, false, false, false, false,
false, false, false, false, false, false, false, false, false,
false, false, false, false, false, false, false, false, false,
false, false, false, false, false, false, false, false, false,
false, false, false, false, false, false, false, false];

% Train a classifier
% This code specifies all the classifier options and trains the
classifier.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Function of KNN %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
classificationKNN = fitcknn(...
    predictors, ...
    response, ...
    'Distance', 'Euclidean', ...
    'Exponent', [], ...
    'NumNeighbors', 1, ...
    'DistanceWeight', 'Equal', ...
    'Standardize', true, ...
    'ClassNames', [1; 5]);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Function of SVM %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
classificationSVM = fitsvm(...
    predictors, ...
    response, ...
    'KernelFunction', 'polynomial', ...
    'PolynomialOrder', 2, ...
    'KernelScale', 'auto', ...
    'BoxConstraint', 1, ...
    'Standardize', true, ...
    'ClassNames', [1; 5]);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Function of BT %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
classificationEnsemble = fitensemble(...
    predictors, ...
    response, ...
    'Bag', ...
    30, ...
    'Tree', ...
    'Type', 'Classification', ...
    'ClassNames', [1; 5]);

% Create the result struct with predict function

```

```

predictorExtractionFcn = @(x) array2table(x, 'VariableNames',
predictorNames);
knnPredictFcn = @(x) predict(classificationKNN, x);
trainedClassifier.predictFcn = @(x)
knnPredictFcn(predictorExtractionFcn(x));
svmPredictFcn = @(x) predict(classificationSVM, x);
trainedClassifier.predictFcn = @(x)
svmPredictFcn(predictorExtractionFcn(x));
ensemblePredictFcn = @(x) predict(classificationEnsemble, x);
trainedClassifier.predictFcn = @(x)
ensemblePredictFcn(predictorExtractionFcn(x));

% Add additional fields to the result struct
trainedClassifier.ClassificationKNN = classificationKNN;
trainedClassifier.ClassificationSVM = classificationSVM;
trainedClassifier.ClassificationEnsemble = classificationEnsemble;

% Extract predictors and response
% This code processes the data into the right shape for training the
% classifier.
% Convert input to table
inputTable = array2table(x, 'VariableNames', {'column_1',
'column_2', 'column_3', 'column_4', 'column_5', 'column_6',
'column_7', 'column_8', 'column_9', 'column_10', 'column_11',
'column_12', 'column_13', 'column_14', 'column_15', 'column_16',
'column_17', 'column_18', 'column_19', 'column_20', 'column_21',
'column_22', 'column_23', 'column_24', 'column_25', 'column_26',
'column_27', 'column_28', 'column_29', 'column_30', 'column_31',
'column_32', 'column_33', 'column_34', 'column_35', 'column_36',
'column_37', 'column_38', 'column_39', 'column_40', 'column_41'});

predictorNames = {'column_1', 'column_2', 'column_3', 'column_4',
'column_5', 'column_6', 'column_7', 'column_8', 'column_9',
'column_10', 'column_11', 'column_12', 'column_13', 'column_14',
'column_15', 'column_16', 'column_17', 'column_18', 'column_19',
'column_20', 'column_21', 'column_22', 'column_23', 'column_24',
'column_25', 'column_26', 'column_27', 'column_28', 'column_29',
'column_30', 'column_31', 'column_32', 'column_33', 'column_34',
'column_35', 'column_36', 'column_37', 'column_38', 'column_39',
'column_40'};
predictors = inputTable(:, predictorNames);
response = inputTable.column_41;
isCategoricalPredictor = [false, false, false, false, false, false,
false, false, false, false, false, false, false, false, false,
false, false, false, false, false, false, false, false, false,
false, false, false, false, false, false, false, false, false,
false, false, false, false, false, false, false];

% Compute resubstitution accuracy
resubstitutionAccuracy1 = 1 -
resubLoss(trainedClassifier.ClassificationKNN, 'LossFun',
'ClassifError');
Accuracy_KNN=resubstitutionAccuracy1*100
resubstitutionAccuracy2 = 1 -
resubLoss(trainedClassifier.ClassificationSVM, 'LossFun',
'ClassifError');
Accuracy_SVM=resubstitutionAccuracy2*100
resubstitutionAccuracy3 = 1 -
resubLoss(trainedClassifier.ClassificationEnsemble, 'LossFun',
'ClassifError');
Accuracy_BT=resubstitutionAccuracy3*100

```

## 2- Implementation of the second EEG database

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                %%% Read Focal/Non-focal EEG data %%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clear all
clc
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Read data from 1 to 750 focal epilepsy %%%%%%%%%

B1=[];
for i=1:100
    N=[];
    if i<10
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_F_Ind_1_750 (1)\Data_F_Ind000' num2str(i)
            '.txt'];
    else
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_F_Ind_1_750 (1)\Data_F_Ind00' num2str(i)
            '.txt'];
    end
    if (i==100)
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_F_Ind_1_750 (1)\Data_F_Ind0' num2str(i)
            '.txt'];
    end;
    A = importdata(N);
    B1=[B1,A];
end
N=[];
i=[];
A=[];
for i=101:750
    N=[];
    if i<750
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_F_Ind_1_750 (1)\Data_F_Ind0' num2str(i)
            '.txt'];
    end
    if (i==750)
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_F_Ind_1_750 (1)\Data_F_Ind0' num2str(i)
            '.txt'];
    end;
    A = importdata(N);
    B1=[B1,A];
end
N=[];
i=[];
Focall=B1;
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Read data from 1 to 750 non_focal epilepsy %%%%%%%%%

```

## Appendices

---

```
C1=[];
for i=1:100
    N=[];
    if i<10
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_N_Ind_1_750\Data_N_Ind00' num2str(i)
            '.txt'];
    else
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_N_Ind_1_750\Data_N_Ind00' num2str(i)
            '.txt'];
    end
    if (i==100)
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_N_Ind_1_750\Data_N_Ind0' num2str(i) '.txt'];
    end;
    A = importdata(N);
    C1=[C1,A];
end
N=[];
i=[];
for i=101:750
    N=[];
    if i<750
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_N_Ind_1_750\Data_N_Ind0' num2str(i) '.txt'];
    end
    if (i==750)
        N=['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_N_Ind_1_750\Data_N_Ind0' num2str(i) '.txt'];
    end;
    A = importdata(N);
    C1=[C1,A];
end
N=[];
i=[];
A=[];
Non_Focall=C1;

%%%%%%%%%% Read data from 751 to 1500 focal epilepsy %%%%%%%%%%
B2=[];

for i=751:1500
    N=[];
    if i<1000
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_F_Ind_751_1500 (1)\Data_F_Ind0' num2str(i)
            '.txt'];
    else
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_F_Ind_751_1500 (1)\Data_F_Ind' num2str(i)
            '.txt'];
    end
end
```

```

    if (i==1500)
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_F_Ind_751_1500 (1)\Data_F_Ind' num2str(i)
            '.txt'];
    end;
A = importdata(N);
B2=[B2,A];
end
N=[];
i=[];
A=[];
Focal2=B2;
%%
%%%%%%%%% Read data from 751 to 1500 non_focal epilepsy %%%%%%%%%
C2=[];

for i=751:1500
    N=[];
    if i<1000
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_N_Ind_751_1500 (1)\Data_N_Ind0' num2str(i)
            '.txt'];
    else
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_N_Ind_751_1500 (1)\Data_N_Ind' num2str(i)
            '.txt'];
    end
    if (i==1500)
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_N_Ind_751_1500 (1)\Data_N_Ind' num2str(i)
            '.txt'];
    end;
A = importdata(N);
C2=[C2,A];
end
N=[];
i=[];
A=[];
Non_Focal2=C2;

%%%%%%%%% Read data from 1501 to 2250 focal epilepsy %%%%%%%%%
B3=[];

for i=1501:2250
    N=[];
    if i<2250
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_F_Ind_1501_2250\Data_F_Ind' num2str(i)
            '.txt'];
    end
    if (i==2250)
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_F_Ind_1501_2250\Data_F_Ind' num2str(i)
            '.txt'];
    end;

```

## Appendices

---

```
    end;
A = importdata(N);
B3=[B3,A];
end
N=[];
i=[];
A=[];
Focal3=B3;
%%
%%%%%%%%%% Read data from 1501 to 2250 non_focal epilepsy %%%%%%%%%%
C3=[];

for i=1501:2250
    N=[];
    if i<2250
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_N_Ind_1501_2250\Data_N_Ind' num2str(i)
            '.txt'];
    end
    if (i==2250)
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_N_Ind_1501_2250\Data_N_Ind' num2str(i)
            '.txt'];
    end;
    A = importdata(N);
    C3=[C3,A];
end
N=[];
i=[];
A=[];
Non_Focal3=C3;

%%%%%%%%%% Read data from 2251 to 3000 focal epilepsy %%%%%%%%%%
B4=[];

for i=2251:3000
    N=[];
    if i<3000
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_F_Ind_2251_3000\Data_F_Ind' num2str(i)
            '.txt'];
    end
    if (i==3000)
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_F_Ind_2251_3000\Data_F_Ind' num2str(i)
            '.txt'];
    end;
    A = importdata(N);
    B4=[B4,A];
end
N=[];
i=[];
A=[];
Focal4=B4;
%%
%%%%%%%%%% Read data from 2251 to 3000 non_focal epilepsy %%%%%%%%%%
```



## Appendices

---

```
C4=[];

for i=2251:3000
    N=[];
    if i<3000
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_N_Ind_2251_3000\Data_N_Ind' num2str(i)
            '.txt'];
    end
    if (i==3000)
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_N_Ind_2251_3000\Data_N_Ind' num2str(i)
            '.txt'];
    end;
    A = importdata(N);
    C4=[C4,A];
end
N=[];
i=[];
A=[];
Non_Focal4=C4;

%%%%%%%%%% Read data from 3001 to 3750 focal epilepsy %%%%%%%%%%
B5=[];

for i=3001:3750
    N=[];
    if i<3750
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_F_Ind_3001_3750\Data_F_Ind' num2str(i)
            '.txt'];
    end
    if (i==3750)
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_F_Ind_3001_3750\Data_F_Ind' num2str(i)
            '.txt'];
    end;
    A = importdata(N);
    B5=[B5,A];
end
N=[];
i=[];
A=[];
Focal5=B5;
%%
%%%%%%%%%% Read data from 3001 to 3750 non_focal epilepsy %%%%%%%%%%
C5=[];

for i=3001:3750
    N=[];
    if i<3750
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_N_Ind_3001_3750\Data_N_Ind' num2str(i)
            '.txt'];
    %
    else
    end
end
```

```

    if (i==3750)
        N= ['C:\Users\U1054787\Desktop\Papers\Fiveth
            Paper\Wavelet\tqwt_matlab_toolbox\F and NonF\New
            folder\Data_N_Ind_3001_3750\Data_N_Ind' num2str(i)
            '.txt'];
    end;
A = importdata(N);
C5=[C5,A];
end
N=[];
i=[];
A=[];
Non_Focal5=C5;

%%%%%%%%%%%% Whole read data of focal/no-focal EEG %%%%%%%%%%

All_Focal=[Focal1,Focal2,Focal3,Focal4,Focal5];
All_Non_Focal=[Non_Focal1,Non_Focal2,Non_Focal3,Non_Focal4,Non_Focal
5];

%%%%%%%%%%%%%% Using TQWT technique %%%%%%%%%%%%%%

% Set example parameters
% High Q-factor wavelet transform parameters

        Q2 = 6;
        r2 = 3;
        J2 = 5;

%%%%%%%%%%%%%% Focal Signals %%%%%%%%%%%%%%
x1=All_Focal;
[n m ]=size (x1);
for i=1:m
    w1=tqwt_radix2(x1(:,i),Q2,r2,J2);
    p1{i}.o=w1;
% p2{i}.o=w2;
End
%%
%%%%%%%%%%%%%% Plot Focal class %%%%%%%%%%%%%%
fs = 1;
figure(1), clf
PlotSubbands(x1(:,1),w1,Q2,r2,1,J2+1,fs);

%%
for i=1:m
    M1=p1{i};
    T11=M1.o{1,1}; T12=M1.o{1,2}; T13=M1.o{1,3}; T14=M1.o{1,4};
T15=M1.o{1,5};
% T16=M1.o{1,6};
    Fea_new11{i}=T11; Fea_new12{i}=T12; Fea_new13{i}=T13;
Fea_new14{i}=T14; Fea_new15{i}=T15;
% Fea_new15{i}=T16;
end
for j=1:m
    TT=Fea_new11{j}; WQ11(:,j)=TT; TT=[];
    TT1=Fea_new12{j}; WQ12(:,j)=TT1; TT1=[];
    TT2=Fea_new13{j}; WQ13(:,j)=TT2; TT2=[];
    TT3=Fea_new14{j}; WQ14(:,j)=TT3; TT3=[];

```

## Appendices

---

```
TT4=Fea_new15{j}; WQ15(:,j)=TT4; TT4=[];
end
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                %%%%%%%%% Feature extraction from facol %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%% Sub-band 1 %%%%%%%%%
v1=WQ11(:,1:1875);
v2=WQ11(:,1876:3750);
v3=WQ11(:,3751:5625);
v4=WQ11(:,5626:7500);

for k=1:8192
    fs11(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
        :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
        (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:8192
    fs12(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
        :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
        (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:8192
    fs13(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
        :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
        (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:8192
    fs14(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
        :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
        (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_A_Band1=[fs11 fs12 fs13 fs14];
v1=[]; v2=[]; v3=[]; v4=[];
fs11=[]; fs12=[]; fs13=[]; fs14=[];
%%
%%%%%%%% Sub-band 2 %%%%%%%%%

v1=WQ12(:,1:1875);
v2=WQ12(:,1876:3750);
v3=WQ12(:,3751:5625);
v4=WQ12(:,5626:7500);

for k=1:8192
    fs21(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
        :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
        (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:8192
    fs22(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
        :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
        (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:8192
    fs23(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
        :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
        (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:8192
```

## Appendices

---

```
        fs24(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
        :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
        (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_A_Band2=[fs21 fs22 fs23 fs24];
v1=[]; v2=[]; v3=[]; v4=[];
fs21=[]; fs22=[]; fs23=[]; fs24=[];
%%
%%%%%%%%%% Sub-band 3 %%%%%%%%%%%

v1=WQ13(:,1:1875);
v2=WQ13(:,1876:3750);
v3=WQ13(:,3751:5625);
v4=WQ13(:,5626:7500);

for k=1:4096
    fs31(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:4096
    fs32(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
    (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:4096
    fs33(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:4096
    fs34(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_A_Band3=[fs31 fs32 fs33 fs34];
v1=[]; v2=[]; v3=[]; v4=[];
fs31=[]; fs32=[]; fs33=[]; fs34=[];
%%
%%%%%%%%%% Sub-band 4 %%%%%%%%%%%

v1=WQ14(:,1:1875);
v2=WQ14(:,1876:3750);
v3=WQ14(:,3751:5625);
v4=WQ14(:,5626:7500);

for k=1:4096
    fs41(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:4096
    fs42(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
    (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:4096
```

## Appendices

---

```
        fs43(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
        :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
        (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:4096
    fs44(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_A_Band4=[fs41 fs42 fs43 fs44];
v1=[]; v2=[]; v3=[]; v4=[];
fs41=[]; fs42=[]; fs43=[]; fs44=[];
%%
%%%%%%%%%% Sub-band 5 %%%%%%%%%%%

v1=WQ15(:,1:1875);
v2=WQ15(:,1876:3750);
v3=WQ15(:,3751:5625);
v4=WQ15(:,5626:7500);

for k=1:4096
    fs51(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:4096
    fs52(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
    (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:4096
    fs53(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:4096
    fs54(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_A_Band5=[fs51 fs52 fs53 fs54];
v1=[]; v2=[]; v3=[]; v4=[];
fs51=[]; fs52=[]; fs53=[]; fs54=[];
%%
%%%%%%%%%% Non-focal Signals %%%%%%%%%%%

x2=All_Non_Focal;
[n m1 ]=size (x2);
for i=1:m1
    w2=tqwt_radix2(x2(:,i),Q2,r2,J2);
    p2{i}.o=w2;
end
%%
%%%%%%%%%% Plot Non-Focal class %%%%%%%%%%%
fs = 1;
figure(2), clf
PlotSubbands(x2(:,1),w2,Q2,r2,1,J2+1,fs);
```

## Appendices

---

```
%%
for i=1:m1
    M2=p2{i};
    T21=M2.o{1,1}; T22=M2.o{1,2}; T23=M2.o{1,3}; T24=M2.o{1,4};
    T25=M2.o{1,5};
    Fea_new21{i}=T21; Fea_new22{i}=T22; Fea_new23{i}=T23;
    Fea_new24{i}=T24; Fea_new25{i}=T25;
end
for j=1:m1
    TT=Fea_new21{j}; WQ21(:,j)=TT; TT=[];
    TT1=Fea_new22{j}; WQ22(:,j)=TT1; TT1=[];
    TT2=Fea_new23{j}; WQ23(:,j)=TT2; TT2=[];
    TT3=Fea_new24{j}; WQ24(:,j)=TT3; TT3=[];
    TT4=Fea_new25{j}; WQ25(:,j)=TT4; TT4=[];
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                %%%%%%%%%%% Feature extraction from Non-facol %%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%% Sub-band 1 %%%%%%%%%%%%%%%

v1=WQ21(:,1:1875);
v2=WQ21(:,1876:3750);
v3=WQ21(:,3751:5625);
v4=WQ21(:,5626:7500);

for k=1:8192
    fs11(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:8192
    fs12(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
    (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:8192
    fs13(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:8192
    fs14(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_B_Band1=[fs11 fs12 fs13 fs14];
v1=[]; v2=[]; v3=[]; v4=[];
fs11=[]; fs12=[]; fs13=[]; fs14=[];
%%
%%%%%%%%%%%%%% Sub-band 2 %%%%%%%%%%%%%%%

v1=WQ22(:,1:1875);
v2=WQ22(:,1876:3750);
v3=WQ22(:,3751:5625);
v4=WQ22(:,5626:7500);
```

```

for k=1:8192
    fs21(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:8192
    fs22(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
    (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:8192
    fs23(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:8192
    fs24(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_B_Band2=[fs21 fs22 fs23 fs24];
v1=[]; v2=[]; v3=[]; v4=[];
fs21=[]; fs22=[]; fs23=[]; fs24=[];
%%
%%%%%%%%%% Sub-band 3 %%%%%%%%%%%

v1=WQ23(:,1:1875);
v2=WQ23(:,1876:3750);
v3=WQ23(:,3751:5625);
v4=WQ23(:,5626:7500);

for k=1:4096
    fs31(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:4096
    fs32(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
    (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:4096
    fs33(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:4096
    fs34(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_B_Band3=[fs31 fs32 fs33 fs34];
v1=[]; v2=[]; v3=[]; v4=[];
fs31=[]; fs32=[]; fs33=[]; fs34=[];
%%
%%%%%%%%%% Sub-band 4 %%%%%%%%%%%

```

## Appendices

---

```
v1=WQ24(:,1:1875);
v2=WQ24(:,1876:3750);
v3=WQ24(:,3751:5625);
v4=WQ24(:,5626:7500);

for k=1:4096
    fs41(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:4096
    fs42(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
    (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:4096
    fs43(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:4096
    fs44(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_B_Band4=[fs41 fs42 fs43 fs44];
v1=[]; v2=[]; v3=[]; v4=[];
fs41=[]; fs42=[]; fs43=[]; fs44=[];
%%
%% Sub-band 5 %%

v1=WQ25(:,1:1875);
v2=WQ25(:,1876:3750);
v3=WQ25(:,3751:5625);
v4=WQ25(:,5626:7500);

for k=1:4096
    fs51(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),median(v1(k,:))/2,3*median(v1(k,:))/2,range(v1(k,:)),std(v1
    (k,:)),skewness(v1(k,:)),var(v1(k,:))];
end
for k=1:4096
    fs52(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),median(v2(k,:))/2,3*median(v2(k,:))/2,range(v2(k,:)),std(v2
    (k,:)),skewness(v2(k,:)),var(v2(k,:))];
end
for k=1:4096
    fs53(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),median(v3(k,:))/2,3*median(v3(k,:))/2,range(v3(k,:)),std(v3
    (k,:)),skewness(v3(k,:)),var(v3(k,:))];
end
for k=1:4096
    fs54(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),median(v4(k,:))/2,3*median(v4(k,:))/2,range(v4(k,:)),std(v4
    (k,:)),skewness(v4(k,:)),var(v4(k,:))];
end

F_B_Band5=[fs51 fs52 fs53 fs54];
v1=[]; v2=[]; v3=[]; v4=[];
```



## Appendices

---

```
fs51=[]; fs52=[]; fs53=[]; fs54=[];

%%
%%%%%%%%%%%%%% class label %%%%%%%%%%%%%%%
f1(1:8192,1)=[1]; f12(1:4096,1)=[1];
f2(1:8192,1)=[2]; f22(1:4096,1)=[2];

Focal_SBL1=[F_A_Band1,f1]; Focal_SBL2=[F_A_Band2,f1];
Focal_SBL3=[F_A_Band3,f12]; Focal_SBL4=[F_A_Band4,f12];
Focal_SBL5=[F_A_Band5,f12];
Non_SBL1=[F_B_Band1,f2]; Non_SBL2=[F_B_Band2,f2];
Non_SBL3=[F_B_Band3,f22]; Non_SBL4=[F_B_Band4,f22];
Non_SBL5=[F_B_Band5,f22];
%%
%%%%%%%%%%%%%% cases used in this scheme of each sub-band %%%%%%%%%%%%%%%
case1=[Focal_SBL1;Non_SBL1]; case2=[Focal_SBL2;Non_SBL2];
case3=[Focal_SBL3;Non_SBL3]; case4=[Focal_SBL4;Non_SBL4];
case5=[Focal_SBL5;Non_SBL5];

%%
%%%%%%%%%%%%%% 5 fold cross validation %%%%%%%%%%%%%%%
%% for cases 1,2,3,4, and 5
% %
% X=[case2]; % cross validation for subcase 1 and 2
%
% a1=[X(1:1638,:);X(8193:9830,:)];
% a2=[X(1639:3276,:);X(9831:11468,:)];
% a3=[X(3277:4914,:);X(11469:13106,:)];
% a4=[X(4915:6552,:);X(13107:14744,:)];
% a5=[X(6553:8192,:);X(14745:16384,:)];
% x=[a1;a3;a4;a5];
% Xtest=[a2];

X=[case5]; % cross validation for subcase 3, 4 and 5

a1=[X(1:819,:);X(4097:4915,:)];
a2=[X(820:1638,:);X(4916:5734,:)];
a3=[X(1639:2457,:);X(5735:6553,:)];
a4=[X(2458:3276,:);X(6554:7372,:)];
a5=[X(3277:4096,:);X(7373:8192,:)];
x=[a1;a3;a4;a5];
Xtest=[a2];

%%
%%%%%%%%%%%%%% Classification %%%%%%%%%%%%%%%

inputTable = array2table(Xtest, 'VariableNames', {'column_1',
'column_2', 'column_3', 'column_4', 'column_5', 'column_6',
'column_7', 'column_8', 'column_9', 'column_10', 'column_11',
'column_12', 'column_13', 'column_14', 'column_15', 'column_16',
'column_17', 'column_18', 'column_19', 'column_20', 'column_21',
'column_22', 'column_23', 'column_24', 'column_25', 'column_26',
'column_27', 'column_28', 'column_29', 'column_30', 'column_31',
```

```

'column_32', 'column_33', 'column_34', 'column_35', 'column_36',
'column_37', 'column_38', 'column_39', 'column_40', 'column_41'});

predictorNames = {'column_1', 'column_2', 'column_3', 'column_4',
'column_5', 'column_6', 'column_7', 'column_8', 'column_9',
'column_10', 'column_11', 'column_12', 'column_13', 'column_14',
'column_15', 'column_16', 'column_17', 'column_18', 'column_19',
'column_20', 'column_21', 'column_22', 'column_23', 'column_24',
'column_25', 'column_26', 'column_27', 'column_28', 'column_29',
'column_30', 'column_31', 'column_32', 'column_33', 'column_34',
'column_35', 'column_36', 'column_37', 'column_38', 'column_39',
'column_40'};
predictors = inputTable(:, predictorNames);
response = inputTable.column_41;
isCategoricalPredictor = [false, false, false, false, false, false,
false, false, false, false, false, false, false, false, false,
false, false, false, false, false, false, false, false, false,
false, false, false, false, false, false, false, false, false,
false, false, false, false, false, false, false, false];

% Train a classifier
% This code specifies all the classifier options and trains the
classifier.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Function of KNN %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
classificationKNN = fitcknn(...
    predictors, ...
    response, ...
    'Distance', 'Euclidean', ...
    'Exponent', [], ...
    'NumNeighbors', 1, ...
    'DistanceWeight', 'Equal', ...
    'Standardize', true, ...
    'ClassNames', [1; 5]);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Function of SVM %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
classificationSVM = fitsvm(...
    predictors, ...
    response, ...
    'KernelFunction', 'polynomial', ...
    'PolynomialOrder', 2, ...
    'KernelScale', 'auto', ...
    'BoxConstraint', 1, ...
    'Standardize', true, ...
    'ClassNames', [1; 5]);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Function of BT %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
classificationEnsemble = fitensemble(...
    predictors, ...
    response, ...
    'Bag', ...
    30, ...
    'Tree', ...
    'Type', 'Classification', ...
    'ClassNames', [1; 5]);

```

```

% Create the result struct with predict function
predictorExtractionFcn = @(x) array2table(x, 'VariableNames',
predictorNames);
knnPredictFcn = @(x) predict(classificationKNN, x);
trainedClassifier.predictFcn = @(x)
knnPredictFcn(predictorExtractionFcn(x));
svmPredictFcn = @(x) predict(classificationSVM, x);
trainedClassifier.predictFcn = @(x)
svmPredictFcn(predictorExtractionFcn(x));
ensemblePredictFcn = @(x) predict(classificationEnsemble, x);
trainedClassifier.predictFcn = @(x)
ensemblePredictFcn(predictorExtractionFcn(x));

% Add additional fields to the result struct
trainedClassifier.ClassificationKNN = classificationKNN;
trainedClassifier.ClassificationSVM = classificationSVM;
trainedClassifier.ClassificationEnsemble = classificationEnsemble;

% Extract predictors and response
% This code processes the data into the right shape for training the
% classifier.
% Convert input to table
inputTable = array2table(x, 'VariableNames', {'column_1',
'column_2', 'column_3', 'column_4', 'column_5', 'column_6',
'column_7', 'column_8', 'column_9', 'column_10', 'column_11',
'column_12', 'column_13', 'column_14', 'column_15', 'column_16',
'column_17', 'column_18', 'column_19', 'column_20', 'column_21',
'column_22', 'column_23', 'column_24', 'column_25', 'column_26',
'column_27', 'column_28', 'column_29', 'column_30', 'column_31',
'column_32', 'column_33', 'column_34', 'column_35', 'column_36',
'column_37', 'column_38', 'column_39', 'column_40', 'column_41'});

predictorNames = {'column_1', 'column_2', 'column_3', 'column_4',
'column_5', 'column_6', 'column_7', 'column_8', 'column_9',
'column_10', 'column_11', 'column_12', 'column_13', 'column_14',
'column_15', 'column_16', 'column_17', 'column_18', 'column_19',
'column_20', 'column_21', 'column_22', 'column_23', 'column_24',
'column_25', 'column_26', 'column_27', 'column_28', 'column_29',
'column_30', 'column_31', 'column_32', 'column_33', 'column_34',
'column_35', 'column_36', 'column_37', 'column_38', 'column_39',
'column_40'};
predictors = inputTable(:, predictorNames);
response = inputTable.column_41;
isCategoricalPredictor = [false, false, false, false, false, false,
false, false, false, false, false, false, false, false, false,
false, false, false, false, false, false, false, false, false,
false, false, false, false, false, false, false, false, false,
false, false, false, false, false, false, false];

% Compute resubstitution accuracy
resubstitutionAccuracy1 = 1 -
resubLoss(trainedClassifier.ClassificationKNN, 'LossFun',
'ClassifError');
Accuracy_KNN=resubstitutionAccuracy1*100

resubstitutionAccuracy2 = 1 -
resubLoss(trainedClassifier.ClassificationSVM, 'LossFun',
'ClassifError');
Accuracy_SVM=resubstitutionAccuracy2*100

```

```
resubstitutionAccuracy3 = 1 -  
resubLoss(trainedClassifier.ClassificationEnsemble, 'LossFun',  
'ClassifError');  
Accuracy_BT=resubstitutionAccuracy3*100
```

# D

## **Matlab simulation code for Frequency Domain with Information Gain Technique**

A simulation code for classification of epileptic EEG signals is presented.

D.I Matlab code for Frequency Domain with Information Gain Technique.

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%% Read dataset of epileptic EEG signals %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

## Reading epileptic EEG data is same as in the appendices A, B and C

```

%%%%%%%% Five classes of epileptic EEG signals (A-E) %%
A=[z]; %z=set A; o=set B; n=set C; f=set D; s=set E.
B=[o];
C=[n];
D=[f];
E=[s];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%% Using DWT to decompose each class %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

tic
x=A; % do each class individually (A, B, C, D, and E classes)
technique
[n m]=size(x);
for i=1:m
    A=x(:,i);
    [C,L] = wavedec(A,5, 'db5');

% Calculation The Coificients Vectors of every Band:
cD1 = detcoef(C,L,1); %DELTA
cD2 = detcoef(C,L,2); %BETA
cD3 = detcoef(C,L,3); %ALPHA
cD4 = detcoef(C,L,4); %THETA
cD5 = detcoef(C,L,5); %GAMMA
cA5 = appcoef(C,L, 'db5',5); %A5

% Calculation the Details Vectors of every Band :

D1 = wrcoef('d',C,L, 'db1',1); %DELTA
D2 = wrcoef('d',C,L, 'db2',2); %BETA
D3 = wrcoef('d',C,L, 'db3',3); %ALPHA
D4 = wrcoef('d',C,L, 'db4',4); %THETA
D5 = wrcoef('d',C,L, 'db5',5); %GAMMA
A5 = wrcoef('a',C,L, 'db5',5); % A5
POWER_DELTA = (sum(A5.^2))/length(A5);

Delta = D1; %figure, plot(1:1:length(Delta),Delta); title('Delta');
Beta = D2; %figure, plot(1:1:length(Beta),Beta); title('Beta');
Alpha = D3; %figure, plot(1:1:length(Alpha), Alpha); title('Alpha');
Theta = D4; %figure, plot(1:1:length(Theta),Theta); title('Theta');
Gamma = D5; %figure, plot(1:1:length(Gamma),Gamma); title('Gamma');
Approximation = A5; %figure,
plot(1:1:length(Approximation),Approximation);title('Approximation')
;

%Power_Delta=abs(sum(Delta.^2));
%Power_Beta=abs(sum(Beta.^2));
%Power_Alpha=abs(sum(Alpha.^2));
%Power_Theta=abs(sum(Theta.^2));
%Power_Gamma=abs(sum(Gamma.^2));

Wave_D1(:,i)=Delta;
Wave_D2(:,i)=Beta;
Wave_D3(:,i)=Alpha;
Wave_D4(:,i)=Theta;

```

```

Wave_D5(:,i)=Gamma;
Wave_A5(:,i)=Approximation;
end
%toc
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%% Extract feature from each frequency band %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%% 1st band %%%%%%%%%
v1=Wave_D1(:,1:25);
v2=Wave_D1(:,26:50);
v3=Wave_D1(:,51:75);
v4=Wave_D1(:,76:100);

for k=1:n

    fs11(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),mode(v1(k,:)),median(v1(k,:))/2,3*median(v1(k,:))/2,(3*medi
    an(v1(k,:))/2-median(v1(k,:))/2), std(v1(k,:))];
end
for k=1:n

    fs12(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),mode(v2(k,:)),median(v2(k,:))/2,3*median(v2(k,:))/2,(3*medi
    an(v2(k,:))/2-median(v2(k,:))/2), std(v2(k,:))];
end
for k=1:n

    fs13(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
    :)),mode(v3(k,:)),median(v3(k,:))/2,3*median(v3(k,:))/2,(3*medi
    an(v3(k,:))/2-median(v3(k,:))/2), std(v3(k,:))];
end
for k=1:n

    fs14(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
    :)),mode(v4(k,:)),median(v4(k,:))/2,3*median(v4(k,:))/2,(3*medi
    an(v4(k,:))/2-median(v4(k,:))/2), std(v4(k,:))];
end

Feature_Wave_D1=[fs11 fs12 fs13 fs14];
%%
%%%%%%%% 2nd band %%%%%%%%%
v1=Wave_D2(:,1:25);
v2=Wave_D2(:,26:50);
v3=Wave_D2(:,51:75);
v4=Wave_D2(:,76:100);

for k=1:n

    fs21(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
    :)),mode(v1(k,:)),median(v1(k,:))/2,3*median(v1(k,:))/2,(3*medi
    an(v1(k,:))/2-median(v1(k,:))/2), std(v1(k,:))];
end
for k=1:n

    fs22(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
    :)),mode(v2(k,:)),median(v2(k,:))/2,3*median(v2(k,:))/2,(3*medi
    an(v2(k,:))/2-median(v2(k,:))/2), std(v2(k,:))];
end
for k=1:n

```

```

        fs23(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
        :)),mode(v3(k,:)),median(v3(k,:))/2,3*median(v3(k,:))/2,(3*medi
        an(v3(k,:))/2-median(v3(k,:))/2), std(v3(k,:))];
    end
    for k=1:n

        fs24(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
        :)),mode(v4(k,:)),median(v4(k,:))/2,3*median(v4(k,:))/2,(3*medi
        an(v4(k,:))/2-median(v4(k,:))/2), std(v4(k,:))];
    end

    Feature_Wave_D2=[fs21 fs22 fs23 fs24];
    %%
    %%%%%%%%%%% 3rd band %%%%%%%%%%%
    v1=Wave_D3(:,1:25);
    v2=Wave_D3(:,26:50);
    v3=Wave_D3(:,51:75);
    v4=Wave_D3(:,76:100);

    for k=1:n

        fs31(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
        :)),mode(v1(k,:)),median(v1(k,:))/2,3*median(v1(k,:))/2,(3*medi
        an(v1(k,:))/2-median(v1(k,:))/2), std(v1(k,:))];
    end
    for k=1:n

        fs32(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
        :)),mode(v2(k,:)),median(v2(k,:))/2,3*median(v2(k,:))/2,(3*medi
        an(v2(k,:))/2-median(v2(k,:))/2), std(v2(k,:))];
    end
    for k=1:n

        fs33(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
        :)),mode(v3(k,:)),median(v3(k,:))/2,3*median(v3(k,:))/2,(3*medi
        an(v3(k,:))/2-median(v3(k,:))/2), std(v3(k,:))];
    end
    for k=1:n

        fs34(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
        :)),mode(v4(k,:)),median(v4(k,:))/2,3*median(v4(k,:))/2,(3*medi
        an(v4(k,:))/2-median(v4(k,:))/2), std(v4(k,:))];
    end

    Feature_Wave_D3=[fs31 fs32 fs33 fs34];
    %%
    %%%%%%%%%%% 4th band %%%%%%%%%%%
    v1=Wave_D4(:,1:25);
    v2=Wave_D4(:,26:50);
    v3=Wave_D4(:,51:75);
    v4=Wave_D4(:,76:100);

    for k=1:n

        fs41(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
        :)),mode(v1(k,:)),median(v1(k,:))/2,3*median(v1(k,:))/2,(3*medi
        an(v1(k,:))/2-median(v1(k,:))/2), std(v1(k,:))];
    end
    for k=1:n

```



```

        fs42(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
        :)),mode(v2(k,:)),median(v2(k,:))/2,3*median(v2(k,:))/2,(3*medi
        an(v2(k,:))/2-median(v2(k,:))/2), std(v2(k,:))];
    end
    for k=1:n

        fs43(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
        :)),mode(v3(k,:)),median(v3(k,:))/2,3*median(v3(k,:))/2,(3*medi
        an(v3(k,:))/2-median(v3(k,:))/2), std(v3(k,:))];
    end
    for k=1:n

        fs44(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
        :)),mode(v4(k,:)),median(v4(k,:))/2,3*median(v4(k,:))/2,(3*medi
        an(v4(k,:))/2-median(v4(k,:))/2), std(v4(k,:))];
    end

    Feature_Wave_D4=[fs41 fs42 fs43 fs44];
    %%
    %%%%%%%%%% 5th band %%%%%%%%%%
    v1=Wave_D5(:,1:25);
    v2=Wave_D5(:,26:50);
    v3=Wave_D5(:,51:75);
    v4=Wave_D5(:,76:100);

    for k=1:n

        fs51(k,:)=[min(v1(k,:)),max(v1(k,:)),mean(v1(k,:)),median(v1(k,
        :)),mode(v1(k,:)),median(v1(k,:))/2,3*median(v1(k,:))/2,(3*medi
        an(v1(k,:))/2-median(v1(k,:))/2), std(v1(k,:))];
    end
    for k=1:n

        fs52(k,:)=[min(v2(k,:)),max(v2(k,:)),mean(v2(k,:)),median(v2(k,
        :)),mode(v2(k,:)),median(v2(k,:))/2,3*median(v2(k,:))/2,(3*medi
        an(v2(k,:))/2-median(v2(k,:))/2), std(v2(k,:))];
    end
    for k=1:n

        fs53(k,:)=[min(v3(k,:)),max(v3(k,:)),mean(v3(k,:)),median(v3(k,
        :)),mode(v3(k,:)),median(v3(k,:))/2,3*median(v3(k,:))/2,(3*medi
        an(v3(k,:))/2-median(v3(k,:))/2), std(v3(k,:))];
    end
    for k=1:n

        fs54(k,:)=[min(v4(k,:)),max(v4(k,:)),mean(v4(k,:)),median(v4(k,
        :)),mode(v4(k,:)),median(v4(k,:))/2,3*median(v4(k,:))/2,(3*medi
        an(v4(k,:))/2-median(v4(k,:))/2), std(v4(k,:))];
    end

    Feature_Wave_D5=[fs51 fs52 fs53 fs54];

    %%%%%%%%%% Information Gain %%%%%%%%%%

    %%%%%%%%%% 1th %%%%%%%%%%
    TR=Feature_Wave_D1;
    TR=TR';

```

```

TR1=Feature_Wave_D1;
[adp1 av1 array1]=HADI_ENT(TR);

N=1;
for i=1:n
    if array1(i)<adp1
        InfoGain11(N,:)=TR1(i,:);
        N=N+1;
    end
end
%%
%%%%%% 2nd %%%%%%%%%
TR=Feature_Wave_D2;
TR=TR';
TR1=Feature_Wave_D2;
[adp1 av1 array1]=HADI_ENT(TR);

N=1;
for i=1:n
    if array1(i)<adp1
        InfoGain12(N,:)=TR1(i,:);
        N=N+1;
    end
end
%%
%%%%%%%% 3rd %%%%%%%%%
TR=Feature_Wave_D3;
TR=TR';
TR1=Feature_Wave_D3;
[adp1 av1 array1]=HADI_ENT(TR);

N=1;
for i=1:n
    if array1(i)<adp1
        InfoGain13(N,:)=TR1(i,:);
        N=N+1;
    end
end
%%
%%%%%%%% 4th %%%%%%%%%
TR=Feature_Wave_D4;
TR=TR';
TR1=Feature_Wave_D4;
[adp1 av1 array1]=HADI_ENT(TR);

N=1;
for i=1:n
    if array1(i)<adp1
        InfoGain14(N,:)=TR1(i,:);
        N=N+1;
    end
end
%%
%%%%%%%% 5th %%%%%%%%%
TR=Feature_Wave_D5;
TR=TR';
TR1=Feature_Wave_D5;
[adp1 av1 array1]=HADI_ENT(TR);

N=1;

```



```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                %%%%%%%%%% Classification %%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clear all
clc

%%%%%%%%%% Cross validation %%%%%%%%%%

X= DWT_InfoG_AvsE1;                % Do each case of frequency band
x=[X(1:1857,:);X(1858:3713,:)];
y=[1*ones(1857,1);-1*ones(1856,1)];

Xtest= [X(1274:2546,:);X(3820:5092,:)];
Ytest=[1*ones(1273,1);-1*ones(1273,1)];

size(x)
size(y)

%%
%%%%%%%%%% LS_SVM %%%%%%%%%%
% Changeable parameters (gam (γ) and sig2 (σ²))

gam=10;
sig2=1;
%type='classification';
%L_fold=10;
%[gam,sig2]=tunelssvm({x,y,type,1,1,'RBF_kernel'},[],...
    % 'gridsearch',{},'crossvalidate',{x,y,L_fold,'misclass'});

%disp(gam)
%disp(sig2)
[alpha,b] = trainlssvm({x,y,type,gam,sig2,'RBF_kernel'});
disp(b)

Yh=simlssvm({x,y,type,gam,sig2,'RBF_kernel'},{alpha,b},Xtest);
plotlssvm({x,y,type,gam,sig2,'RBF_kernel'},{alpha,b});
%%

[perc,n,which]=misclass(Ytest,Yh); % Which: contains the indices of
the misclassified instances(the first column gives the row, the
second the column index)
n; % is the number of misclassifications
perc; % is the rate of misclassifications (between 0 and 1)
[C,order] = confusionmat(Ytest,Yh);
C
order

%%
Y_latent=latentlssvm({x,y,type,gam,sig2,'RBF_kernel'},{alpha,b},x);
[area,se,thresholds,oneMinusspec,sens,TN,TP,FN,FP]=roc({x,y,type,gam
,sig2,'RBF_kernel'});
%[thresholds oneMinusspec sens ];
%%
sensitivity=TP/(TP+FN)*100
specificity=TN/(TN+FP)*100
Accuracy=(TP+TN)/(TP+TN+FP+FN)*100

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%% Using FFT to decompose each class %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
tic
Test1=[];
Test2=[];
Test3=[];
Test4=[];
Test5=[];
% Test6=[];
% Test7=[];
% Test8=[];
% Test9=[];
% Test10=[];

Denoised_x_total=B; % do each class individually (A, B, C, D, and E
classes)

Test1=Delta_bandfilter(Denoised_x_total);
Test2=Theta_bandfilter(Denoised_x_total);
Test3=Alfa_bandfilter(Denoised_x_total);
Test4=Beta_bandfilter(Denoised_x_total);
Test5=Gama_bandfilter(Denoised_x_total);

% Test6=Test1.^2;
% Test7=Test1.^2;
% Test8=Test1.^2;
% Test9=Test1.^2;
% Test10=Test1.^2;

toc
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%% Extract feature from each frequency band %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%% 1st band %%%%%%%%%
x1=Test1(:,1:25);
x2=Test1(:,26:50);
x3=Test1(:,51:75);
x4=Test1(:,76:100);

for k=1:4097

    fs1(k,:)=[min(x1(k,:)),max(x1(k,:)),mean(x1(k,:)),median(x1(k,:
    )),mode(x1(k,:)),median(x1(k,:))/2,3*median(x1(k,:))/2,(3*media
    n(x1(k,:))/2-median(x1(k,:))/2), std(x1(k,:))];
end
for k=1:4097

    fs2(k,:)=[min(x2(k,:)),max(x2(k,:)),mean(x2(k,:)),median(x2(k,:
    )),mode(x2(k,:)),median(x2(k,:))/2,3*median(x2(k,:))/2,(3*media
    n(x2(k,:))/2-median(x2(k,:))/2), std(x2(k,:))];
end
for k=1:4097

    fs3(k,:)=[min(x3(k,:)),max(x3(k,:)),mean(x3(k,:)),median(x3(k,:
    )),mode(x3(k,:)),median(x3(k,:))/2,3*median(x3(k,:))/2,(3*media
    n(x3(k,:))/2-median(x3(k,:))/2), std(x3(k,:))];
end

```

```

for k=1:4097

    fs4(k,:)=[min(x4(k,:)),max(x4(k,:)),mean(x4(k,:)),median(x4(k,:))
    ),mode(x4(k,:)),median(x4(k,:))/2,3*median(x4(k,:))/2,(3*median(x4(k,:))
    /2-median(x4(k,:))/2), std(x4(k,:))];
end
Feature_FFT_D1=[fs1 fs2 fs3 fs4];
%%
%%%%%%%%%% 2nd band %%%%%%%%%%

x1=Test2(:,1:25);
x2=Test2(:,26:50);
x3=Test2(:,51:75);
x4=Test2(:,76:100);

for k=1:4097

    fs1(k,:)=[min(x1(k,:)),max(x1(k,:)),mean(x1(k,:)),median(x1(k,:))
    ),mode(x1(k,:)),median(x1(k,:))/2,3*median(x1(k,:))/2,(3*median(x1(k,:))
    /2-median(x1(k,:))/2), std(x1(k,:))];
end
for k=1:4097

    fs2(k,:)=[min(x2(k,:)),max(x2(k,:)),mean(x2(k,:)),median(x2(k,:))
    ),mode(x2(k,:)),median(x2(k,:))/2,3*median(x2(k,:))/2,(3*median(x2(k,:))
    /2-median(x2(k,:))/2), std(x2(k,:))];
end
for k=1:4097

    fs3(k,:)=[min(x3(k,:)),max(x3(k,:)),mean(x3(k,:)),median(x3(k,:))
    ),mode(x3(k,:)),median(x3(k,:))/2,3*median(x3(k,:))/2,(3*median(x3(k,:))
    /2-median(x3(k,:))/2), std(x3(k,:))];
end
for k=1:4097

    fs4(k,:)=[min(x4(k,:)),max(x4(k,:)),mean(x4(k,:)),median(x4(k,:))
    ),mode(x4(k,:)),median(x4(k,:))/2,3*median(x4(k,:))/2,(3*median(x4(k,:))
    /2-median(x4(k,:))/2), std(x4(k,:))];
end
Feature_FFT_D2=[fs1 fs2 fs3 fs4];
%%
%%%%%%%%%% 3rd band %%%%%%%%%%

x1=Test3(:,1:25);
x2=Test3(:,26:50);
x3=Test3(:,51:75);
x4=Test3(:,76:100);

for k=1:4097

    fs1(k,:)=[min(x1(k,:)),max(x1(k,:)),mean(x1(k,:)),median(x1(k,:))
    ),mode(x1(k,:)),median(x1(k,:))/2,3*median(x1(k,:))/2,(3*median(x1(k,:))
    /2-median(x1(k,:))/2), std(x1(k,:))];
end
for k=1:4097

    fs2(k,:)=[min(x2(k,:)),max(x2(k,:)),mean(x2(k,:)),median(x2(k,:))
    ),mode(x2(k,:)),median(x2(k,:))/2,3*median(x2(k,:))/2,(3*median(x2(k,:))
    /2-median(x2(k,:))/2), std(x2(k,:))];
end
for k=1:4097

```

```

        fs3(k,:)=[min(x3(k,:)),max(x3(k,:)),mean(x3(k,:)),median(x3(k,:))
        ),mode(x3(k,:)),median(x3(k,:))/2,3*median(x3(k,:))/2,(3*media
        n(x3(k,:))/2-median(x3(k,:))/2), std(x3(k,:))];
    end
    for k=1:4097

        fs4(k,:)=[min(x4(k,:)),max(x4(k,:)),mean(x4(k,:)),median(x4(k,:))
        ),mode(x4(k,:)),median(x4(k,:))/2,3*median(x4(k,:))/2,(3*media
        n(x4(k,:))/2-median(x4(k,:))/2), std(x4(k,:))];
    end
    Feature_FFT_D3=[fs1 fs2 fs3 fs4];
    %%
    %%%%%%%%% 4th band %%%%%%%%%

    x1=Test4(:,1:25);
    x2=Test4(:,26:50);
    x3=Test4(:,51:75);
    x4=Test4(:,76:100);

    for k=1:4097

        fs1(k,:)=[min(x1(k,:)),max(x1(k,:)),mean(x1(k,:)),median(x1(k,:))
        ),mode(x1(k,:)),median(x1(k,:))/2,3*median(x1(k,:))/2,(3*media
        n(x1(k,:))/2-median(x1(k,:))/2), std(x1(k,:))];
    end
    for k=1:4097

        fs2(k,:)=[min(x2(k,:)),max(x2(k,:)),mean(x2(k,:)),median(x2(k,:))
        ),mode(x2(k,:)),median(x2(k,:))/2,3*median(x2(k,:))/2,(3*media
        n(x2(k,:))/2-median(x2(k,:))/2), std(x2(k,:))];
    end
    for k=1:4097

        fs3(k,:)=[min(x3(k,:)),max(x3(k,:)),mean(x3(k,:)),median(x3(k,:))
        ),mode(x3(k,:)),median(x3(k,:))/2,3*median(x3(k,:))/2,(3*media
        n(x3(k,:))/2-median(x3(k,:))/2), std(x3(k,:))];
    end
    for k=1:4097

        fs4(k,:)=[min(x4(k,:)),max(x4(k,:)),mean(x4(k,:)),median(x4(k,:))
        ),mode(x4(k,:)),median(x4(k,:))/2,3*median(x4(k,:))/2,(3*media
        n(x4(k,:))/2-median(x4(k,:))/2), std(x4(k,:))];
    end
    Feature_FFT_D4=[fs1 fs2 fs3 fs4];
    %%
    %%%%%%%%% 5th band %%%%%%%%%

    x1=Test5(:,1:25);
    x2=Test5(:,26:50);
    x3=Test5(:,51:75);
    x4=Test5(:,76:100);

    for k=1:4097

        fs1(k,:)=[min(x1(k,:)),max(x1(k,:)),mean(x1(k,:)),median(x1(k,:))
        ),mode(x1(k,:)),median(x1(k,:))/2,3*median(x1(k,:))/2,(3*media
        n(x1(k,:))/2-median(x1(k,:))/2), std(x1(k,:))];
    end
    for k=1:4097

```

```

        fs2(k,:)=[min(x2(k,:)),max(x2(k,:)),mean(x2(k,:)),median(x2(k,:))
        ),mode(x2(k,:)),median(x2(k,:))/2,3*median(x2(k,:))/2,(3*media
        n(x2(k,:))/2-median(x2(k,:))/2), std(x2(k,:))];
end
for k=1:4097

        fs3(k,:)=[min(x3(k,:)),max(x3(k,:)),mean(x3(k,:)),median(x3(k,:))
        ),mode(x3(k,:)),median(x3(k,:))/2,3*median(x3(k,:))/2,(3*media
        n(x3(k,:))/2-median(x3(k,:))/2), std(x3(k,:))];
end
for k=1:4097

        fs4(k,:)=[min(x4(k,:)),max(x4(k,:)),mean(x4(k,:)),median(x4(k,:))
        ),mode(x4(k,:)),median(x4(k,:))/2,3*median(x4(k,:))/2,(3*media
        n(x4(k,:))/2-median(x4(k,:))/2), std(x4(k,:))];
end
Feature_FFT_D5=[fs1 fs2 fs3 fs4];
%%
                %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                %%%%%%%%% Information Gain %%%%%%%%%%%
                %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

TR=Feature_FFT_D1; % first band of FFT
TR=TR';
TR1=Feature_FFT_D1;
[adp1 av1 array1]=HADI_ENT(TR);

N=1;
for i=1:4097
    if array1(i)<adp1
        InfoadpFFT21(N,:)=TR1(i,:);
        N=N+1;
    end
end

TR=Feature_FFT_D2; % 2nd Band of FFT
TR=TR';
TR1=Feature_FFT_D2;
[adp1 av1 array1]=HADI_ENT(TR);

N=1;
for i=1:4097
    if array1(i)<adp1
        InfoadpFFT22(N,:)=TR1(i,:);
        N=N+1;
    end
end

TR=Feature_FFT_D3; % 3rd Band of FFT
TR=TR';
TR1=Feature_FFT_D3;
[adp1 av1 array1]=HADI_ENT(TR);

N=1;
for i=1:4097
    if array1(i)<adp1
        InfoadpFFT23(N,:)=TR1(i,:);
        N=N+1;
    end
end

```



```

end

TR=Feature_FFT_D4; % 4th Band of FFT
TR=TR';
TR1=Feature_FFT_D4;
[adp1 av1 array1]=HADI_ENT(TR);

N=1;
for i=1:4097
    if array1(i)<adp1
        InfoadpFFT24(N,:)=TR1(i,:);
        N=N+1;
    end
end

TR=Feature_FFT_D5; % 5th Band of FFT
TR=TR';
TR1=Feature_FFT_D5;
[adp1 av1 array1]=HADI_ENT(TR);

N=1;
for i=1:4097
    if array1(i)<adp1
        InfoadpFFT25(N,:)=TR1(i,:);
        N=N+1;
    end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Seven cases of epileptic EEG using FFT %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% case A vs E %%%%%%%%%
FFT_InfoG_Avse1=[InfoadpFFT11; InfoadpFFT51];
FFT_InfoG_Avse2=[InfoadpFFT12; InfoadpFFT52];
FFT_InfoG_Avse3=[InfoadpFFT13; InfoadpFFT53];
FFT_InfoG_Avse4=[InfoadpFFT14; InfoadpFFT54];
FFT_InfoG_Avse5=[InfoadpFFT15; InfoadpFFT55];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% case A vs B %%%%%%%%%
FFT_InfoG_AvsB1=[InfoadpFFT11; InfoadpFFT21];
FFT_InfoG_AvsB2=[InfoadpFFT12; InfoadpFFT22];
FFT_InfoG_AvsB3=[InfoadpFFT13; InfoadpFFT23];
FFT_InfoG_AvsB4=[InfoadpFFT14; InfoadpFFT24];
FFT_InfoG_AvsB5=[InfoadpFFT15; InfoadpFFT25];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% case A vs D %%%%%%%%%
FFT_InfoG_AvsD1=[InfoadpFFT11; InfoadpFFT41];
FFT_InfoG_AvsD2=[InfoadpFFT12; InfoadpFFT42];
FFT_InfoG_AvsD3=[InfoadpFFT13; InfoadpFFT43];
FFT_InfoG_AvsD4=[InfoadpFFT14; InfoadpFFT44];
FFT_InfoG_AvsD5=[InfoadpFFT15; InfoadpFFT45];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% case B vs C %%%%%%%%%
FFT_InfoG_BvsC1=[InfoadpFFT21; InfoadpFFT31];
FFT_InfoG_BvsC2=[InfoadpFFT22; InfoadpFFT32];
FFT_InfoG_BvsC3=[InfoadpFFT23; InfoadpFFT33];
FFT_InfoG_BvsC4=[InfoadpFFT24; InfoadpFFT34];

```

```

FFT_InfoG_BvsC5=[InfoadpFFT25; InfoadpFFT35];

%%%%%%%% case B vs E %%%%%%%%%
FFT_InfoG_BvsE1=[InfoadpFFT21; InfoadpFFT51];
FFT_InfoG_BvsE2=[InfoadpFFT22; InfoadpFFT52];
FFT_InfoG_BvsE3=[InfoadpFFT23; InfoadpFFT53];
FFT_InfoG_BvsE4=[InfoadpFFT24; InfoadpFFT54];
FFT_InfoG_BvsE5=[InfoadpFFT25; InfoadpFFT55];

%%%%%%%% case C vs E %%%%%%%%%
FFT_InfoG_CvsE1=[InfoadpFFT31; InfoadpFFT51];
FFT_InfoG_CvsE2=[InfoadpFFT32; InfoadpFFT52];
FFT_InfoG_CvsE3=[InfoadpFFT33; InfoadpFFT53];
FFT_InfoG_CvsE4=[InfoadpFFT34; InfoadpFFT54];
FFT_InfoG_CvsE5=[InfoadpFFT35; InfoadpFFT55];

%%%%%%%% case D vs E %%%%%%%%%
FFT_InfoG_DvsE1=[InfoadpFFT41; InfoadpFFT51];
FFT_InfoG_DvsE2=[InfoadpFFT42; InfoadpFFT52];
FFT_InfoG_DvsE3=[InfoadpFFT43; InfoadpFFT53];
FFT_InfoG_DvsE4=[InfoadpFFT44; InfoadpFFT54];
FFT_InfoG_DvsE5=[InfoadpFFT45; InfoadpFFT55];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                %%%%%%%%% Classification %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clear all
clc
load('implement of FFT A-B,C-D');
%%%%%%%% Cross_validation %%%%%%%%%
X=InfoGadpFFT_A_B1;
x=[X(1:1857,:);X(1858:3713,:)];
y=[1*ones(1857,1);-1*ones(1856,1)];

Xtest= [X(1274:2546,:);X(3820:5092,:)];
Ytest=[1*ones(1273,1);-1*ones(1273,1)];

size(x)
size(y)
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% LS_SVM %%%%%%%%%

% Changeable parameters (gam (Y) and sig2( $\sigma^2$ ))
gam=10;
sig2=1;
%type='classification';
%L_fold=10;
%[gam,sig2]=tunelssvm({x,y,type,1,1,'RBF_kernel'},[],...
    % 'gridsearch',{},'crossvalidate',{x,y,L_fold,'misclass'});

%disp(gam)
%disp(sig2)
[alpha,b] = trainlssvm({x,y,type,gam,sig2,'RBF_kernel'});
disp(b)

Yh=simlssvm({x,y,type,gam,sig2,'RBF_kernel'},{alpha,b},Xtest);
plotlssvm({x,y,type,gam,sig2,'RBF_kernel'},{alpha,b});

```

```

%%

[perc,n,which]=misclass(Ytest,Yh); % Which: contains the indices of
the misclassified instances(the first column gives the row, the
second the column index)
n; % is the number of misclassifications
perc; % is the rate of misclassifications (between 0 and 1)
[C,order] = confusionmat(Ytest,Yh);
C
order

%%
Y_latent=latentlssvm({x,y,type,gam,sig2,'RBF_kernel'},{alpha,b},x);
[area,se,thresholds,oneMinusspec,sens,TN,TP,FN,FP]=roc({x,y,type,gam
,sig2,'RBF_kernel'});
%[thresholds oneMinusspec sens ];
%%
sensitivity=TP/(TP+FN)*100
specificity=TN/(TN+FP)*100
Accuracy=(TP+TN)/(TP+TN+FP+FN)*100

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                    %%%%%%%%% Function of Entropy %%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

function [Adp,Ave,Y] = Untitled3( X )
%UNTITLED3 Summary of this function goes here
% Detailed explanation goes here
[N1,M1] = size(X);
Totals = sum(X,1);
N = sum(sum(X));
Temp_P = (Totals' * ones(1,N1))';
warning off;
Temp_fre = X ./ Temp_P;
warning on;
Temp_fre(isnan(Temp_fre))=0;
LogN = Temp_fre.*log2(Temp_fre);
entropies = -sum(LogN,1);
my_entropy= sum((Totals/N).*entropies);
Y=entropies';
Ave=mean(Y);
Adp=my_entropy;

```

End of thesis