KING'S College LONDON

University of London

6CCS3PRJ

Classification of Epilepsy Types using Deep Learning

Final Project Report

Author: Ayberk Demirkol

Programme: BSc Computer Science

Supervisor: Dr. Hak-Keung Lam

Student ID: 1723743

April 23, 2020

Abstract

Epilepsy is a common neurological disorder affecting millions. Epilepsy is characterized by the nature of seizures patients go through. However, misdiagnosis is a problem due to nature of EEG (Electroencephalogram). A multi-modal system can replace EEG as state of the art diagnosis system. This project proposes classification of EMG (Electromyography), ECG (Electrocardiogram), ACC (Accelerometry) and GSR (Galvanic Skin Response) signals to classify motor and non motor seizures using real clinical data. Features were extracted from time and time frequency domains using DWT (Discrete Wavelet Transform). Multiple traditional classifiers as well as ensemble classifiers were implemented to classify 2 classes of seizure types. Evaluation and comparisons are performed between each classifier. Gaussian noise has been added to original signals to measure the robustness of each classifier. It is concluded that SVM classifiers outperform the others. However, when subject to noise k-Nearest Neighbour and ensemble methods also show promising results.

Originality Avowal

I verify that I am the sole author of this report, except where explicitly stated to the contrary. I grant the right to King's College London to make paper and electronic copies of the submitted work for purposes of marking, plagiarism detection and archival, and to upload a copy of the work to Turnitin or another trusted plagiarism detection service. I confirm this report does not exceed 25,000 words.

Ayberk Demirkol

April 23, 2020

Acknowledgements

I would like to thank my supervisor Dr. Hak-Keung Lam for offering his guidance, expertise and support throughout this project.

Contents

Acronyms

ACC Accelerometry. 1, 5, 8, 10, 13, 15, 38

BCS British Computer Society. 27

DWT Discrete Wavelet Transform. 1, 11, 15–17, 24

ECG Electrocardiogram. 1, 5, 8, 9, 13, 15, 16, 23, 38

EDA Electrodermal Activity. 10

EEG Electroencephalogram. 1, 5, 6, 8–10, 13

EMG Electromyography. 1, 5, 8–10, 13, 16, 17, 23, 38

EMU Epilepsy Monitoring Unit. 11

GSR Galvanic Skin Response. 1, 5, 6, 10, 13, 14, 17, 38

GTCS Generalized Tonic-Clonic Seizure. 9, 10

kNN k-Nearest Neighbour. 10, 21

MAD Median Absolute Deviation. 15

MLP Multilayer Perceptron Neural Network. 9, 20

NN Neural Network. 20

PC Principal Component. 17, 25

PCA Principal Component Analysis. 11, 17, 25

RMS Root Means Squared. 15, 17

SD Standard Deviation. 15–17

SUDEP Sudden Unexpected Death in Epilepsy. 4

SVM Support Vector Machine. 9, 10, 21, 29, 30, 38

Chapter 1

Introduction

A seizure is a burst of uncontrolled electrical disturbance in brain cells. Epilepsy is condition of having multiple, unprovoked seizures that can cause involuntary muscle twitching, spasms or altered consciousness[2]. People who had a head injury, brain infection, stroke or brain tumor are more prone to suffer from seizures, but anybody can get them [19]. Epilepsy appear not as common as other disorders yet it is frequently seen in every age groups [48]. According to [9], there are over 50 million epilepsy patients regardless of age, gender, or race. For most people, the disease can be controlled with therapies, drugs and good seizure management skills. If seizures are not kept under control, it can even lead to Sudden Unexpected Death in Epilepsy (SUDEP). SUDEP is when a person with epilepsy, who was otherwise healthy, passes away in sleep and does appear to be seizure-free. There are over 1 case of SUDEP each year for every 1,000 people with epilepsy [5]. The most effective method to prevent SUDEP known today is lowering risk with seizure control [17]. For seizure control, depending on epilepsy type, patient's needs have to be supplied, and as previous stages, detection of seizure and accurate diagnosis of epilepsy is critical. To prevent possible problems that could be revealed from uncontrolled seizures like SUDEP those stages needs to be processed precisely. As a matter of fact, effective treatment needs to be applied which can be obtained by appropriate methods that are mentioned above. In later chapters, both, seizure types and mechanisms, will be discussed in depth while seizure types will be illustrated, as well.

1.1 Motivations

Currently there are many methods to detect epilepsy, the most common one being Electroencephalogram (EEG) wave analysis. However, EEG analysis is a time consuming procedure that can be monitored and used only by neurologists manually. Also, if a routine EEG is normal, patient is required to stay in a clinic for days [44]. In addition, there is a possibility of misdiagnosis. Misdiagnosing of epilepsy could be in different forms such as misdiagnosis of the seizure type, diagnosing a healthy human with epilepsy and vice versa [34]. This may cause patients to get wrong treatments which would be ineffective and time consuming. According to [6], three quarters of primary care specialists claimed that implementation of IT could diminish possible errors. Due to misdiagnose of epilepsy, misattribution will firstly hinder precise treatment and prolong the process, and waste a necessary portion of time of patient [34]. As a matter of fact, reduction of these errors play a crucial role for the misdiagnosis of epilepsy. Another important aspect of misdiagnose is economic consequences. More than half of physicians indicated IT appliance would decrease the costs and assists patients with higher quality [6]. Therefore, there is a demand for an alternative model which shortens the detection process whilst making the process easier for patients. In literature there are classifiers developed in order to detect seizure and seizure phases [9, 10, 12, 21, 42, 49]. However, there is no classifier specifically focusing on seizure type classification based on motor component. This paper will focus on creating a classifier that classifies seizures based on motor component using Electrocardiogram (ECG), Electromyography (EMG), Accelerometry (ACC) and Galvanic Skin Response (GSR) signals which could be implemented into a single wearable device, thus aiding the patient to get the correct diagnosis.

Along with medicine itself, paper intent to show how machine learning can be implemented into arena of medicine and demolish deficiencies of the field. Even though, computer technology is seen in medicine; the notion of machine learning should be integrated in a greater extent.

1.2 Scope and Objectives

It should be noted that this project will focus on using different classifiers to detect seizure and classify seizure types by using ECG, GSR, EMG, and ACC signals. Classifiers will be created based on the dataset provided by King's College London. RADAR-CAN dataset was collected in the Institute of Psychiatry, Psychology and Neuroscience at King's College London and approved by the committee for research purposes. The dataset includes EMG, ECG, ACC and

GSR signals from patients who had seizures. Seizure start, end times and types are pinpointed by specialists.

In order to develop effective classifiers, a feature vector will be extracted from the raw data to reduce the time and memory used to train the classifiers. After training, the classifiers will be compared and best classifier will be the proposed model for this paper. This proposed model then could be applied to future wearable devices such as watches.

The objective is to create a method that detects and classifies seizure types. This method will contribute to medicine literature within the context of epilepsy type detection. The method will facilitate detection procedure and enable patients to get easier and quicker diagnose, and afterwards treatment. This method will provide advantageous procedure for both clinicians and patients. As it is indicated, EEG is one of the most common methods that has been used in diagnosing epilepsy but has disadvantages that prevent quick-action, or mislead the process. Aforementioned method will eliminate manual process and will classify seizures based on their types. This would diminish possibility of misdiagnosis while increasing speed of precise detection. In view of its role as a solution to existing procedures, this method propose a pragmatic, and quick application that may overcome problems that are stated above.

This chapter gave insight into the main topic, delivered scope and objective, and briefly outlined the main purpose of the paper. Next chapter will define and analyze seizure types, and seizure diagnosis. Additionally, the chapter will acknowledge existing work while in further chapter it will be compared with the paper. Chapter 2 will be followed by the design and specification of this project, explaining the system architecture. Chapter 4 and 5 will describe the seizure type classification system in detail and its implementation respectively. Followed by results and evaluation, the classification results will be analyzed and evaluated with the main aim of the paper, seizure type classification. Lastly, the conclusion and future work will be indicated based on previous chapters of the paper.

Chapter 2

Background

2.1 Seizure Types

Seizures are divided into 2 major groups, being focal onset and generalized onset. Main difference of these types of seizures is the possibility of propagation through different parts of brain. Focal onset seizures start in one side of the brain and stay in that area. Symptom severity may change depending on the electrical discharges and effected lobe in the brain. These seizures typically last from few seconds up to 2 minutes. The patient may lose awareness depending on the type of focal seizure [19]. On the other hand, generalized onset seizures start in one side of brain and spread throughout all. Most of the time patients have impaired awareness in this seizure type [3].

These 2 major groups are further categorized depending on whether the seizure is motor component or non-motor component based. For both major groups, motor seizures may have symptoms such as rhythmic jerking movements, muscle twitching, spasms and limp muscles. Generalized non-motor seizures are mostly called absence seizures. During these seizures, the person might be mistaken for losing focus or they can stop talking in the middle of their sentence. On the other hand, focal non-motor seizure symptoms may include rapid changes in feelings, thinking, autonomic functions or behavioural arrest [13]. Even though there are distinct seizure types yet for detection of epilepsy, mainly, same methods are applied in the field. This will be explained and examined in the next section.

2.2 Seizure Diagnosis

There are various techniques to diagnose epilepsy such as EEG, ECG and EMG signals. Electrical discharges in the brain can be most accurately observed by EEG, thus it is one of the most common techniques used for epilepsy diagnosis. However, this technique may require a patient to be monitored from several hours up to few days. After monitoring, these signals are read by trained specialists [1]. However, EEG's can mislead the neurologists by movement or even blinking artifacts if not supported by video or sound. Some EEG's have automatic phase and spike detection algorithms implemented to pinpoint the seizure and help the experts [44]. Nonetheless, there are no known algorithms implemented for epilepsy type classification. In order to capture and classify the characteristics of a seizure, a prolonged video EEG-monitoring is necessary [1]. This shows that EEG is time consuming, inconvenient, and requires a neurologist and patient to stay in a clinic for a long time. Therefore, EEG is not optimal for quick and clinic-free diagnosis of epilepsy. To tackle this problem, there are multi-modal devices proposed in the literature for different types of seizures. Multi-modal methods to classify seizures makes sense as different types of seizures may be best detected by different sensors. For example [22] proposes that using EMG to detect tonic seizures, a type of motor seizure that causes stiffness in muscle tone, may be more accurate rather than using movement sensors such as ACC. Similarly, [48] explains that while generalized tonic-clonic seizures, a type of motor seizure, can be detected by ACC, EMG; absence seizures are more challenging to detect as they hardly show any movement artifacts and only a minimal impaired awareness. Thus, EEG's are used to capture them. The most common devices used to detect seizure will be explained broadly in the subsections below followed by their place in machine learning literature.

2.2.1 Electroencephalogram

In an EEG test, several electrodes are placed on the patients scalp. These electrodes are connected to an EEG recording machine by wires. Next, electrical discharges in the brain can be seen as waves from a digital platform. These waves can be used for classification purposes. Several researches have been conducted on detection of seizures using EEG waves [12, 21, 42]. have focused on classification of seizure phases into 3 classes which are seizure-free, pre-seizure and seizure. Same dataset have been used for classification in all 3, but with different classifiers. Classification accuracy averages are between 56.7-93.9%. [35, 36] have focused on automatic seizure detection based on these waves. Both trained and tested their models on the same dataset and their approaches were patient specific, meaning the classifiers were trained for each

individual patient and then tested on the same patient. Thus, resulted in a high sensitivity of up to 96.55-97.5% and low false positive rate of 0.21-0.27 per hour. Even though these results are successful, a state of the art seizure detection model cannot be trained for each individual separately.

2.2.2 Electromyography

EMG procedure involves inserting tiny needles into muscles to record electrical discharges. A surface EMG similarly shows electrical activity of muscles, except that needles were placed on the skin. A research on surface EMG in [8] showed that EMG artifacts due to convulsive seizures are different than artifacts due to voluntary muscle contractions and from non-epileptic seizures. In the same research, a generic seizure detection algorithm based on EMG data from 11 patients with Generalized Tonic-Clonic Seizure (GTCS) detected all seizures with 100% sensitivity and false detection rate of 0.03/24 hours. [46] used sEMG signals to detect GTCS. Out of 21 observed GTCS, the algorithm had 95% sensitivity detecting them. It can be concluded that sEMG has promising results for GTCS detection and can potentially detect other seizures with motor-component.

2.2.3 Electrocardiogram

ECG is used to record the heart beats with respect to time. Seizures have been associated with cardiovascular and respiratory changes, thus ECG provides valuable information for the classification purposes. [54] researched the ECG abnormalities and heart rate changes during seizures and found that heart rate elevation was seen in 93% of their patients, often around seizure onset. They have also found a positive correlation between seizure length and number of ECG abnormalities. [27] made use of these abnormalities in ECG with the help of EEG to implement an online seizure detection model. Heart rate absolute value based on R-peak detection algorithm and statistical methods were used for feature extraction from ECG channel. Support Vector Machine (SVM), Multilayer Perceptron Neural Network (MLP), k-Nearest Neighbour and C4.5 Decision Tree classifiers were trained and tested on the dataset. The most successful classifier was found to be SVM with 92.5% accuracy, followed by MLP with 87.6% accuracy. [15] also used ECG and EEG waves to predict epileptic seizure onsets using linear Bayes and k-nearest neighbour classifiers. R-peak detection is implemented alongside with R-R Interval signal obtained by detected R-waves, to be used in feature extraction. ECG-based classifier had accuracy over 99% for all patients, showing promising results for ECG use in

predicting epileptic seizures.

2.2.4 Accelerometry

Although uncommon, ACC is another way to detect motor seizures. ACC is a technique used to estimate acceleration and displacement of a body. ACC devices are commonly small, portable devices that can be used in an ambulatory setting [18]. However, it is challenging to distinguish seizures from daily, repetitive movements [33]. [52] shows that ACC can be used to detect and distinguish physical activities such as walking, sitting and falling. [33] shows that it can be further used for detection of motor-seizures. Stereotypical patterns were observed caused by motor seizures that are easily recognisable to human observers. 48% of the seizures in the study were detected by ACC. In 11% of patients, EEG signal alone was insufficient to detect; hence they used ACC signal as well. The seizures that could not be detected were seizures without motor component. Furthermore, a recent study shows that different classifiers can detect epileptic seizures with high sensitivity. The study makes use of signal entropy and energy of different frequency bands in feature set. k-Nearest Neighbour (kNN), SVM and Random Forest classifiers were tested on 570 hours of data and detected 10 Tonic-Clonic seizures with 90-100% sensitivity and 0.01 false positive per hour [18]. It can be concluded that ACC signals provide valuable information on different types of motor seizures and it would be advantageous to include these signals in a multi-modal detection system. It would also be advantageous to validate motor seizures detected by EMG with ACC or vice-versa.

2.2.5 Galvanic Skin Response

Electrodermal Activity (EDA) is an autonomic response of human body due to skin conductivity. As our body sweats, conductivity of our body increases [51]. GSR, an umbrella term of EDA, reflects the changes in sweat glands. Considering the term, autonomic responses such as sweating are often associated with partial seizures [23]. In [41] it is shown that Complex Partial Seizures, currently known as focal impaired awareness seizures [14], cause EDA increase around 0.7 μS, whereas GTCS cause increase around 20 μS. The same research also found that EDA waves remained amplified for a longer period in GTCS compared to EDA. However, physical activities might result in artifacts similar to a seizure artifact. To better understand the significance of GSR in detecting epileptic seizures, experiments in ambulatory settings are required.

Chapter 3

Design & Specification

The project aims to test accuracy and robustness of different classifiers on classification of seizure types. The proposed system can be used to gather data for a period of time, and then use effective classifiers to accurately predict the type of seizures.

3.1 System Architecture

The system architecture is shown in Figure 3.1. The software is composed of 5 main components. Data is gathered in a clinical setting such as an Epilepsy Monitoring Unit (EMU). Seizure times are identified by a professional and then cropped to be used for feature extraction. In order to test robustness of classifiers Gaussian noise is applied to each signal wave at different levels. Feature extraction is applied in both time-domain and frequency domain using Discrete Wavelet Transform (DWT). Combinations of these result in a feature vector of size 131. Principal Component Analysis (PCA) will be applied to in order to reduce the dimensionality of the feature vector, thus increasing the performance of the classifiers. Feature vectors of each seizure phase is later used to train and test different classifiers used in the project.

3.2 System Requirements

The acquired dataset was originally provided in MATLAB format. Using MATLAB API, dataset is transformed into .csv format to be processed using Python3.8. In order to use the software, Python3.8 must be installed. The project was developed in macOS, however, Python3.8 can be installed on Linux, Windows Vista -and later- (64-bit) and macOS 10.9 and later. In order to install the 3rd party libraries (Scikit-learn, numpy, pickle, pywt and scipy),

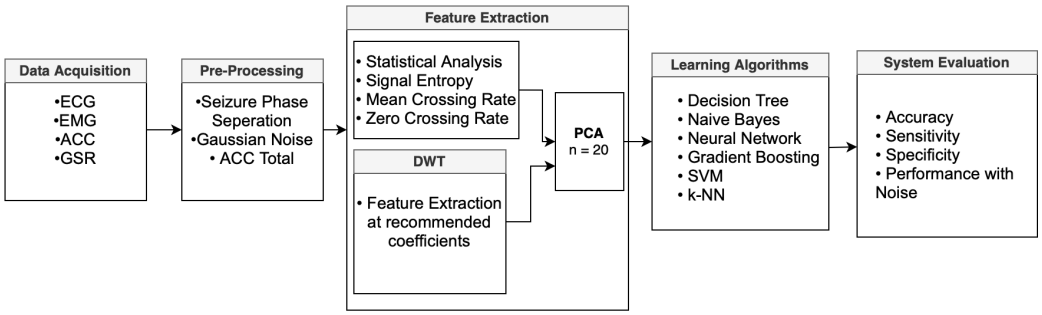


Figure 3.1: Flowchart of System Architecture

a package manager such as PIP is required. All of the above said modules must be installed using PIP in order to run the software.

Chapter 4

Seizure Type Classification System

The details of the proposed seizure type classification system is explained in this chapter.

4.1 Data Acquisition

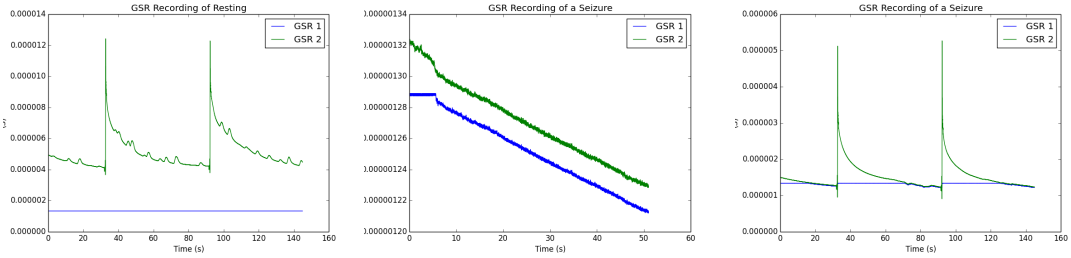
RADAR-CAN dataset used to train the models proposed in this project was provided by the Institute of Psychiatry, Psychology and Neuroscience at King's College London. The study protocol has been approved for research purposes. The dataset consists of EMG, ECG, GSR and ACC signals from patients suffering from different types of seizures. Clinical start and end of the seizures were diagnosed by clinicians using EEG signals which was not provided in the dataset. Only the seizures marked usable by co-operators for seizure type classification were used. Data for each patient is cropped according to seizure periods diagnosed by clinicians.

The 3-axis ACC device provides 3 different channels being x, y and z. Magnitude of the acceleration was calculated by:

$$ACC_t = \sqrt{acc_x^2 + acc_y^2 + acc_z^2} \tag{4.1}$$

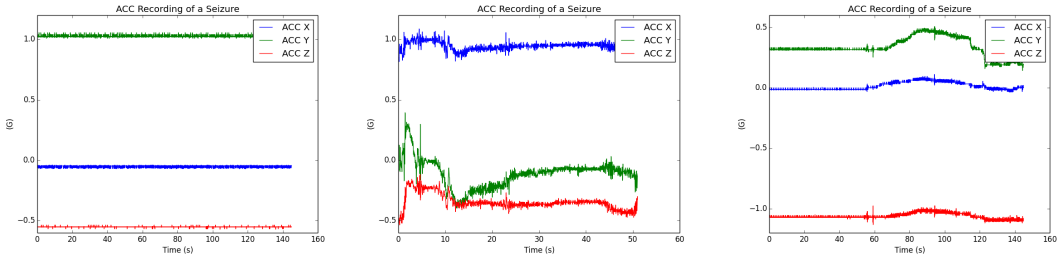
This is done iteratively for every data point in ACC, providing a new channel to be used in feature extraction. The ECG and EMG devices provide 1 channel each. GSR consists of 2 channels, making the total channel count 8. Figures 4.1, 4.2, 4.3 and 4.4 shows examples of the cropped recordings from three different states. From the figure 4.3, it is easy to distinguish

non-motor and motor seizures by looking at the magnitudes of the voltages. During a motor seizure the heart rate is around 0.7mV whereas, it is around 0.07mV during a non-motor seizure. Furthermore, spikes are observed more frequently during a non-motor seizure. 4.1 also provides insight to GSR artifacts seen during non-motor seizures. As mentioned in the chapter 2, GSR measures the electrical conductivity of the body and its measuring unit is in Siemens, shown as 'S' or Ω^{-1}. During a non-motor seizure a continuous decrease in conductivity of the body is observed. However, the resting and motor seizure recordings seems to be showing similar patterns. The only difference is minor spike pattern in the resting recording which might be caused by noise in the environment. Also, from 4.2 the resting pattern is obvious and easy to classify.



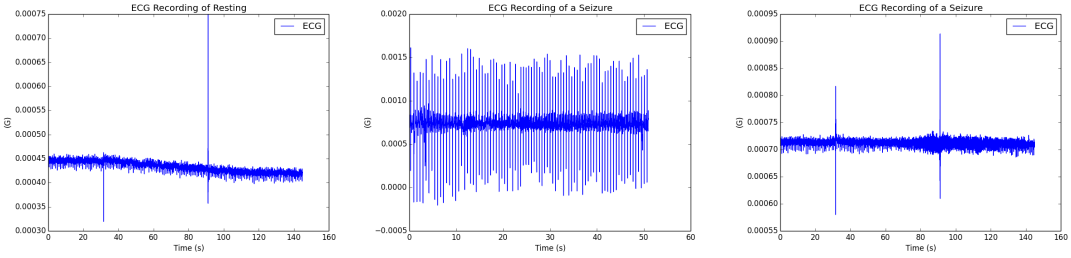
(A) (B) (C)

Figure 4.1: GSR Recordings. Resting (A), Non-Motor Seizure (B), Motor Seizure (C)



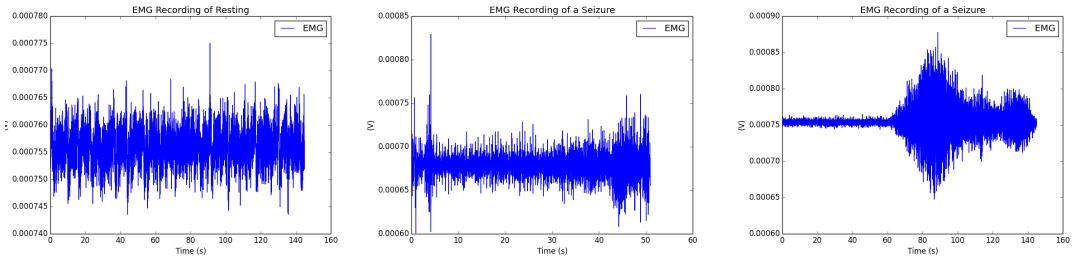
(A) (B) (C)

Figure 4.2: ACC Recordings. Resting (A), Non-Motor Seizure (B), Motor Seizure (C)



(A) (B) (C)

Figure 4.3: ECG Recordings. Resting (A), Non-Motor Seizure (B), Motor Seizure (C)



(A) (B) (C)

Figure 4.4: EMG Recordings. Resting (A), Non-Motor Seizure (B), Motor Seizure (C)

4.2 Feature Extraction

Feature extraction is one of the most important parts in machine learning. The purpose is to eliminate redundant information by selecting important features from the raw data. It also enables faster training process as the size of the data to be trained decreases significantly. Including the ACC_t, magnitude of the 3-axis ACC, there are 8 channels to extract features from for each patient. A feature vector will be created using these channels to be used for classification. Each device is analyzed separately in literature for feature extraction.

In order to capture different characteristics of seizures, features are calculated globally, meaning entire signal length was used to extract features for each channel in time domain. It can be seen from figures 4.1, 4.2, 4.3 and 4.4 that features in time domain is important as sometimes there are no significant pattern difference between two recordings of different seizure types.

For each channel of ACC feature extraction was done only in time domain as there were examples of time-frequency features were not common in literature. Mean, Standard Deviation (SD), Root Means Squared (RMS), Median Absolute Deviation (MAD), interquartile range, maximum, minimum, median, variance, entropy, range and sum of absolute values were calculated and added to feature vector for each channel. Furthermore, correlation values between all 4-channels were calculated and added to feature vector as well. These values are selected specifically for ACC channels as [16, 20, 30, 31, 32, 43, 45] suggest that these are the most interpretive features of ACC device to describe human activities. This resulted in a total of 54 features from ACC device only.

ECG signals were analyzed both in time domain and time-frequency domain. Time domain features used are minimum, maximum, 25-50-75 percentiles, mean, SD, variance, interquartile range, mean absolute deviation and range as suggested by [27, 28]. Features extracted from time-frequency domain are reportedly less prone to noise and more informative due to ECG signals being non-stationary [24]. For time-frequency domain DWT will be used as it is very

common method to use for ECG wave decomposition in literature [7, 25, 47].

DWT is used to clear signals from redundancy. The main signal is decomposed into different levels; and each level provides multi resolution analysis with different resolutions [29]. For each level, the signal is passed through high and low pass filters. This results in 2 values, being approximation and detail coefficients that are produced by low pass filter and high pass filter respectively. Both of these coefficients at each level can be used for feature extraction purposes. For each of the next levels, approximation coefficient is further decomposed using the same mother wavelet function [42]. A 4-level DWT decomposition scheme is shown in figure 4.5 for a signal of length n. Signal frequencies are downsized by a factor of 2 for each decomposition, shown by $\downarrow^2$. DWT is defined by the equation [53]:

$$DWT(j,b) = \frac{1}{\sqrt{2^j}} \int x(t)\psi^*(\frac{t-b}{2^j})dt \tag{4.2}$$

where $x(t)$ is the signal, ψ is the analysing wavelet, 2^j is scalar parameter and b is the position parameter. Finding a mother wavelet function is also important for DWT. Daubechies wavelets were found to be most suitable for ECG. Therefore, Daubechies 4 (db4) is used in this system. Maximum, minimum, mean and SD were calculated from sub bands D2, D3, D4 and A4 as these bands contain most of the signal frequencies that are important [24]. These features are combined with the time domain features of ECG, resulting in a feature vector of size 27 from ECG signals.

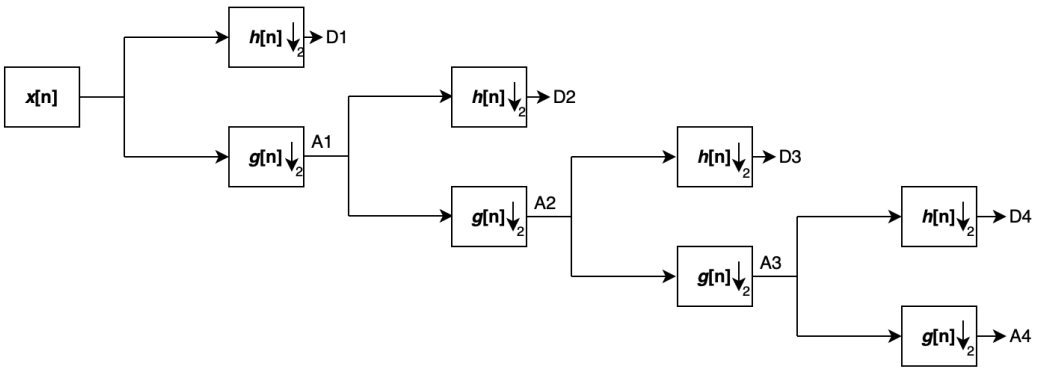


Figure 4.5: 4-level DWT Decomposition

Similarly, features from both time and time-frequency domain were calculated for EMG signals.

Table 4.1: Summary of average classifier score for different number of PCs.

# of PCs	Classifier Average Score
15	76.57
20	78.57
30	74.84
45	72.87
50	69.43

Popular features from literature are selected for time domain feature extraction [38, 39, 40]. SD, variance, mean absolute deviation, RMS, median, mean, range are the features calculated in the time domain. DWT is used for time-frequency domain features. Decomposition level and mother wavelet are again 2 important factors to be considered. [39] concluded that orthogonal Daubechies wavelets (db1-db10) with decomposition level of 4 is most suitable for EMG analysis. 4-level decomposition with Daubechies 10 mother wavelet is used for this system. Kurtosis, entropy, median, zero crossing rate and RMS are calculated from sub band D4. A feature vector of size 12 is formed from EMG analysis.

Finally, time and time-frequency domains are analyzed for GSR feature extraction. There are 2 channels to be analyzed, namely GSR_1 and GSR_2. Mean, SD, maximum, minimum, RMS, median, skewness, kurtosis and range were calculated for both channels. Furthermore, both of these channels were decomposed into 4-levels using Daubechies 4 as mother wavelet. Minimum, maximum, mean, SD, and variance are calculated for D3 and D4 sub bands for both channels. In total, the feature extraction process for all signals result in a feature vector of size 131. In order to decrease the dimensionality of the feature vector Principal Component Analysis (PCA) is used. PCA identifies n number of Principal Components (PC) using orthogonal transformation and ranks them by variance. For this system, different number of PCs have been tested. Table 4.1 shows the average performance of different number of PCs. As 20 PCs performs the best overall, a feature vector of size 20 is decided to be used for classification.

4.3 Machine Learning

Machine learning techniques allow the system to learn specific patterns from data. Data is mostly represented by a tuple of values, which we call feature vector. This learning process is called training. Learning is divided into supervised, unsupervised and reinforcement learning. Different supervised learning algorithms were considered in this project. Supervised learning algorithms train on data (input) and their corresponding outcomes (label/class/output). They learn a mapping function based on the training dataset. After training process, as new data is

introduced to the system, they predict an outcome based on the trained data. This is called classification.

4.3.1 Decision Tree Classifier

A decision tree algorithm finds a small tree consistent with the training dataset. The learned function can be represented by a decision tree. Learned trees can also be represented by if-then rules. Basic idea behind decision tree algorithm is to choose recursively the best features from the given feature vector. At each iteration, the algorithm splits the training data by using a feature as a criteria. However, some splits are better than others. This can be measured by GINI impurity or entropy. GINI impurity is used for this system. For a binary classification problem, GINI impurity can be defined as:

$$G(S) = 1 - ((\frac{p}{p+n})^2 + (\frac{n}{p+n})^2) \tag{4.3}$$

where S is a set of positive (p) and negative (n) at the root. For n classes it can be calculated as:

$$G(S) = \sum_{i=1}^{n} p_i \sum_{k \neq i} p_k \tag{4.4}$$

GINI impurity is calculated for each feature that splits the dataset S into subsets S_i. Therefore, for a feature $X = x_1, x_2, ..., x_n \in \Re$ that splits a set S into subsets S_i the GINI impurity can be defined as:

$$G(S, X) = \sum_{i=1} \frac{|S_i|}{|S|} G(S_i) \tag{4.5}$$

where $|S_i|$ and $|S|$ are the sizes of the sets. An example decision tree after training is shown in figure 4.6. It is worth noting that, this figure is just an example visualisation of a decision tree; it is not a tree created by an algorithm for this project. After learning process, when a new sample is presented, it is assigned to a class by following the nodes of the tree.

One disadvantage of decision tree classifier is overfitting. Overfitting happens when a classifier learns the training data to an extent that it impacts the performance on testing data poorly. Therefore, decision tree classifier is expected to have high training accuracy, whereas lower testing accuracy compared to other classifiers.

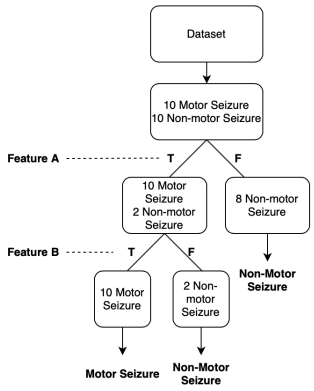


Figure 4.6: Sketch of an example decision tree

4.3.2 Naive Bayes Classifier

Naive Bayes classifier is an algorithm based on the naive assumption of applying Bayes' Theorem. For a feature vector $X = x_1, x_2, ..., x_n \in \Re$ and class label c Bayes' theorem is defined by:

$$P(c|x_1, x_2, ..., x_n) = \frac{P(x_1, x_2, ..., x_n|c)P(c))}{P(x_1, x_2, ..., x_n))} \tag{4.6}$$

Naive Bayes assumes that features are conditionally independent. Conditional independence assumption is shown by:

$$P(x_i|c, x_1, ..., x_{i-1}, x_{i+1}, x_n) = P(x_i, c) \tag{4.7}$$

Therefore, theorem can be simplified to[37]:

$$P(c|x_1, ..., x_n) = \frac{P(c)\prod_{i=1}^{n} P(x_i|c)}{P(x_1, ..., x_n)} \tag{4.8}$$

Finally, since $P(x_1, ..., x_n)$ is a constant value, we can use Maximum a Posteriori theory to further simplify the equation into[37]:

$$\hat{c} = arg\max_{c} P(c)\prod_{i=1}^{n} P(x_i|c) \tag{4.9}$$

This shows that once the algorithm learns $P(c)$ and $P(x_i|c)$ it can start making predictions about c. One disadvantage of Naive Bayes classifier is that it can underfit due to small numbers in the feature vector. Hence, it can result in poor predictions.

4.3.3 Neural Network

A NN is a classification algorithm that learns a function $f(\cdot) : R^i \rightarrow R^o$ where i is the dimensionality of input and o is the dimensionality of output [37]. The simplest NN is a perceptron, which is also called a single-layer NN. For a feature vector $X = x_1, x_2, ..., x_n \in \Re$ it learns synaptic weights $w_1, w_2, ..., w_n$. Afterwards, it is passed into an activation function:

$$Y = \begin{cases} +1, if \sum_{i=1}^{n} w_i x_i \geq 0 \\ 0, if \sum_{i=1}^{n} w_i x_i < 0 \end{cases} \tag{4.10}$$

The learning process consists of changing values of $w_0, ..., w_n$ until a specified number of iterations or convergence, when $f(\cdot)$ function can match all inputs with their corresponding output. However, a perceptron can only compute linearly separable functions. Multilayer Perceptron Neural Network (MLP) overcomes this challenge by using hidden layers. MLP is able to compute non-linear functions due to hidden layers, which use non-linear activation functions. Figure 4.7 is an example of a 1 hidden-layer MLP classifier. In this project MLP has 20 inputs,

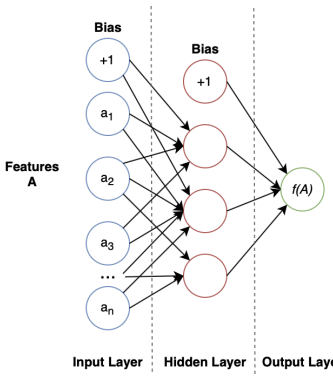


Figure 4.7: Sketch of Multi-layer Perceptron with 1 hidden layer

5 hidden layers with 10 hidden nodes and 1 output node. Different number of hidden layers were tested and it has been concluded that 5 hidden layers with 10 hidden nodes give the best outcome.

4.3.4 Gradient Boosting

Boosting is a method to increase performance of weak learning classifiers. Gradient boosting trains an additive model sequentially [4]. It starts by training a decision tree, then at each

stage trees are fit on a negative gradient loss function. The loss function defines how good model's learned coefficients are at fitting trained data. One of the main advantages of Gradient Boosting is that number of stages to perform boosting can be defined and high numbers usually result in better performance as gradient boosting classifier is highly resistant to overfitting [37].

4.3.5 Support Vector Machine

During training process SVM algorithm aims to find the best hyperplane that divides the dataset in such a way that the distance from closest point from each class to the hyperplane is maximized. The points closest to the hyperplane are called support vectors. After the hyperplane is formed, when a new data is introduced to SVM for testing, it's class is determined depending on which part of the hyperplane it's put. The confidence of the class increases as the distance from hyperplane increases. However, this is only a solution for linearly seperable data. To get a non-linear solution the data is transformed into a higher dimension shown by transition function $\psi(X)$ for feature vector $X = x_1, x_2, ..., x_n \in \Re$. Therefore, the hyperplane can be defined as:

$$\omega \cdot z + b = 0 \tag{4.11}$$

where $z = \psi(X)$, ω is weight vector and b is bias. However, as the feature space expands, computational complexity becomes more expensive. Therefore, kernel tricks are used. Kernel trick allows SVM to calculate higher dimensional coordinates in a lower dimension. There are different kernel functions such as polynomial, Gaussian (rbf) and sigmoid. These 3 kernel functions as well as linear kernel were tested in the system.

4.3.6 k-Nearest Neighbour

kNN is a traditional classification algorithm. As the name implies, it looks at the k points in the training set that is closest to the testing data. The distance is calculated by the Euclidian distance. For a feature vector of $X = x_1, x_2, ..., x_n$, and two points p_1 and p_2 Euclidian distance is calculated as:

$$d(p_1, p_2) = \sqrt{\sum_{i=1}^{n} (p_{1_{x_i}} - p_{2_{x_i}})^2} \tag{4.12}$$

Although kNN is a decent classifier, it suffers due to data requirement. The classifier tends to overfit as the dimensionality increases whilst the training samples remain limited. Due to limited amount of training data, kNN is expected to perform poorly compared to other classifiers. The classifier performance was tested on 6, 5, 4 and 3 nearest neighbours respectively. The results

will be discussed in later chapters.

4.4 Validation Method

Cross validation is a statistical procedure to measure performance of a classifier generalize to an independent dataset. Cross validation is made up of 4 steps. First, dataset is shuffled to uniformly distribute classes. Afterwards, dataset is split into k subsets and for each iteration each unique subset is chosen as testing data and the remaining subsets are used for training. Finally, average evaluation score is calculated using the scores from models trained in each iteration. 10-fold cross validation is used in this system. A sketch of 10-fold cross validation is shown in figure 4.8. Cross validation results will be discussed in results chapter.

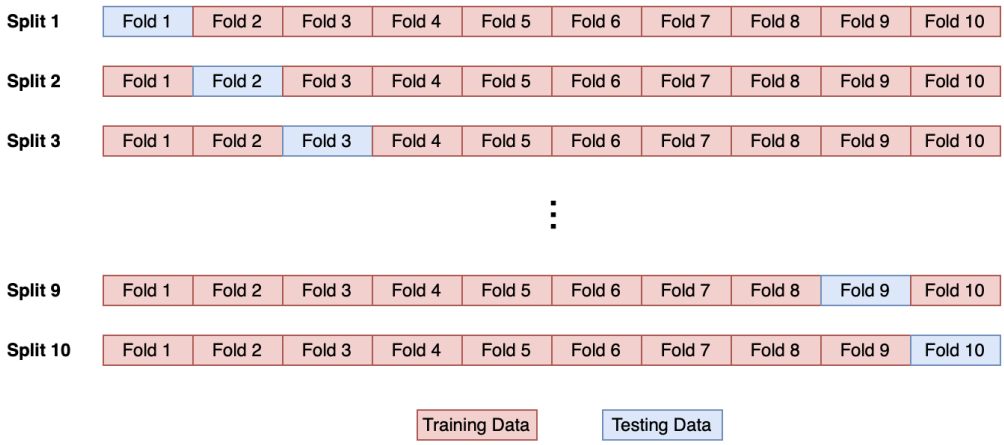


Figure 4.8: 10-fold Cross Validation

Chapter 5

Implementation

This chapter explains in detail how the seizure classification system was implemented according to the system architecture described in design chapter. Iterative, agile development cycle principles were followed in development of this project.

5.1 Data Pre-processing

Retrieving a dataset containing ECG and EMG information with their corresponding seizure types and times were the primary condition for this project. RADAR-CAN dataset met the data requirement as it contained multiple seizure types and their corresponding periods diagnosed by clinicians. As the language chosen to use in this project was Python and dataset was provided in MATLAB, dataset was required to be converted into .csv format. The code snippet in listing 5.1 has been implemented for every recording in the dataset. By using `writematrix()` function provided by Matlab API, dataset is converted to .csv format.

Listing 5.1: Algorithm to convert the dataset into .csv format

```
1  loadfold = [pwd '/' select_fold];
2  load([loadfold '/time_tab.mat']);
3  load([loadfold '/keepTime.mat']);
4  nfile = 10; % Number of files to download
5  for i = 1:nfile
6      data_tmp = load([loadfold '/data_FT2' num2str(i) '.mat']);
7      dataa = data_tmp.data_FT2.trial{1}(:,:);
8      timee = data_tmp.data_FT2.time{1}(:,:); % in seconds
```

```
9       fs(i,1) = data_tmp.data_FT2.fsample;
10      timevec = time_tab{i,3}+seconds(timee);
11      data_re{i,1} = dataa;
12      data_re{i,2} = timevec;
13      % convert to .csv
14      time_data = "KCL_**_time" + i + ".csv"
15      signal_data = "KCL_**_signal" + i + ".csv"
16      writematrix(data_re{i,1}, signal_data)
17      writematrix(data_re{i,2}, time_data)
18  end
```

This results in a different number of files for each recording due to length of the dataset. As some recordings contains 100+ hours of information, this results in 10+ files for those recordings. The rest of the coding is done in Python.

Second condition is cropping the seizure periods from the original data so that feature extraction can be applied globally on the cropped data. For each recording, a `kcl_**_cropdata.py` script was created and seizure periods were cropped manually by iterating through every file created by the previous code. For every seizure period, this script creates a .csv file consisting of every channel recorded during the seizure. As this script is created multiple times for each recording, only one of these files will be provided in the appendix.

5.2 Feature Extraction

In order to achieve accurate classification results feature extraction is one of the most important aspects. After in-depth literary review different features from each signal have been extracted. In order to do that, `kcl_**_feature_extraction.py` script was created for every recording. This script, basically opens every file in a recording an extract features from each signal provided. An array `seizure_feature` is appended for each cropped seizure period in a recording as well as its seizure type redcap provided by clinicians and then each feature vector with its corresponding type label is combined into a csv file. For time frequency domain feature extraction DWT is applied using PyWavelets, an open source module in Python under MIT license. For skewness and kurtosis statistics another open source Python module, SciPy [50]. In order to compute complex features such as entropy, median absolute deviation and mean absolute deviation functions named `calculate_entropy(list)`, `mad_median(list)`, `mad_mean(list)`

were implemented. For the remainder of statistical features Python module NumPy, licensed under BSD license, was used. After features were extracted, each feature vector is combined into a single csv file manually in order to use for classification.

5.3 Gaussian Noise

Gaussian noise is added to each signal before classification. It is required in order to test robustness of a classifier. Gaussian noise is a normally distributed list, having a mean of 0, and standard deviation as noise level. In order to add noise to each signal, a normal distribution with equal length to the signal was created and added to the original signal. 5 different noise levels were used to evaluate classifiers in this project, being 0.01, 0.05, 0.1, 0.2 and 0.5. NumPy module was used for the Gaussian noise implementation.

5.4 Classification

Implementation of effective classifiers to classify motor seizures is the primary objective of this project. Therefore, implementation and evaluation of different classifiers are crucial. In order to implement effective classifiers, Python module Scikit-learn is used. Scripts were implemented to train and test each classifier. Each classifier is tuned using the parameters provided by Scikit-learn API to perform better. Initially, each script reads a csv file to store the feature vectors and their corresponding targets in list. Seizure type redcaps as targets are corrected according to motor component inclusion iteratively. In order to perform PCA accurately, feature vectors are normalized using `StandardScaler()` class provided by Scikit-learn. Normalization is followed by PCA. Different number of PCs have been tested to improve the performance of the classifiers. 20 PC has given the best performance overall. Finally, classifiers are created and each classifier have been validated using 10-fold cross validation. Listing 5.2 shows the code implemented for cross validation. '0' is used for seizures with motor component whereas '1' is used for seizures without motor component as label. Local list variable `'scores'` is used to keep the classifier score in each iteration of cross validation and another local variable `'training_scores'` is used to keep track of the training scores of the classifier. Confusion matrix is another important aspect of machine learning evaluation. It is used to calculate specificity and sensitivity of the classifier. A local variable named `'matrix'` is used to sum up the confusion matrix created by each iteration. After validation trained models are saved as a .pkl file using Pickle module in Python.

Listing 5.2: Cross Validation

```
1 for i in range(10):
2     X_train, X_test, y_train, y_test = train_test_split(data, target
          , test_size=0.1,random_state=i)
3     classifier.fit(X_train,y_train)
4     training_scores.append(classifier.score(X_train,y_train))
5     scores.append(classifier.score(X_test,y_test))
6     y_pred = classifier.predict(X_test)
7     matrix += confusion_matrix(y_test, y_pred,labels=[0,1])
```

Chapter 6

Legal, Social, Ethical and Professional Issues

Method is designed to be used in a professional environment, medicine, thus there are many issues that should be considered.

6.1 Legal and Professional Issues

The British Computer Society (BCS), as a professional computer organisation, is acquainted with importance of many issues that affect all domains of life. To ensure beneficial outcomes to environment, computing industry has responsibilities. While method was developed and paper was written, the Code of Conduct and Code of Good Practice taken into account.

6.1.1 Plagiarism and Licensing

The presented project made use of numerous libraries and tools that are open-source, such as NumPy, Scikit-learn and MATLAB. All libraries and tools that has been used during project have distinct licenses yet all are open software resources and allowed to be used in this project.

6.2 Social and Ethical Issues

In IT, due to intense use of computers and increased dependence on technology, every action have a crucial role in real-world. Likewise, social aspect of computer technology allows further ethical problems. With the aim of aiming protection and advancement of human values, in-

tegration of computer technology and humanitarian approach, essentially, has been necessary [11]. Therefore, to ensure objectives and outcomes aim public good, an ethical approach should be appraised. The work has been done with the responsibility of public health, and well being of others while considering privacy and security rights. The retrieved data is collected anonymously and it protects privacy of subjects as it shall be. Purpose of project is designed with consideration of ethical problems and it ensures an ethical clearance with approve of the committee. Objective of the method was to detect seizures types as soon as possible and increase the number of precise diagnosis. Project is related with medicine field, and intention is based on public health and well-being which shows clear social, and ethical aspect of the project.

Chapter 7

Results and Evaluation

Accuracy of all the classifiers were evaluated using k-fold cross validation. To further extend the aim of this project it has been decided that it is not sufficient enough to only classify between motor and non-motor seizures. In a real life scenario it is also crucial to classify between a seizure and a physical activity. Tables 7.1 - 7.6 show the evaluation results of classifiers only classifying 2 classes which are motor and non-motor seizures. Tables 7.7 - 7.12 show the evaluation results of classifiers used to classify between a seizure and no seizure time period. Finally, tables 7.13 - 7.18 present classification of 3 classes being non-motor seizure, motor seizure and seizure-free. In order to create signals similar to a real-life scenario and evaluate robustness of the classifiers data has been contaminated with different levels (0.01, 0.05, 0.1, 0.2 and 0.5) of Gaussian noise. Features were extracted from these contaminated signals the same way. After feature extraction, models trained on clean signals were used for classification of these signals. For table 7.13 - 7.18 classes 1, 2 and 3 represent motor seizure, non-motor seizure and seizure-free periods respectively. In each table SVM represents Support Vector Machine, and #-NN represents #-nearest neighbour. Kernels used for SVM are shown with a '-' next to corresponding SVM. Tables 7.1, 7.7 and 7.13 show average, best, worst and standard deviation for both training and testing results. In addition to that, specificity and sensitivity of testing accuracy have been given. Sensitivity and specificity are calculated by the following equations:

$$Specificity = TN/(TN + FP) \tag{7.1}$$

$$Sensitivity = TP/(TP + FN) \tag{7.2}$$

where TN is true negatives, FP is false positives, TP is true positives and FN is false negatives. For seizure type classification SVM with Gaussian kernel outperforms other classifiers with 87.14% testing average for classification of seizure types with no noise whilst SVM with polynomial kernel performs the worst with an average of 52.86%. The range between the best individual result and worst individual result of each classifier is believed to be high because of the small data size. As it can be seen from the summary of the tables with Gaussian noise, the result of the classifiers were lower than expected and there might be several reasons of this. First of all, added noises may distort results due to limited amount of data. On the other hand, without regarding data it is conceivable to take notice of another system that affect the results. Previously mentioned added noises can result in stochastic resonance [26]. As tables 7.7 - 7.12 show, each classifier has evaluated the same data with and without gaussian noise. When data is contaminated with Gaussian noise there was fluctuation in the results which is a common outcome of stochastic resonance. Added noises could make signals more detectable by resonating with each other. Additionally, as it is demonstrated in tables with Gaussian noise, results are nonlinear. For instance, in tables 7.2 - 7.4 for noise levels 0.01, 0.05, 0.1, 0.2 and 0.5 performance of SVM with Gaussian (rbf) kernel is 70.97%, 58.06%, 66.13%, 59.35% and 58.06% respectively. This shows that there is fluctuation rather than a linear increase or decrease. This is opposite of known literature which indicates level of noise and classification performance should have negative correlation [12, 21, 42]. By following same phenomenon, signal to noise ratio could be the reason of lower results. Gaussian signal was added proportionately for each signal using the equation:

$$Signal_{noisy} = Signal_{normal} + Random.normal(0, n) * Signal_{normal} \tag{7.3}$$

where n is amount of noise, in other words standard deviation of the normal distribution. Even though noises are added proportionately there might be some disruptions. Noise has been added proportionately because as it can be seen from the figures 4.1, 4.2, 4.3 and 4.4 in chapter 4, these signals have small amplitudes. Addition of a normal distribution to these signals without proportioning to amplitude would make these signals invisible. However, tables 7.7 and 7.13 show that the models cannot classify physical activity from seizures well. Table 7.7 shows average score for classification of seizure and no seizure periods and the maximum average score is 61.17% with SVM with Sigmoid kernel. As the classification problem gets more complex in 7.13 with signals divided into 3 classes, the accuracy of the classifiers decrease even further. The most successful classifiers are 4-NN, 6-NN and SVM with Linear kernel with

average score of 53.33% each. In general, models were more successful at correctly classifying motor and seizure-free periods for classification of 3 classes.

Table 7.1: Summary of testing classification accuracy (%) for original dataset

Noise = 0	**Training**				**Testing**					
Classifier	Avg	Best	Worst	Std	Avg	Best	Worst	Std	Specificity	Sensitivity
Decision Tree	100	100	100	0	65.71	85.71	42.86	0.17	0.68	0.64
Naive Bayes	96.36	100	90.91	0.02	72.86	100	42.86	0.19	0.68	0.78
Neural Network	100	100	100	0	81.43	100	57.14	0.14	0.82	0.81
Gradient Boosting	100	100	100	0	71.43	100	28.57	0.18	0.68	0.75
SVM - Linear	100	100	100	0	78.57	100	57.14	0.15	0.82	0
SVM - Polynomial	90.91	98.18	78.18	0.07	52.86	85.71	14.29	0.23	0.53	0.53
SVM - rbf	98.73	100	98.18	0.01	87.14	100	71.43	0.12	0.88	0.86
SVM - Sigmoid	87.45	92.73	80	0.03	78.57	100	42.86	0.18	0.94	0.64
6-NN	92	92.73	89.09	0.01	84.29	100	57.14	0.12	0.82	0.86
5-NN	91.09	92.73	87.27	0.02	85.71	100	57.14	0.13	0.79	0.92
4-NN	90.73	94.55	87.27	0.02	84.29	100	71.43	0.12	0.85	0.83
3-NN	92.36	94.55	89.09	0.02	82.86	100	57.14	0.14	0.79	0.86

Table 7.2: Summary of testing classification accuracy (%) for dataset subject to noise level 0.01.

Noise = 0.01			
Classifier	Score	Specificity	Sensitivity
Decision Tree	67.74	0.56	0.8
Naive Bayes	70.97	0.78	0.63
Neural Network	62.9	0.78	0.47
Gradient Boosting	79.03	0.88	0.7
SVM - Linear	69.35	0.72	0.67
SVM - Polynomial	66.13	0.5	0.83
SVM - rbf	70.97	0.81	0.6
SVM - Sigmoid	67.74	0.78	0.57
6-NN	62.9	0.66	0.6
5-NN	58.06	0.53	0.63
4-NN	61.29	0.62	0.6
3-NN	61.29	0.56	0.67

Table 7.3: Summary of testing classification accuracy (%) for dataset subject to noise level 0.05.

Noise = 0.05			
Classifier	Score	Specificity	Sensitivity
Decision Tree	59.68	0.56	0.63
Naive Bayes	51.61	0.59	0.43
Neural Network	58.06	0.62	0.53
Gradient Boosting	56.45	0.62	0.5
SVM - Linear	48.39	0.47	0.5
SVM - Polynomial	67.74	0.5	0.87
SVM - rbf	58.06	0.72	0.43
SVM - Sigmoid	51.61	0.62	0.4
6-NN	69.35	0.62	0.77
5-NN	72.58	0.59	0.87
4-NN	67.74	0.66	0.7
3-NN	70.97	0.66	0.77

Table 7.4: Summary of testing classification accuracy (%) for dataset subject to noise level 0.1.

Noise = 0.1			
Classifier	Score	Specificity	Sensitivity
Decision Tree	75.81	0.72	0.8
Naive Bayes	74.19	0.69	0.8
Neural Network	83.87	0.91	0.77
Gradient Boosting	70.97	0.75	0.67
SVM - Linear	66.13	0.53	0.8
SVM - Polynomial	70.97	0.59	0.83
SVM - rbf	66.13	0.72	0.6
SVM - Sigmoid	77.42	0.84	0.7
6-NN	83.87	0.81	0.87
5-NN	83.87	0.78	0.9
4-NN	83.87	0.81	0.87
3-NN	82.26	0.75	0.9

Table 7.5: Summary of testing classification accuracy (%) for dataset subject to noise level 0.2.

Noise = 0.2			
Classifier	Score	Specificity	Sensitivity
Decision Tree	85.48	0.84	0.87
Naive Bayes	67.74	0.78	0.57
Neural Network	62.9	0.66	0.6
Gradient Boosting	72.58	0.78	0.67
SVM - Linear	58.06	0.56	0.6
SVM - Polynomial	38.71	0.16	0.63
SVM - rbf	59.35	0.69	0.7
SVM - Sigmoid	64.52	0.75	0.53
6-NN	70.97	0.72	0.7
5-NN	66.13	0.56	0.77
4-NN	72.58	0.72	0.73
3-NN	69.35	0.59	0.8

Table 7.6: Summary of testing classification accuracy (%) for dataset subject to noise level 0.5.

Noise = 0.5			
Classifier	Score	Specificity	Sensitivity
Decision Tree	38.71	0.38	0.4
Naive Bayes	41.94	0.47	0.37
Neural Network	41.94	0.47	0.37
Gradient Boosting	38.71	0.44	0.33
SVM - Linear	46.77	0.44	0.5
SVM - Polynomial	38.71	0.09	0.7
SVM - rbf	58.06	0.56	0.6
SVM - Sigmoid	41.94	0.44	0.4
6-NN	45.16	0.47	0.43
5-NN	48.39	0.44	0.53
4-NN	51.61	0.56	0.47
3-NN	53.23	0.5	0.57

Table 7.7: Summary of testing detection accuracy (%) of seizures for original dataset.

Noise = 0	**Training**				**Testing**					
Classifier	Avg	Best	Worst	Std	Avg	Best	Worst	Std	Specificity	Sensitivity
Decision Tree	100	100	100	0	47.5	66.67	16.67	0.13	0.47	0.48
Naive Bayes	64.45	68.69	58.59	0.03	45.83	66.67	16.67	0.17	0.81	0.13
Neural Network	100	100	100	0	54.17	83.33	33.33	0.14	0.57	0.52
Gradient Boosting	100	100	100	0	50.83	75	33.33	0.13	0.55	0.47
SVM - Linear	81.82	85.86	77.78	0.02	56.67	75	25	0.14	0.64	0.5
SVM - Polynomial	64.95	69.7	61.62	0.02	49.17	91.67	16.67	0.21	0.95	0.06
SVM - rbf	82.32	89.9	79.8	0.03	51.67	66.67	33.33	0.1	0.62	0.42
SVM - Sigmoid	71.43	73.74	68.69	0.02	61.67	83.33	33.33	0.15	0.76	0.49
6-NN	73.33	75.76	68.69	0.02	58.33	83.33	33.33	0.14	0.76	0.42
5-NN	78.69	82.83	74.75	0.03	58.33	83.33	33.33	0.15	0.67	0.5
4-NN	78.79	82.83	76.77	0.02	56.67	66.67	41.67	0.1	0.76	0.39
3-NN	83.74	85.86	80.81	0.01	57.5	75	41.67	0.14	0.6	0.55

Table 7.8: Summary of testing detection accuracy (%) of seizures for dataset subject to noise level 0.01.

Noise = 0.01			
Classifier	Score	Specificity	Sensitivity
Decision Tree	36.94	0.4	0.33
Naive Bayes	49.55	0.74	0.18
Neural Network	45.05	0.5	0.39
Gradient Boosting	44.14	0.5	0.37
SVM - Linear	44.14	0.56	0.29
SVM - Polynomial	55.86	1	0
SVM - rbf	42.34	0.55	0.27
SVM - Sigmoid	38.74	0.56	0.16
6-NN	40.54	0.56	0.16
5-NN	42.34	0.5	0.33
4-NN	39.64	0.63	0.1
3-NN	41.44	0.55	0.25

Table 7.9: Summary of testing detection accuracy (%) of seizures for dataset subject to noise level 0.05.

Noise = 0.05			
Classifier	Score	Specificity	Sensitivity
Decision Tree	65.77	0.65	0.67
Naive Bayes	61.26	0.85	0.31
Neural Network	62.16	0.77	0.43
Gradient Boosting	62.16	0.77	0.43
SVM - Linear	64.86	0.68	0.61
SVM - Polynomial	56.76	1	0.02
SVM - rbf	58.56	0.74	0.39
SVM - Sigmoid	62.16	0.69	0.53
6-NN	56.76	0.66	0.45
5-NN	53.35	0.5	0.53
4-NN	52.25	0.66	0.35
3-NN	47.75	0.5	0.45

Table 7.10: Summary of testing detection accuracy (%) of seizures for dataset subject to noise level 0.1.

Noise = 0.1			
Classifier	Score	Specificity	Sensitivity
Decision Tree	57.66	0.65	0.49
Naive Bayes	57.66	0.81	0.29
Neural Network	55.86	0.63	0.47
Gradient Boosting	60.36	0.71	0.47
SVM - Linear	60.36	0.73	0.45
SVM - Polynomial	58.56	1	0.06
SVM - rbf	63.96	0.81	0.43
SVM - Sigmoid	61.26	0.74	0.45
6-NN	57.66	0.82	0.27
5-NN	52.25	0.58	0.45
4-NN	57.66	0.81	0.29
3-NN	54.05	0.58	0.49

Table 7.11: Summary of testing detection accuracy (%) of seizures for dataset subject to noise level 0.2.

Noise = 0.2			
Classifier	Score	Specificity	Sensitivity
Decision Tree	52.25	0.48	0.58
Naive Bayes	47.75	0.65	0.27
Neural Network	57.66	0.69	0.43
Gradient Boosting	63.06	0.74	0.49
SVM - Linear	54.05	0.61	0.45
SVM - Polynomial	58.56	1	0.06
SVM - rbf	58.56	0.69	0.45
SVM - Sigmoid	53.15	0.58	0.47
6-NN	51.35	0.71	0.27
5-NN	56.76	0.65	0.47
4-NN	53.15	0.74	0.27
3-NN	48.65	0.52	0.45

Table 7.12: Summary of testing detection accuracy (%) of seizures for dataset subject to noise level 0.5.

Noise = 0.5			
Classifier	Score	Specificity	Sensitivity
Decision Tree	55.86	0.56	0.55
Naive Bayes	46.85	0.56	0.35
Neural Network	44.14	0.52	0.35
Gradient Boosting	53.15	0.63	0.41
SVM - Linear	45.05	0.55	0.33
SVM - Polynomial	56.76	0.55	0.33
SVM - rbf	51.35	0.65	0.35
SVM - Sigmoid	46.85	0.53	0.39
6-NN	47.75	0.61	0.31
5-NN	48.65	0.48	0.49
4-NN	42.34	0.61	0.18
3-NN	49.55	0.45	0.55

Table 7.13: Summary of testing classification accuracy (%) of seizures and types for original dataset. Class 1 = motor seizure, Class 2 = non motor seizure, Class 3 = seizure-free

Noise = 0	**Training**				**Testing**						
Classifier	Avg	Best	Worst	Std	Avg	Best	Worst	Std	Class 1	Class 2	Class 3
Decision Tree	100	100	100	0	43.33	66.67	8.33	0.15	0.39	0	0.28
Naive Bayes	68.98	71.72	60.61	0.04	40.83	58.33	8.33	0.16	0.36	0.12	0.36
Neural Network	100	100	100	0	46.67	75	25	0.16	0.5	0.03	0.27
Gradient Boosting	100	100	100	0	49.17	66.67	33.33	0.12	0.54	0.09	0.23
SVM - Linear	89.09	92.93	85.86	0.02	53.33	75	25	0.15	0.52	0.03	0.25
SVM - Polynomial	56.15	58.59	53.54	0.02	42.5	75	8.33	0.17	0	0.43	0.26
SVM - rbf	79.39	82.83	75.76	0.02	49.17	75	16.67	0.18	0.42	0.05	0.27
SVM - Sigmoid	61.01	63.64	58.59	0.02	47.5	66.67	25	0.14	0.42	0.05	0.28
6-NN	71.62	75.76	68.69	0.03	53.33	75	33.33	0.15	0.52	0.09	0.3
5-NN	73.43	79.8	69.7	0.03	52.5	75	33.33	0.17	0.52	0.1	0.3
4-NN	76.46	81.82	72.73	0.02	53.33	75	33.33	0.12	0.56	0.11	0.24
3-NN	79.8	83.84	75.76	0.02	52.5	75	25	0.17	0.58	0.13	0.21

Table 7.14: Summary of testing classification accuracy (%) of seizures and types for dataset subject to noise level 0.01. Class 1 = motor seizure, Class 2 = non motor seizure, Class 3 = seizure-free

Noise = 0.01				
Classifier	Avg	Class 1	Class 2	Class 3
Decision Tree	27.03	0.21	0.39	0.32
Naive Bayes	24.32	0.23	0.32	0.32
Neural Network	33.33	0.23	0.36	0.28
Gradient Boosting	27.93	0.22	0.33	0.3
SVM - Linear	27.03	0.21	0.38	0.29
SVM - Polynomial	38.74	0.21	0.36	0.29
SVM - rbf	37.84	0.21	0.36	0.29
SVM - Sigmoid	28.83	0.21	0.38	0.29
6-NN	36.94	0.24	0.34	0.29
5-NN	36.04	0.26	0.3	0.29
4-NN	34.23	0.28	0.28	0.3
3-NN	34.23	0.29	0.26	0.3

Table 7.15: Summary of testing classification accuracy (%) of seizures and types for dataset subject to noise level 0.05. Class 1 = motor seizure, Class 2 = non motor seizure, Class 3 = seizure-free

Noise = 0.05				
Classifier	Avg	Class 1	Class 2	Class 3
Decision Tree	44.14	0.3	0.41	0.25
Naive Bayes	46.85	0.35	0.25	0.28
Neural Network	45.95	0.38	0.28	0.25
Gradient Boosting	50.45	0.35	0.27	0.27
SVM - Linear	47.75	0.35	0.28	0.26
SVM - Polynomial	40.54	0.35	0.28	0.27
SVM - rbf	42.34	0.33	0.27	0.28
SVM - Sigmoid	34.23	0.31	0.28	0.29
6-NN	37.84	0.3	0.29	0.28
5-NN	36.94	0.29	0.3	0.28
4-NN	33.33	0.28	0.31	0.28
3-NN	29.73	0.26	0.31	0.29

Table 7.16: Summary of testing classification accuracy (%) of seizures and types for dataset subject to noise level 0.1. Class 1 = motor seizure, Class 2 = non motor seizure, Class 3 = seizure-free

Noise = 0.1				
Classifier	Avg	Class 1	Class 2	Class 3
Decision Tree	37.84	0.25	0.22	0.35
Naive Bayes	36.94	0.25	0.21	0.39
Neural Network	41.44	0.26	0.19	0.37
Gradient Boosting	49.55	0.27	0.19	0.36
SVM - Linear	46.85	0.27	0.19	0.36
SVM - Polynomial	40.54	0.27	0.2	0.33
SVM - rbf	49.55	0.28	0.2	0.33
SVM - Sigmoid	47.75	0.29	0.19	0.32
6-NN	44.14	0.3	0.2	0.33
5-NN	40.54	0.29	0.21	0.33
4-NN	43.24	0.29	0.21	0.33
3-NN	44.14	0.3	0.21	0.32

Table 7.17: Summary of testing classification accuracy (%) of seizures and types for dataset subject to noise level 0.2. Class 1 = motor seizure, Class 2 = non motor seizure, Class 3 = seizure-free

Noise = 0.2				
Classifier	Avg	Class 1	Class 2	Class 3
Decision Tree	34.23	0.2	0.35	0.32
Naive Bayes	33.33	0.23	0.25	0.35
Neural Network	39.64	0.21	0.27	0.34
Gradient Boosting	40.54	0.21	0.27	0.33
SVM - Linear	37.84	0.22	0.26	0.33
SVM - Polynomial	38.74	0.21	0.28	0.32
SVM - rbf	33.33	0.2	0.28	0.32
SVM - Sigmoid	36.04	0.21	0.29	0.31
6-NN	39.64	0.22	0.27	0.32
5-NN	43.24	0.23	0.27	0.32
4-NN	40.54	0.24	0.26	0.32
3-NN	34.23	0.24	0.26	0.32

Table 7.18: Summary of testing classification accuracy (%) of seizures and types for dataset subject to noise level 0.5. Class 1 = motor seizure, Class 2 = non motor seizure, Class 3 = seizure-free

Noise = 0.5				
Classifier	Avg	Class 1	Class 2	Class 3
Decision Tree	28.83	0.33	0.14	0.31
Naive Bayes	28.83	0.28	0.18	0.34
Neural Network	27.03	0.27	0.18	0.34
Gradient Boosting	29.73	0.28	0.19	0.34
SVM - Linear	29.73	0.28	0.22	0.32
SVM - Polynomial	37.84	0.28	0.23	0.31
SVM - rbf	37.84	0.29	0.22	0.31
SVM - Sigmoid	34.23	0.29	0.21	0.31
6-NN	32.43	0.3	0.23	0.3
5-NN	34.23	0.3	0.24	0.3
4-NN	27.93	0.31	0.25	0.3
3-NN	34.23	0.31	0.26	0.29

Chapter 8

Conclusion and Future Work

8.1 Conclusion

This project presented various classification algorithms for seizure type classification using multi-modal system. Literature was analysed deeply for feature extraction from each signal type. From the analysis it has been decided that time-frequency analysis is important for ECG, EMG and GSR signals whereas it was unnecessary for ACC. Classification results show that the current dataset is not big enough for accurate classification. For data augmentation, Generative Adversarial Network can be implemented in the future. Results from the classifiers show that SVM based classifiers outperform others when the data is noise free. As the data is contaminated with noise, classifiers tend to vary in performance. This is likely to be due to stochastic resonance in the noise addition process. However, classification results for noise free dataset shows promising results as most of the classifiers show 80% accuracy and all of them except SVM with Polynomial kernel and decision tree can reach 100% accuracy for a specific split.

8.2 Future Work

As there is no example of seizure type classification in literature, project results cannot be directly compared to another work. As it can be seen from the tables in results chapter classifier performance is not sufficient enough to decide that seizure types can be classified precisely using machine learning methods. Therefore, there is a need for further use and evaluation of different classifiers in this subject. Fuzzy learning methods as well as self organising maps can be used

for robustness in the future. If successful, the trained models can be implemented to wearable accessories such as watched for seizure detection without going to a clinic or epilepsy monitoring unit. As current wearable devices such as some watches can already measure some of these signals used in this project, it can be developed and enhanced to a real life state of the art solution for seizure classification.

References

[1] Diagnosing seizures and epilepsy. "https://www.hopkinsmedicine.org/health/conditions-and-diseases/epilepsy/diagnosing-seizures-and-epilepsy", 2018. [Online; accessed 10-December-2019].

[2] Epilepsy and seizures: Conditions we treat — the johns hopkins epilepsy center. "https://www.hopkinsmedicine.org/neurology_neurosurgery/centers_clinics/epilepsy/conditions.html", 2018. [Online; accessed 10-December-2019].

[3] Types of seizures. "https://www.hopkinsmedicine.org/health/conditions-and-diseases/epilepsy/types-of-seizures", 2018. [Online; accessed 10-December-2019].

[4] Understanding gradient boosting machines. "https://towardsdatascience.com/understanding-gradient-boosting-machines-9be756fe76ab", 2018. [Online; accessed 18-April-2020].

[5] Epilepsy Action. Epilepsy-related deaths and sudep. "https://www.epilepsy.org.uk/info/sudep-sudden-unexpected-death-in-epilepsy", 2019. [Online; accessed 19-April-2020].

[6] James G Anderson. Social, ethical and legal barriers to e-health. *International journal of medical informatics*, 76(5-6):480–483, 2007.

[7] Syed Muhammad Anwar, Maheen Gul, Muhammad Majid, and Majdi Alnowami. Arrhythmia classification of ecg signals using hybrid features. *Computational and mathematical methods in medicine*, 2018, 2018.

[8] Sándor Beniczky, Isa Conradsen, Ronit Pressler, and Peter Wolf. Quantitative analysis of surface electromyography: biomarkers for convulsive seizures. *Clinical Neurophysiology*, 127(8):2900–2907, 2016.

[9] Isa Conradsen, Sándor Beniczky, Peter Wolf, Jonas Henriksen, Thomas Sams, and Helge BD Sorensen. Seizure onset detection based on a uni-or multi-modal intelligent seizure acquisition (uisa/misa) system. In *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, pages 3269–3272. IEEE, 2010.

[10] Thomas De Cooman, Anouk Van de Vel, Berten Ceulemans, Lieven Lagae, Bart Vanrumste, and Sabine Van Huffel. Online detection of tonic-clonic seizures in pediatric patients using ecg and low-complexity incremental novelty detection. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 5597–5600. IEEE, 2015.

[11] Gordana Dodig-Crnkovic and Robert Feldt. Professional and ethical issues of software engineering curricula. *2010*, 2009.

[12] Udeme Ekong, Hak-Keung Lam, Bo Xiao, Gaoxiang Ouyang, Hongbin Liu, Kit Yan Chan, and Sai Ho Ling. Classification of epilepsy seizure phase using interval type-2 fuzzy support vector machines. *Neurocomputing*, 199:66–76, 2016.

[13] Elaine Kiriakopoulos and Patricia O. Shafer. Types of seizures. "https://www.epilepsy.com/learn/types-seizures", 2017. [Online; accessed 10-December-2019].

[14] Robert S Fisher, J Helen Cross, Carol D'souza, Jacqueline A French, Sheryl R Haut, Norimichi Higurashi, Edouard Hirsch, Floor E Jansen, Lieven Lagae, Solomon L Moshé, et al. Instruction manual for the ilae 2017 operational classification of seizure types. *Epilepsia*, 58(4):531–542, 2017.

[15] Keider Hoyos-Osorio, Jairo Castañeda-Gonzaiez, and Genaro Daza-Santacoloma. Automatic epileptic seizure prediction based on scalp eeg and ecg signals. In *2016 XXI Symposium on signal processing, images and artificial vision (STSIVA)*, pages 1–7. IEEE, 2016.

[16] Andrey Ignatov. Real-time human activity recognition from accelerometer data using convolutional neural networks. *Applied Soft Computing*, 62:915–922, 2018.

[17] SUDEP Institute. Sudep. "https://www.epilepsy.com/learn/early-death-and-sudep/sudep", 2013. [Online; accessed 10-December-2019].

[18] Dongni Johansson, Fredrik Ohlsson, David Krỳsl, Bertil Rydenhag, Madeleine Czarnecki, Niclas Gustafsson, Jan Wipenmyr, Tomas McKelvey, and Kristina Malmgren. Tonic-clonic

seizure detection using accelerometry-based wearable sensors: A prospective, video-eeg controlled study. *Seizure*, 65:48–54, 2019.

[19] Elaine Kiriakopoulos. Focal onset aware seizures (simple partial seizures). "https://www.epilepsy.com/learn/types-seizures/focal-onset-aware-seizures-aka-simple-partial-seizures", 2017. [Online; accessed 10-December-2019].

[20] Narayanan C Krishnan and Sethuraman Panchanathan. Analysis of low resolution accelerometer data for continuous human activity recognition. In *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 3337–3340. IEEE, 2008.

[21] Hak-Keung Lam, Udeme Ekong, Bo Xiao, Gaoxiang Ouyang, Hongbin Liu, Kit Yan Chan, and Sai Ho Ling. Variable weight neural networks and their applications on material surface and epilepsy seizure phase classifications. *Neurocomputing*, 149:1177–1187, 2015.

[22] Frans SS Leijten, Dutch TeleEpilepsy Consortium, J van Andel, C Ungureanu, J Arends, F Tan, J van Dijk, G Petkov, S Kalitzin, T Gutter, et al. Multimodal seizure detection: A review. *Epilepsia*, 59:42–47, 2018.

[23] Tobias Loddenkemper and Prakash Kotagal. Lateralizing signs during seizures in focal epilepsy. *Epilepsy & Behavior*, 7(1):1–17, 2005.

[24] Roshan Joy Martis, M Muthu Rama Krishnan, Chandan Chakraborty, Sarbajit Pal, Debranjan Sarkar, KM Mandana, and Ajoy Kumar Ray. Automated screening of arrhythmia using wavelet based machine learning techniques. *Journal of medical systems*, 36(2):677–688, 2012.

[25] Ahmet Mert. Ecg feature extraction based on the bandwidth properties of variational mode decomposition. *Physiological measurement*, 37(4):530, 2016.

[26] Sanya Mitaim and Bart Kosko. Adaptive stochastic resonance. *Proceedings of the IEEE*, 86(11):2152–2183, 1998.

[27] Iosif Mporas, Vasiliki Tsirka, Evangelia I Zacharaki, Michalis Koutroumanidis, and Vasileios Megalooikonomou. Online seizure detection from eeg and ecg signals for monitoring of epileptic patients. In *Hellenic Conference on Artificial Intelligence*, pages 442–447. Springer, 2014.

[28] Iosif Mporas, Vasiliki Tsirka, Evangelia I Zacharaki, Michalis Koutroumanidis, Mark Richardson, and Vasileios Megalooikonomou. Seizure detection using eeg and ecg signals for computer-based monitoring, analysis and management of epileptic patients. *Expert systems with applications*, 42(6):3227–3233, 2015.

[29] Truong Nguyen. *Wavelets and filter banks*. Wellesley-Cambridge Press, 1996.

[30] Ben Nham, Kanya Siangliulue, and Serena Yeung. Predicting mode of transport from iphone accelerometer data. *Standford University Class Project*, 2008.

[31] Tamara ME Nijsen, Ronald M Aarts, Johan BAM Arends, and Pierre JM Cluitmans. Automated detection of tonic seizures using 3-d accelerometry. In *4th European conference of the international federation for medical and biological engineering*, pages 188–191. Springer, 2009.

[32] Tamara ME Nijsen, Ronald M Aarts, Pierre JM Cluitmans, and Paul AM Griep. Time-frequency analysis of accelerometry data for detection of myoclonic seizures. *IEEE Transactions on Information Technology in Biomedicine*, 14(5):1197–1203, 2010.

[33] Tamara ME Nijsen, Johan BAM Arends, Paul AM Griep, and Pierre JM Cluitmans. The potential value of three-dimensional accelerometry for detection of motor seizures in severe epilepsy. *Epilepsy & Behavior*, 7(1):74–84, 2005.

[34] Maria Meritxell Oto. The misdiagnosis of epilepsy: appraising risks and managing uncertainty. *Seizure*, 44:143–146, 2017.

[35] Nilufer Ozdemir and Esen Yildirim. Patient specific seizure prediction system using hilbert spectrum and bayesian networks classifiers. *Computational and mathematical methods in medicine*, 2014, 2014.

[36] Yun Park, Lan Luo, Keshab K Parhi, and Theoden Netoff. Seizure prediction with spectral power of eeg using cost-sensitive support vector machines. *Epilepsia*, 52(10):1761–1770, 2011.

[37] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[38] Angkoon Phinyomark, Chusak Limsakul, and Pornchai Phukpattaranont. Application of wavelet analysis in emg feature extraction for pattern classification. *Measurement Science Review*, 11(2):45–52, 2011.

[39] Angkoon Phinyomark, Asan Nuidod, Pornchai Phukpattaranont, and Chusak Limsakul. Feature extraction and reduction of wavelet transform coefficients for emg pattern classification. *Elektronika ir Elektrotechnika*, 122(6):27–32, 2012.

[40] Angkoon Phinyomark, Pornchai Phukpattaranont, and Chusak Limsakul. Feature reduction and selection for emg signal classification. *Expert systems with applications*, 39(8):7420–7431, 2012.

[41] Ming-Zher Poh, Tobias Loddenkemper, Nicholas C Swenson, Shubhi Goyal, Joseph R Madsen, and Rosalind W Picard. Continuous monitoring of electrodermal activity during epileptic seizures using a wearable sensor. In *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, pages 4415–4418. IEEE, 2010.

[42] Khurram I Qazi, HK Lam, Bo Xiao, Gaoxiang Ouyang, and Xunhe Yin. Classification of epilepsy using computational intelligence techniques. *CAAI Transactions on Intelligence Technology*, 1(2):137–149, 2016.

[43] Nishkam Ravi, Nikhil Dandekar, Preetham Mysore, and Michael L Littman. Activity recognition from accelerometer data. In *Aaai*, volume 5, pages 1541–1546, 2005.

[44] Chia-Ping Shen, Shih-Ting Liu, Wei-Zhi Zhou, Feng-Seng Lin, Andy Yan-Yu Lam, Hsiao-Ya Sung, Wei Chen, Jeng-Wei Lin, Ming-Jang Chiu, Ming-Kai Pan, et al. A physiology-based seizure detection system for multichannel eeg. *PloS one*, 8(6):e65862, 2013.

[45] Pekka Siirtola and Juha Röning. Recognizing human activities user-independently on smartphones based on accelerometer data. *IJIMAI*, 1(5):38–45, 2012.

[46] Charles Ákos Szabó, Lola C Morgan, Kameel M Karkar, Linda D Leary, Octavian V Lie, Michael Girouard, and José E Cavazos. Electromyography-based seizure detector: preliminary results comparing a generalized tonic–clonic seizure detection algorithm to video-eeg recordings. *Epilepsia*, 56(9):1432–1437, 2015.

[47] Manu Thomas, Manab Kr Das, and Samit Ari. Automatic ecg arrhythmia classification using dual tree complex wavelet based features. *AEU-International Journal of Electronics and Communications*, 69(4):715–721, 2015.

[48] A Ulate-Campos, F Coughlin, M Gaínza-Lein, I Sánchez Fernández, PL Pearl, and T Loddenkemper. Automated seizure detection systems and their effectiveness for each type of seizure. *Seizure*, 40:88–101, 2016.

[49] Anouk Van de Vel, Kris Cuppens, Bert Bonroy, Milica Milosevic, Katrien Jansen, Sabine Van Huffel, Bart Vanrumste, Patrick Cras, Lieven Lagae, and Berten Ceulemans. Non-eeg seizure detection systems and potential sudep prevention: state of the art: review and update. *Seizure*, 41:141–153, 2016.

[50] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, CJ Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake Vand erPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1. 0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.

[51] Braxton B Wannamaker. Autonomic nervous system and epilepsy. *Epilepsia*, 26:S31–S39, 1985.

[52] Che-Chang Yang and Yeh-Liang Hsu. A review of accelerometry-based wearable motion detectors for physical activity monitoring. *Sensors*, 10(8):7772–7788, 2010.

[53] Xiaodong Zhang, Weifeng Diao, and Zhiqiang Cheng. Wavelet transform and singular value decomposition of eeg signal for pattern recognition of complicated hand activities. In *International Conference on Digital Human Modeling*, pages 294–303. Springer, 2007.

[54] Maeike Zijlmans, Danny Flanagan, and Jean Gotman. Heart rate changes and ecg abnormalities during epileptic seizures: prevalence and definition of an objective clinical sign. *Epilepsia*, 43(8):847–854, 2002.

Appendix A

User Guide

A.1 Install Python

A.2 Install Pip

A.3 Install Modules

A.4 Run Scripts

Appendix B

Source Code

B.1 MatLab Conversion

B.1.1 convert_to_csv.m

B.2 Cropping Seizures

B.2.1 crop_data.py

B.3 Feature Extraction

B.3.1 feature_extraction.py

B.3.2 noisy_feature_extraction.py

B.4 Training and Evaluation

B.4.1 decision_tree_classifier.py

B.4.2 neural_network_classifier.py

B.4.3 gradient_boosting_classifier.py

B.4.4 naive_bayes_classifier.py

B.4.5 svm_classifier.py

B.4.6 knn_classifier.py

B.5 Robustness

B.5.1 Type Classification

B.5.2 Seizure Classification

B.5.3 Seizure and Type Classification