Circuits and Systems
Mekelweg 4,
2628 CD Delft
The Netherlands
https://sps.ewi.tudelft.nl/

SPS-2024-00

M.Sc. Thesis

Automated Epilepsy Diagnosis beyond IEDs by Multimodal Features and Deep Learning

Yash Mirwani

Abstract

Automated diagnosis of epilepsy for differentiating epileptic EEGs without Interictal Epileptic Discharges (IEDs) from normal EEGs remains a critical challenge in clinical settings. Current state-of-the-art methods use algorithms that can effectively detect epilepsy seizures which improves the current treatment methods for people suffer from epilepsy. Electroencephalograms (EEGs) analyzed by neurologists which are not able to meet the criteria are further looked into to obtain an efficient classification. This thesis presents an automated epilepsy diagnosis approach using a multi-algorithmic feature extraction pipeline. The final models include the development of a robust multi-processing feature extraction pipeline, the application of advanced machine learning / deep learning algorithms, and the validation of the proposed methods on comprehensive EEG datasets. The results, achieved using an XGBoost classifier with leave-one-subject-out (LOSO) cross-validation, demonstrate comparable performance to state-of-the-art epilepsy detectors. The study emphasizes the detection of epilepsy without IEDs, optimizing models through nested cross-validation, and evaluates their performance on the Temple University Hospital (TUH) and Erasmus Medical Center (EMC) Rotterdam datasets.

Automated Epilepsy Diagnosis beyond IEDs by Multimodal Features and Deep Learning

Thesis

submitted in partial fulfillment of the
requirements for the degree of

Master of Science

in

Electrical Engineering

by

Yash Mirwani
born in Philipsburg, St. Maarten

This work was performed in:

Circuits and Systems Group
Department of Signals & Systems
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology

Delft University of Technology

Delft University of Technology
Department of
Signals & Systems

The undersigned hereby certify that they have read and recommend to the Faculty of Electrical Engineering, Mathematics and Computer Science for acceptance a thesis entitled **"Automated Epilepsy Diagnosis beyond IEDs by Multimodal Features and Deep Learning"** by **Yash Mirwani** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: August 26, 2024

Chairman:

prof.dr.ir. Justin Dauwels

Advisor:

prof.dr.ir. Justin Dauwels

Committee Members:

prof.dr.ir. Justin Dauwels

prof.dr.ir. Wouter Serdijn

dr. Robert van den Berg

Abstract

Automated diagnosis of epilepsy for differentiating epileptic EEGs without Interictal Epileptic Discharges (IEDs) from normal EEGs remains a critical challenge in clinical settings. Current state-of-the-art methods use algorithms that can effectively detect epilepsy seizures which improves the current treatment methods for people suffer from epilepsy. Electroencephalograms (EEGs) analyzed by neurologists which are not able to meet the criteria are further looked into to obtain an efficient classification. However, this manual process can be time-consuming and prone to errors. The main objectives of this research include the development of a robust multi-processing feature extraction pipeline, the application of VGG16 model / XGBoost classifier, and the validation of the proposed methods on comprehensive EEG datasets. Specifically, the focus is on detecting epilepsy in EEG data without Interictal Epileptic Discharges (IEDs) which poses a significant challenge due to the complex nature of the EEG signals in such cases.

This thesis presents an automated epilepsy diagnosis approach using a multi-algorithmic feature extraction pipeline. The final models include the development of a multi-processing feature extraction pipeline, the application of advanced machine learning / deep learning algorithms, and the validation of the proposed methods on comprehensive EEG datasets. The results, achieved using an XGBoost classifier with leave-one-subject-out (LOSO) cross-validation, demonstrate comparable performance to state-of-the-art epilepsy detectors. The study emphasizes the detection of epilepsy without IEDs, optimizing models through nested cross-validation, and evaluates their performance on the Temple University Hospital (TUH) and Erasmus MC (EMC) Rotterdam datasets.

Acknowledgments

First and foremost, I would like to express my deep gratitude to my thesis supervisor Dr. Justin Dauwels for his guidance, support and insightful critiques throughout my research journey. His mentorship has been crucial in directing my thesis to a successful completion with countless invaluable lessons. The guidance and motivation he provided during the first few months helped me a lot to structure the thesis based on my current skills. I would also like to thank Dr. van den Berg for his ability to offer innovative solutions towards this complex problem. Our meetings helped me understand the clinical significance and relevance of the models. Without their expertise and encouragement, this work would not have been possible.

I would also like to extend my appreciation to my committee members for their valuable feedback and suggestions that significantly enhanced the quality of my research. My heartfelt thanks go to my family and friends. To my parents and my sister, Vanisha, for their unwavering support, endless patience, and unconditional love throughout this journey. Your belief in me has been a constant source of motivation. Special thanks to my colleagues, R. Moesman, M. Rom, A. Kannan, and P. van der Kleij for their constant support and my colleagues at Praxa Sense during my internship for advice and comfort. Your companionship and encouragement have been invaluable.

Finally, I would like to thank SPS department & EMC Rotterdam for providing the resources and facilities that made this research possible and to all the staff members who offered their assistance throughout my studies.
Thank you all for your unwavering support and encouragement.

Yash Mirwani
Delft, The Netherlands
August 26, 2024

Contents

List of Figures

List of Tables

Nomenclature

AUC	Area Under the Curve
BAC	Balanced Accuracy
CAR	Common Average Referential
CLI	Command line interface
CNN	Convolutional Neural Network
CV	Cross-Validation
CWT	Continuous Wavelet Transform
C-C	Connectivity—Cross Correlation
C-PLV	Connectivity—Phase Lock Value
db4	Daubechies Wavelet
DWT	Discrete Wavelet Transform
EDF	European Data Format
EEG	Electroencephalogram
Epileptic EEG	EEG from Patient Diagnosed with Epilepsy
EEG w IEDs	Patient whose EEGs Exhibit IED Patterns
EEG w/o IEDs	Patient whose EEGs Do Not Exhibit IED Patterns
EMC	Erasmus Medical Center
Ep	Number of epileptic patients
FFT	Fast Fourier transform
FN	False Negatives
FP	False Positives
IED	Interictal Epileptiform Discharge
IED-independent features	Computed from the EEG without Expert Annotated IED Segments
LOIO	Leave-One-Institution-Out
LOSO	Leave-One-Subject-Out
morl	Morlet Wavelet
Non-epileptic EEG	Normal EEG with No Marked Abnormality in the Clinical Report
Np	Number of healthy patients
ROC	Receiver Operating Characteristics
ST	Stockwell Transform
STFT	Short-Time Fourier Transform
SVM	Support vector machine
TP	True Positives
TN	True Negatives
tqdm	Progress meter bar in CLI
TUH	Temple University Hospital
UTM	Univariate Temporal Measures
XGBoost	eXtreme Gradient Boosting
XAI	Explainable AI

1 Introduction

1.1 Epilepsy Diagnosis

Epilepsy is a long-term brain condition where individuals have repeated seizures caused by brain activity and it affects 50 million people globally [1]. Doctors can identify epilepsy using an electroencephalogram (EEG) i.e. a method that records the brain's activity through electrodes placed on the scalp. According to the World Health Organization, experiencing one seizure does not automatically indicate epilepsy; 10% of individuals may have a seizure at some point in their lives [1]. Moreover, epilepsy is diagnosed after two or more seizures occur over a period of time. Early diagnosis following the seizure is very important for determining the effective long term treatment options. In regions, such as the Netherlands, diagnosing epilepsy often involves both EEG and magnetic resonance imaging (MRI) to understand seizure risks [2, 3].

There are various kinds of brain wave patterns that can be seen in epilepsy such as seizures and Interictal Epileptic Discharges (IEDs) which are essential for diagnosing epilepsy [4]. These type of patterns can be observed not only during seizures but also in between areas that would indicate potential epileptic activity as IEDs [5]. Since epilepsy is a healthcare issue in the Netherlands, the need for diagnostic approaches is important. To mitigate these far-reaching implications, the implementation of a diagnostic system aims to identify high-risk epilepsy patients using techniques from signal processing and machine learning that can help reduce the occurrence of epilepsy. Furthermore, one of the main goals is to contribute on epilepsy diagnostics which will improve the Netherland's position in neurological studies.

The past few years researchers have been using machine learning and deep learning models to determine whether a patient has epilepsy or not. This research shows a detailed data-processing pipeline based on the methodologies shown in Thangavel et al.'s research [6]. The pipeline includes four essential stages used in many literature: data acquisition [7], pre-processing [8], feature extraction[4, 6], and classification as shown in Figure 1.1. Initially, EEG recordings are collected at the medical center / hospital and processed into a raw EEG data (EDF format). During the pre-processing phase, there are many techniques used such as artefact removal, sampling, and filtering for improving the data quality and removing noise in each channel. Next, feature extraction is used to identify different unique patterns from the processed EEG signals. Finally, these features are used in the classification stage to differentiate between epileptic and healthy patients. This systematic approach forms the foundation of this research methodology [6].

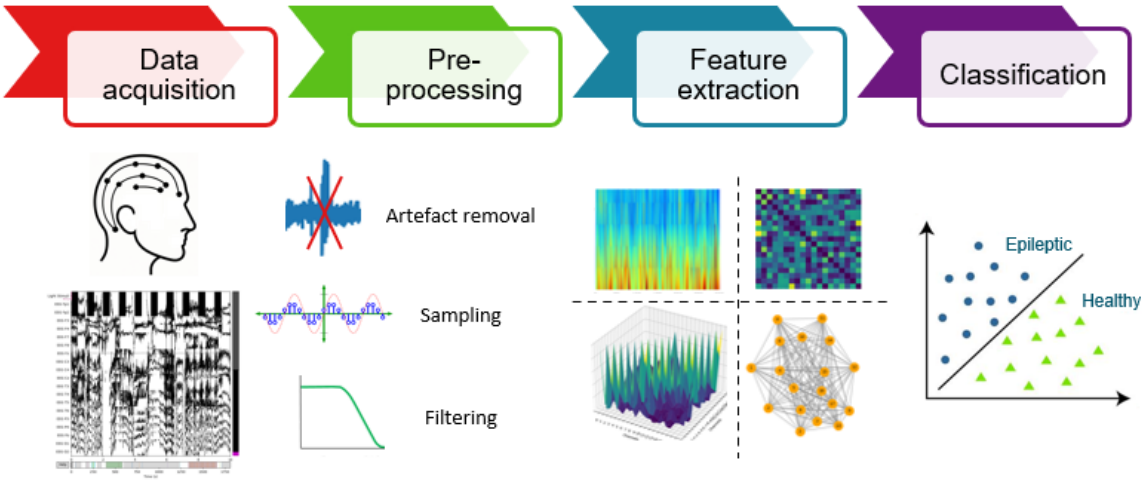


Figure 1.1: Data processing and analysis pipeline in EEG-based BCI systems.

1.2 The Main Challenges

Epilepsy diagnostics depend mainly on detecting interictal epileptiform discharges (IEDs) in EEG recordings. The presence of IEDs suggests a high risk of seizure recurrence up to 90% which often requires immediate concern [1]. On the other hand, a standard EEG without visible abnormalities usually leads to a follow-up sleep-deprived EEG to improve sensitivity [2]. However, even with these approaches, a significant number of patients develop epilepsy despite having initially normal EEG results highlighting a major diagnostic challenge that will be studied in this thesis.

The current problem in clinical neurology lies in the group of patients with visually normal EEGs who nonetheless face a significant risk of future seizures. The inability to identify these patients without IEDs early denies them the benefits of treatment which can potentially affect their quality of life and increase the burden on healthcare systems. This highlights a huge gap in the diagnostic process and underscores the need for enhanced detection methods.

The diagnosis process typically begins with an initial clinical evaluation where the neurologist gathers detailed medical history and information about the patient's seizures. This is followed by a neurological examination to assess brain function and identify any abnormalities. As mentioned before, the primary diagnostic tools used are the EEG and MRI. In some cases, additional tests such as blood tests, lumbar puncture, or neuropsychological evaluations may be conducted to gather more information [2].

The main goal of this thesis is to introduce an automated way to analyze EEGs that uncovers hidden indicators of epilepsy beyond the typical IEDs. This framework can be seen in Figure 1.1. The study seeks to address the gap by using computational

methods to identify subtle but medically important EEG patterns that can predict epilepsy in patients showing no apparent abnormalities on standard recordings.

1.3 Research Questions

This thesis investigates various methods for feature extraction, machine learning / deep learning classification, and model evaluation in the context of EEG signal analysis. This research aims to address the following questions:

- **RQ1: How effective are the traditional feature extraction methods in distinguishing between epileptic and healthy EEG signals for EEG signals w/o IEDs?**
- **RQ2: What is the impact of including additional patient-specific information (e.g. age, gender, vigilance state) on the classification accuracy of EEG signals?**
- **RQ3: How do different machine learning algorithms and hyperparameter tuning methods affect the classification performance of EEG signals?**
- **RQ4: What are the comparative performances of the proposed classification methods on different EEG datasets (e.g., TUH vs. EMC)?**
- **RQ5: How do deep learning architectures affect the performance metrics compared to traditional methods and how do these results align with those reported in recent research studies?**

These questions guide the research presented in this thesis aiming to improve the accuracy and reliability of EEG signal classification through advanced feature extraction, machine learning techniques, and different evaluation methodologies.

1.4 Thesis Outline

The thesis is structured to provide a comprehensive exploration of the research topic through several sections. It begins with an introduction to epilepsy diagnosis highlighting the main challenges outlining the problem statement and discussing the clinical significance of the research. Following this, the research questions guiding the study are presented along with an overview of the thesis outline.

The background chapter starts off with the current state of knowledge discussing montages, segment lengths, and combiners, as well as pre-processing techniques and EEG-level feature extraction methods. This chapter covers various features and provides an overview of machine learning and classification techniques relevant to the study.

The methods chapter describes the datasets used followed by an overview of the

pipeline showing feature selection, data analysis, and feature selection processes. This chapter also explains the model development, training, and evaluation processes. In stage one, the evaluation of individual features is explained while stage two focuses on the evaluation of combined features. In addition to this, the pre-trained model process is discussed with. The chapter also discusses hyperparameter tuning and model evaluation along with the evaluation metrics used in this thesis.

The statistical analysis chapter presents a detailed examination of the statistical methods used in the research including the Mann-Whitney U test and the subsequent findings. It also provides an overview of the VGG16 model features output analysis explaining significant features. The results chapter presents the findings from the feature extraction process and the performance of the models. It includes a detailed comparison of epileptic patients with interictal epileptiform discharges (IEDs) versus healthy controls using the TUH dataset and epileptic patients without IEDs versus healthy controls across both the TUH and EMC datasets. The chapter also discusses the results from the ensemble method and hyperparameter tuning.

Finally, the discussion chapter shows the results connecting them back to the research questions and the existing literature which was observed during the literature review phase. The future research directions and conclusion chapter summarizes the key findings of the study discusses their challenges for clinical practice and suggests potential areas for future research in the topic of epilepsy diagnosis.

2 Background

This chapter provides a brief background on the various techniques and methods used in this thesis. The current state of knowledge in the field of EEG-based epilepsy diagnosis is discussed followed by an exploration of specific methods such as montages, segment lengths, and combiners. Then, the pre-processing steps and the feature extraction techniques used to analyze EEG data is discussed in detail. Followed by this is a section discussing the machine learning and deep learning methods used. The processes done in MATLAB are provided by P. Thangavel, J. Thomas & J. Dauwels and it is replicated in Python.

2.1 Current State of Knowledge

For the past few years, diagnosing epilepsy involves examining EEG recordings to detect IEDs and other irregular patterns that can be used in clinical decision making [9, 10, 11]. Despite progress in the field of ML / AI, there are still obstacles in diagnosing epilepsy especially in patients with normal or inconclusive EEG results which then require long-term analysis [10]. There are limitations of approaches that prove the need for precise diagnostic techniques that can provide increased sensitivity and specificity.
Recent studies have focused on making automated systems to detect IEDs using signal processing and machine learning methods [12]. However, the effectiveness of these systems may vary depending on factors such as EEG data quality, feature selection and algorithm usage. This thesis seeks to fill some of these gaps through exploring techniques for feature extraction and classification to improve the precision of epilepsy diagnosis. Moreover, the articles studied during the literature phase were from different

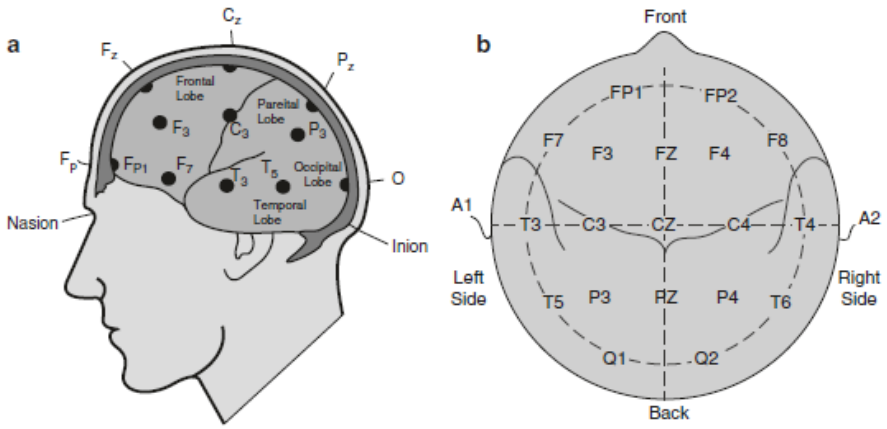


Figure 2.1: EEG figures show the locations of the major coordinates of the International 10–20 system with respect to the anatomy of the head and brain [13].

search engines as can be seen in Figure B.3.

2.2 Montages, Segment Lengths, & Combiners

In EEG analysis, montages refer to how electrodes are arranged / represented on the scalp [14]. There are different montages used in this thesis that can impact the interpretation of EEG signals. By evaluating the features for various non-overlapping EEG segment lengths from 2s to 300s, different outcomes are possible. For each segment length, various statistical combiners are calculated from multiple non-overlapping epoch segments of the entire EEG as shown in Table 2.1. In this study, as shown in

Table 2.1: Different montages, segment lengths, statistical combiners and feature sets investigated in this study similar to Thangavel et al. [6].

Montages	**Segment lengths [s]**	**Statistical combiners**
CAR Cz Bipolar Laplacian	2 5 10 20 30 60 300	Mean Median Standard deviation Skewness Kurtosis

Table 2.1, four different types of EEG montages are used i.e. Common Average Referential (CAR), Cz Referential, Longitudinal Bipolar, and Laplacian montage. In the CAR montage, the potential at each electrode is measured against the mean potential of all electrodes removing the influence of a physical reference electrode and the computation can be seen in Equation 2.1 [15].

$$V_{\mathrm{CAR},i} = V_{\mathrm{ER},i} - \frac{1}{n}\sum_{j=1}^{n} V_{\mathrm{ER},j}, \tag{2.1}$$

where $V_{\mathrm{ER},i}$ is the potential of the i-th electrode relative to the reference, and n is the total number of electrodes.

In the Cz montage, each electrode potential 'V' is referenced to a specific type of electrode i.e. in this case the 'Cz' electrode which is useful for analyzing non-localized EEG abnormalities and the computation can be seen in Equation 2.2 [6].

$$V_{\mathrm{Cz},i} = V_{\mathrm{ER},i} - V_{\mathrm{Cz}}, \tag{2.2}$$

where V_{Cz} represents the potential at the Cz electrode.

The Longitudinal Bipolar montage involves connecting each electrode to its immediate neighbor in a specific manner i.e. in 'an anterior-to-posterior direction which

then provides a good spatial resolution for detecting focal brain activity' as stated in Demoulin et al.'s paper [14, 16, 17]. The brain regions can then be deduced by the electrode pairs (19 channels) using Figure 2.1 and this can be seen in Table 2.2.

Table 2.2: Bipolar montage electrode pairs and their corresponding brain regions [18].

Electrode Pair	**Brain Region**
FP1 - F7	Frontal - Temporal
F7 - T3	Temporal
T3 - T5	Temporal - Parietal
T5 - O1	Parietal - Occipital
FP1 - F3	Frontal
F3 - C3	Frontal - Central
C3 - P3	Central - Parietal
P3 - O1	Parietal - Occipital
FP2 - F8	Frontal - Temporal
F8 - T4	Temporal
T4 - T6	Temporal - Parietal
T6 - O2	Parietal - Occipital
FZ - CZ	Frontal - Central
CZ - PZ	Central - Parietal
FP2 - F4	Frontal
F4 - C4	Frontal - Central
C4 - P4	Central - Parietal
P4 - O2	Parietal - Occipital

The Laplacian montage uses the average potential of the nearest surrounding electrodes as the reference for each electrode [19]. Reduction of local artifacts is the reason this montage is known and the computation can be seen in Equation 2.3 [15].

$$V_{\text{Laplacian},i} = V_{\text{ER},i} - \frac{1}{n} \sum_{j \in S_i} V_{\text{ER},j}, \tag{2.3}$$

where S_i is the set of neighbouring electrodes around the i-th electrode. With the statistical combiners found in Appendix B, the montages and segment lengths, this method can create multiple combinations that can be used to obtain a higher accuracy.

2.3 Pre-processing

Pre-processing of raw EEG data is an important step to improve the quality and reliability of the signals before feature extraction and classification. The following pre-processing pipeline was used for this study:

- **Noise Removal and Filtering:**
  The first step involves removing very high noise values i.e. ± 9999 which is an indication for artifacts during measurements [6]. This is followed by applying a notch filter (4th order, Butterworth) at 60 Hz to eliminate electrical artifacts [20].

A high-pass filter (4th order) is then used to remove the direct current (DC) offset and baseline variations from the signals [21]. These filters help in cleaning the data by removing both high-frequency noise and low-frequency drifts [22].

The filtering process is implemented using the `filtfilt` function which performs zero-phase filtering to avoid phase distortions. The padding length (`padlen`) is calculated as shown in Equation 2.4.

$$padlen = 3 \times (\max(\text{len}(b), \text{len}(a)) - 1) \tag{2.4}$$

where b and a are the filter coefficients [23]. The `padtype` used is 'odd' which pads the signal with odd symmetry.

- **Resampling:**
  To standardize this process further, the sampling rate of the EEG data is down-sampled to 200 Hz. This uniform sampling rate provides consistent feature extraction and analysis and helps with reducing the number of samples [24]. It can help especially if the original sampling rate is, for example, 1000 Hz as this can reduce the number of samples per EEG channel by 5 times.

- **Segmentation and Artifact Rejection:**
  The EEG data is segmented into 1-second intervals using a buffering technique. Each segment is then analyzed for artifacts by computing the root mean square (RMS) value. This buffering technique replicates the 'buffer' function in MATLAB and it can be seen in Algorithm 5.

  Segments with RMS values exceeding a certain threshold, determined based on noise statistics, are identified as noisy and are subsequently rejected [25]. This step ensures that the remaining data is free from significant artifacts such as muscle movements or environmental interferences[12].

- **Removing Empty Segments:**
  To further clean the data, segments that contain minimal EEG activity are identified and removed. This is achieved by analyzing the difference between adjacent channels and flagging segments with zero or near-zero values. Removing these empty segments helps in retaining only the relevant EEG data for analysis [26].

- **Conversion to Microvolts:**
  Finally, the EEG data is converted to microvolts, which is a standard unit for EEG measurements. This conversion ensures consistency in the amplitude values of the EEG signals across different recordings and datasets [27].

- **Removal of Light stimulation segments:**
  The segments of EEG data that contain light stimulation events were excluded to match the data similar to other datasets used for comparison. This is done by identifying the onset and duration of light stimulation events from an events file. Moreover, this step is important in eliminating potential artifacts created by the light stimulation that could mislead the classification results.

The preprocessing steps described above are crucial for minimizing the impact of various artifacts and improving the quality of the EEG data for subsequent analysis.
The pre-processing pipeline ensures that the EEG data is clean, segmented, and standardized, making it suitable for extracting meaningful features and performing accurate classification.

2.4 EEG-Level Feature Extraction

Feature extraction is an important step in the analysis of EEG data aimed at identifying important characteristics of the signals. This process includes extracting IED-independent features from various domains such as time, frequency, and time-frequency. Time-domain features include mean, median, and standard deviation, while frequency-domain features involve power spectral densities and band power ratios. Time-frequency analysis, such as wavelet transforms, provides a more detailed representation of the signal characteristics over time.

2.4.1 Univariate Temporal Measures (UTMs)

Univariate Temporal Measures (UTMs) are essential for analyzing the time-domain characteristics of EEG signals. These measures provide insights into the statistical properties of the signals which are important for identifying patterns associated with epilepsy. This section details the UTMs used in the pipeline explaining their importance and relevance to epilepsy diagnosis [6]. Consider $x(t)$ to be the EEG signal for each channel.

Mean: The mean μ of an EEG signal $x(t)$ over a time window is the average value of the signal.

$$\mu = \frac{1}{N}\sum_{i=1}^{N} x(i) \tag{2.5}$$

where N is the number of samples in the time window. The mean provides a baseline level of the signal [28].

Median: The median is the value separating the higher half from the lower half of the signal values. Unlike the mean, the median is robust to outliers and provides a better measure for skewed distributions.

Standard Deviation (std): The standard deviation measures the amount of variation of the signal.

$$\text{Standard Deviation} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x(i)-\mu)^2} \tag{2.6}$$

where μ is the mean of the signal. A high standard deviation means a lot of variability in the signal.

Kurtosis: Kurtosis measures how often outliers occur in the signal distribution.

$$\text{Kurtosis} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{x(i)-\mu}{\sigma}\right)^4 - 3 \tag{2.7}$$

where σ is the standard deviation of the signal. High kurtosis values indicate the presence of sharp peaks in the signal [29]. In Python, the difference of 3 is deducted from the measure to match the output of the kurtosis function in MATLAB.

Skewness: Skewness measures the asymmetry of the signal distribution which is important to see the deviations between each feature and the formula can be seen in Equation 2.8.

$$\text{Skewness} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{x(i)-\mu}{\sigma}\right)^3 \tag{2.8}$$

Peak-to-Peak Amplitude (V_{pp}): The peak-to-peak amplitude is the difference between the maximum and minimum values of the EEG signal.

$$V_{pp} = \max(x) - \min(x) \tag{2.9}$$

Number of Zero Crossings: The number of zero crossings is the number of times the signal crosses the zero voltage line.

Number of Peaks: The number of peaks is the count of local maxima in the EEG signal. Peaks in the EEG signal can show events such as spikes or artefacts in epileptic patients.

Nonlinear Energy Operators:

- **Envelope-Derivative NLEO (ED):** Measures the non-linearity in the envelope of the signal where H is the frequency response [30].

$$\text{ED}(x) = x'(t)^2 + H[x'(t)]^2 \tag{2.10}$$

- **Teager–Kaiser NLEO (TE):** Estimates the instantaneous energy of the signal [30].

$$\text{TE}(x) = x'(t)^2 - x''(t) \tag{2.11}$$

Nonlinear energy operators provide information about the signal's instantaneous energy helpful for detecting the changes in EEG data [30].

Signal Energy:

- **Time Domain Energy (E_t):** Total energy of the signal in the time domain [31].

$$E_t = log(\frac{1}{N}\sum_{i=1}^{N} x(i)^2) \tag{2.12}$$

- **Frequency Domain Energy (E_f):** Total energy of the signal in the frequency domain. This can be computed using the Discrete Fourier Transform (DFT) [31].

$$E_f = log(\sum_{k=1}^{N} |X(k)|^2) \tag{2.13}$$

  where $X(k)$ is the DFT of the signal $x(t)$.

Shannon Entropy H(x): Shannon entropy measures the uncertainty or randomness in the signal [32] and Equation 2.14 shows the formula with how to calculate entropy of a signal x(i).

$$\mathrm{H(x)} = -\sum_{i=1}^{N} p(x(i)) \log p(x(i)) \tag{2.14}$$

where $p(x(i))$ is the probability distribution of the signal values.

The box plots can be seen in Figure A.16 which compares the healthy and epileptic patients in the TUH dataset. The combination of these features allows for a comprehensive understanding of the EEG signal's statistical properties which are important for distinguishing between normal and abnormal brain activities.

2.4.2 Spectral

We evaluate spectral features, specifically, relative power (RP_f) obtained from five EEG frequency bands: delta (δ, 1–4 Hz), theta (θ, 4–8 Hz), alpha (α, 8–13 Hz), beta (β, 13–30 Hz), and gamma (γ, 30 Hz) [6]. Relative power (RP_f) is defined as shown in Equation 2.15 [6].

$$RP_f = \frac{P_f}{P_{total}} \tag{2.15}$$

where total power (P_{total}) is the sum of the powers of all frequency bands:

$$P_{\text{total}} = P_\delta + P_\theta + P_\alpha + P_\beta + P_\gamma$$

and f indicates different frequency bands ($f \in \{\delta, \theta, \alpha, \beta, \gamma\}$). This calculation results in five feature values for each single-channel EEG segment.

2.4.3 Wavelet Features

Wavelet transform is a method that shows the time-frequency representation of EEG signals and can be used for feature extraction of signals [33]. It captures both the frequency and temporal content. In this study, the wavelet coefficients are extracted using Discrete Wavelet Transform (DWT) and Continuous Wavelet Transform (CWT) [6].

Continuous Wavelet Transform (CWT): The CWT of a signal $x(t)$ is defined as shown in Equation 2.16 [34].

$$\mathrm{W}(a,b) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} x(t)\psi^* \left(\frac{t-b}{a}\right) dt \tag{2.16}$$

where a is the scale parameter, b is the translation parameter, $\psi(t)$ is the mother wavelet, and $\psi^*(t)$ denotes the complex conjugate of the mother wavelet. The type of mother wavelet used for the CWT is the Morlet (morl) because of its good balance between time and frequency resolution which is one of the main reasons most research papers use this [35]. When applied to a discrete signal $x[n]$, the CWT results in a matrix of coefficients:

$$\text{CWTmatrix} = \begin{bmatrix} W(a_1,b_1) & W(a_1,b_2) & \cdots & W(a_1,b_N) \\ W(a_2,b_1) & W(a_2,b_2) & \cdots & W(a_2,b_N) \\ \vdots & \vdots & \ddots & \vdots \\ W(a_M,b_1) & W(a_M,b_2) & \cdots & W(a_M,b_N) \end{bmatrix} \tag{2.17}$$

For this study, specific parameters are used to compute the CWT matrix based on the paper from Thangavel et al [6]. These parameters include the scale parameter s_0, the scale increment ds, and the number of scales $Nbsc$. The scales above frequency of 2 Hz are calculated as follows:

$$s_0 = 2.45, \quad ds = 0.4875$$

Note that the s_0 value in MATLAB is 0.2 whereas in Python 2.45 was the lowest value to be chosen due to difference in computation complexity. The number of scales $Nbsc$ and scales are computed as shown in Equation 2.18 and Equation 2.19 respectfully [36].

$$Nbsc = \left\lfloor \frac{\log_2(n_{samples} \cdot dt/s_0)}{ds} \right\rfloor \tag{2.18}$$

$$\text{scales} = s_0 \cdot 2^{(k \cdot ds)}, \quad k = 0, 1, \ldots, Nbsc - 1 \tag{2.19}$$

By using the FFT algorithm, the CWT computation captures the time-frequency characteristics of the EEG signals with high precision. This approach is essential for identifying and analyzing patterns associated with epileptic activity leading to more robust diagnostics. Besides this, the algorithm is faster than the convolution algorithm used in Python. This results in truncating the resulting CWT matrix to include the first 13 scales.

Discrete Wavelet Transform (DWT): The DWT decomposes a signal into a set of basis functions called wavelets obtained by shifting and scaling the mother wavelet function. It decomposes the signal into approximation (known as low pass sub-band) and detailed coefficients (high pass sub-band) [37]. Considering a signal x[n], each phase involves high pass and low pass filters i.e. g[n] and h[n], and double down samplings. From the approximation coefficients $A_j(k)$ at level j, the coefficients

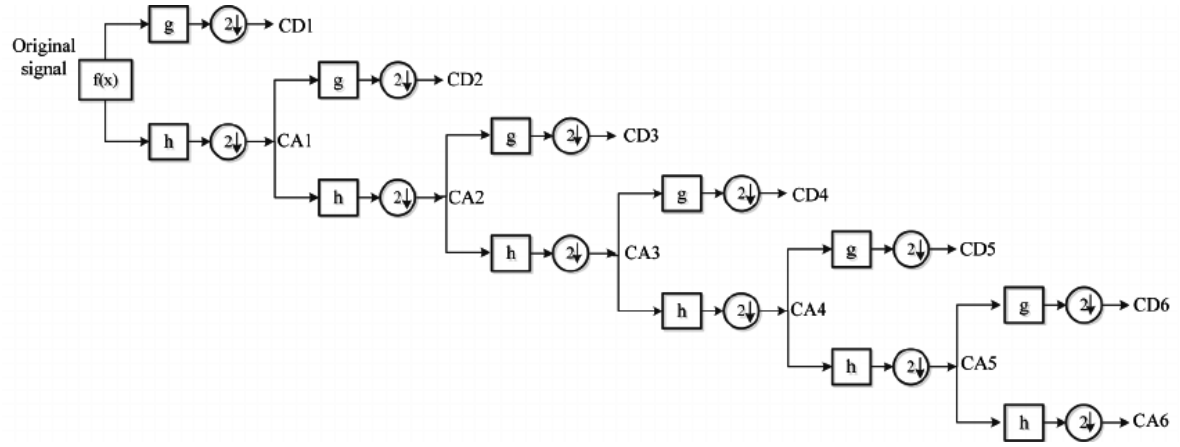


Figure 2.2: Structure of decomposition with DWT at level 6 [40].

$A_{j+1}(k)$ and $D_{j+1}(k)$ is computed as shown in Equation 2.20 [38].

$$\begin{aligned} D_{j+1}(k) &= \sum_{n \in z} g(2k-n) A_j(n) = c_{Dj+1} \\ A_{j+1}(k) &= \sum_{n \in z} h(2k-n) A_j(n) = c_{Aj+1} \end{aligned} \tag{2.20}$$

The type of mother wavelet used is the Daubechies (db4) i.e. 6 level wavelet decomposition [39].

$$\mathrm{DWT}(j,k) = \sum_{n=0}^{N-1} x[n] \psi_{j,k}[n] \tag{2.21}$$

where $\psi_{j,k}[n]$ is the mother wavelet function for the DWT.
The DWT results in a set of coefficients for each level of decomposition:

$$\mathrm{DWTmatrix} = \begin{bmatrix} c_{A1} : c_{A6} \\ c_{D1} \\ c_{D1} \\ \vdots \\ c_{Dj} \end{bmatrix} \tag{2.22}$$

where c_{Ai} are the approximation coefficients at level i and c_{Dj} are the detailed coefficients at level j. Note that MATLAB has pre-defined functions to obtain the approximation and detailed coefficients. In python, the Pywavelets package is used [41]. The psuedo code for this is shown in Algorithm 1.

Algorithm 1 Discrete Wavelet Transform (DWT) Process

Input: $data, Fs, montage, wavelet$
Output: c_all (combined coefficients)
$data \leftarrow$ apply_montage$(data, montage)$
for each channel j in $data$ **do**
    $signal \leftarrow data[j]$
    $coeffs \leftarrow$ pywt.wavedec$(signal, wavelet =' db4', \text{level} = 6)$
    for i from 1 to 6 **do**
        $cD[i] \leftarrow coeffs[i]$
    end for
    for i from 1 to 6 **do**
        $max\_level \leftarrow \text{len}(coeffs) - 1$
        if $i == max\_level$ **then**
            $cA[i] \leftarrow coeffs[0]$
        else
            $Aj \leftarrow$ pywt.waverec$(coeffs[: -i], wavelet =' db4')$
            if $\text{abs}(Aj[-1] - Aj[-2]) < 0.00001$ **then**
                $Aj \leftarrow Aj[: -1]$
            end if
            $cA[i] \leftarrow Aj$
        end if
    end for
    $c\_all \leftarrow [cD[i], cA[i]$ for i from 1 to 6$]$
end for
return c_all

Extracted Features:
From the wavelet coefficients, two features are extracted: the mean and the standard deviation of the square of the absolute values of the coefficients. These features capture the statistical properties of the wavelet coefficients providing valuable information about the EEG signal.
For a given set of wavelet coefficients $W(a_i, b_j)$, the detailed mathematical formulation is shown in Equation 2.23 [42].

$$\begin{aligned} \text{MSA} &= \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} |W(a_i, b_j)|^2 \\ \text{SSA} &= \sqrt{\frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} (|W(a_i, b_j)|^2 - \text{MSA})^2} \end{aligned} \tag{2.23}$$

The mean and standard deviation of the absolute values of the Continuous Wavelet Transform (CWT) and Discrete Wavelet Transform (DWT) coefficients are calculated as shown in Equation 2.24 [6].

$$\begin{aligned} \text{MSA}_{\text{morl}} &= \text{mean}(|\text{CWTmatrix}|) \\ \text{SSA}_{\text{morl}} &= \text{std}(|\text{CWTmatrix}|) \\ \text{MSA}_{\text{db4}} &= \text{mean}(|\text{DWTmatrix}|) \\ \text{SSA}_{\text{db4}} &= \text{std}(|\text{DWTmatrix}|) \end{aligned} \tag{2.24}$$

These simplified features, calculated as the mean and standard deviation of the absolute values of the wavelet coefficient matrices, offer a practical and computationally efficient means to capture essential characteristics of the EEG signals for further analysis. To illustrate the wavelet features, the average CWT coefficients across a subset of epileptic and healthy EEGs can be computed and visualized. The visualizations in Figure A.8 help to illustrate the differences in the time-frequency characteristics of EEG signals from epileptic and healthy patients. These differences are crucial for developing accurate diagnostic models and understanding the underlying brain activity. The color bar indicates the squared magnitude (power) of the CWT coefficients.

2.4.4 Connectivity Features

In this study, we evaluate the connectivity between the n channels of EEG to understand the interactions and communication between different regions of the brain. The connectivity features are derived by computing the n^2 connectivity matrix between the channels and then extracting the lower triangular matrix features. This results in $(n^2 - n)/2$ features which is exactly replicating the method proposed by Thangavel et al. [43]. We focus on two primary connectivity measures: maximum normalized cross-correlation (C-C) and phase locking value (C-PLV). These measures provide valuable insights into the temporal and phase relationships between EEG channels, which are crucial for understanding the underlying neural dynamics associated with epilepsy [43, 44].

Maximum Normalized Cross-Correlation (C-C)
The maximum normalized cross-correlation between two input signals x_n and y_n is a measure of the similarity between the signals as a function of the time-lag applied to one of them. It is computed using the following Equation 2.25 and 2.26 [43].

$$\hat{R}_{xy,\max}(m) = \frac{1}{\sqrt{\hat{R}_{xx}(0)\hat{R}_{yy}(0)}}\hat{R}_{xy}(m), \tag{2.25}$$

$$\hat{R}_{xy}(m) = \begin{cases} \sum_{n=0}^{N-m-1} x_{n+m} \times y_m^*, & \text{for } m \geq 0, \\ \hat{R}_{yx}^*(-m), & \text{for } m < 0, \end{cases} \tag{2.26}$$

where $\hat{R}_{xy}(m)$ is the cross-correlation function, $\hat{R}_{xx}(0)$ and $\hat{R}_{yy}(0)$ are the autocorrelations of x_n and y_n at lag 0, respectively, and * indicates complex conjugation. This measure captures the strength and direction of the relationship between two EEG signals over different time lags [43].

Phase Locking Value (C-PLV)
The phase locking value (C-PLV) quantifies the consistency of the phase difference between two signals across time. It is computed as [44]:

$$\mathrm{C-PLV} = \frac{1}{N}\left|\sum_{n=1}^{N} \exp\left(\mathrm{i}\left[\psi_x(n) - \psi_y(n)\right]\right)\right|, \tag{2.27}$$

where $\psi_x(n)$ and $\psi_y(n)$ are the instantaneous phase values of the signals x_n and y_n at time n, computed using the Hilbert transform, and i denotes the imaginary unit. The C-PLV ranges from 0 to 1, with 1 indicating perfect phase synchronization and 0 indicating no phase synchronization [44].
Visualisation of the connectivity matrices comparing a healthy and epileptic patient are shown in Figure A.9. By using these connectivity measures, we can gain insights into the functional connectivity patterns in the brain, which are critical for understanding the neural mechanisms underlying epilepsy. These features are extracted and used in subsequent analysis and classification tasks to improve the accuracy and robustness of epilepsy diagnosis.

2.4.5 Stockwell transform

The Stockwell Transform (ST), also known as the S-transform, is a hybrid of the Short-Time Fourier Transform (STFT) and the Continuous Wavelet Transform (CWT) [45]. It combines the frequency localization properties of the STFT with the multiresolution capabilities of the CWT making it highly suitable for analyzing non-stationary signals such as EEG.
The Stockwell Transform is defined as a CWT with a specific mother wavelet $w(t, f)$ multiplied by a phase factor [45]. The ST of a signal $x(t)$ is shown in Equation 2.28.

$$S(\tau, f) = \exp(i2\pi f\tau)W(\tau, d) \tag{2.28}$$

where the CWT of an input function $x(t)$ is defined as shown in Equation 2.29 [46].

$$W(\tau, d) = \int_{-\infty}^{\infty} x(t)w(t-\tau, d)\,dt \tag{2.29}$$

and the specific mother wavelet is defined as shown in Equation 2.30 [6].

$$w(t, f) = \frac{f}{\sqrt{2\pi}} \exp\left(-\frac{t^2 f^2}{2}\right) \exp(-i2\pi ft)\,dt \tag{2.30}$$

Here, the scale parameter d is the inverse of frequency f. In MATLAB, the edge removal process is often handled automatically, but in Python, we need to apply edge removal manually before performing the Hilbert transform and then applying the Stockwell Transform. The edge removal process is shown in the psuedo-code in Algorithm 2. The following steps are crucial for ensuring accurate ST computation:

- Edge Removal: This step removes polynomial trends from the data and applies a window function to smooth the edges of the timeseries data [47]. To mitigate edge

Algorithm 2 Edge Removal Process

```
function APPLY_EDGE_REMOVAL(timeseries)
    n ← length of timeseries
    ind ← range from 0 to n
    r ← POLYNOMIAL_FIT(ind, timeseries, 2)
    fit ← POLYNOMIAL_EVALUATE(r, ind)
    timeseries ← timeseries - fit
    sh_len ← ⌊n/10⌋
    wn ← HANNING_WINDOW(sh_len)
    if sh_len == 0 then
        sh_len ← n
        wn ← array of ones with length sh_len
    end if
    timeseries[:sh_len//2] ← timeseries[:sh_len//2] × wn[: sh_len//2]
    timeseries[-sh_len//2:] ← timeseries[-sh_len//2:] × wn[−sh_len//2]
    return timeseries
end function
```

effects, a second-degree polynomial fit was removed from the time series data. The time series $X(t)$ is detrended by fitting a polynomial $P(t)$ and subtracting it:

$$P(t) = at^2 + bt + c \tag{2.31}$$

where a, b, and c are coefficients determined by polynomial fitting. The detrended signal is then given by:

$$X_{\text{detrended}}(t) = X(t) - P(t) \tag{2.32}$$

In addition to this, a Hann window was applied to the edges of the signal for smoothing.

- Hilbert Transform: This step is used to compute the analytic signal which provides instantaneous amplitude and phase information [48]. The Hilbert transform is applied to the edge-removed time-series data.
- Stockwell Transform: Finally, the Stockwell Transform is applied to the preprocessed signal to obtain the time-frequency representation using the stockwell package [49].

Extracted Features From the Stockwell Transform, two features are extracted that are particularly relevant for analyzing EEG signals:

- **Mean Square Root of Standard Deviations (ST-SR)**: This feature captures the mean square root of the standard deviations of the ST matrix, providing a measure of the variability in the signal across different frequencies and time epochs.

$$\text{ST-SR} = \text{mean}\left(\sqrt{\text{std}(\text{STmatrix})}\right) \tag{2.33}$$

- **Skewness of Sum of Powers (ST-P)**: This feature measures the skewness of the sum of the powers of the Stockwell Transform over epochs of $\frac{F_s}{2}$, giving insights into the asymmetry of the power distribution in the signal [50].

$$\text{ST-P} = \text{skewness}\left(\sum |\text{STmatrix}|\right) \tag{2.34}$$

In Python, edge removal and Hilbert transform are performed before computing the Stockwell Transform to ensure accuracy. Moreover, the features above capture the statistical properties of the Stockwell Transform coefficients providing valuable information about the underlying EEG signal. The ST-SR feature helps in understanding the signal's variability while the ST-P feature highlights the asymmetry in the power distribution which are both important for distinguishing between normal and abnormal brain activities.

2.4.6 Graph metrics

Graph metrics play a crucial role in understanding the connectivity and structural properties of EEG networks. Derived from the Cross-Correlation (C-C) and Phase Locking Value (PLV) feature sets, these metrics provide insights into the topological characteristics of brain networks. This section details the graph metrics used in this study, computed using the MATLAB Brain Connectivity Toolbox [51]. From the C-C and C-PLV feature sets, two types of networks are constructed:

- **C-C Network (C-C-net)**: Based on Cross-Correlation features [6].
- **C-PLV Network (C-PLV-net)**: Based on Phase Locking Value features [6].

Nodal features provide information about individual nodes (channels) within the network. The following nodal features are calculated (see Table 2.3). Note that these descriptions were taken from the BCT toolbox and a few research papers [52, 53, 54]. Edge features describe the properties of connections between nodes. The following edge features are computed (see Table 2.4). An aggregate feature provides a summary measure of the network. In this case, there is only 1 feature i.e. matching index that provides a measure of the similarity of neighbors between pairs of nodes [53]. In addition to this, the graph metric visualisations are shown in Figure A.11. Moreover, some of the features didn't work the same as in MATLAB so a few of the algorithms had to be re-written due to conflicts in the BCT Python toolbox. Some of the algorithms are re-written and implemented to match the MATLAB code and this can be seen in Appendix D.

Table 2.3: Nodal Features in EEG Network Analysis with descriptions taken from BCT Toolbox [53, 52].

Feature Name	**Description**
Degree	The number of connections a node has.
Strength	The sum of weights of connections a node has.
Assortativity	The tendency of nodes to connect to others that are similar.
Characteristic Path Length	The average shortest path length between nodes.
Local Efficiency	Efficiency of information transfer within a node's neighborhood.
Eccentricity	Most distance b/w 2 nodes.
Betweenness Centrality	Count of shortest paths through each node.
Eigenvector Centrality	Strength of a node in a network.
Clustering Coefficient	The degree to which nodes cluster together.
Node Coreness	The level of connectivity of a node within the core of the network.
Participation Coefficient	The extent of a node's connections within different communities.
Diversity Coefficient	The diversity of a node's connections across different communities.

Table 2.4: Edge Features in EEG Network Analysis with descriptions taken from BCT Toolbox [52].

Feature Name	**Description**
Assortativity Coefficient	The correlation between the degrees of connected nodes.
Global Efficiency	The efficiency of information transfer across the entire network.
Radius	The minimum eccentricity of any node in the network. [54]
Diameter	The maximum eccentricity of any node in the network. [54]
Transitivity	The ratio of triangles to triplets in the network.
Edge Neighborhood Overlap	The overlap of group of nodes between connected nodes.
Node Pair Degree	The product of degrees of connected nodes.

2.4.7 Short-Time Fourier Transform

The Short-Time Fourier Transform (STFT) is used in time-frequency analysis to capture both frequency and temporal information and is effective for examining EEG signals that vary over time [55]. By dividing the signal into short segments of equal length and then applying the Fourier transform to each segment, the STFT can be computed. A Hamming window is used as the window function (where $'nperseg' = 64$ in this case). Here, larger window size provides better frequency resolution [55]. This is important to detect patterns to distinguish the 2 classes. Mathematically, the STFT of a signal x(t) is defined as shown in Equation 2.35 [55].

$$\text{STFT}(t, f) = \int_{-\infty}^{\infty} x(\tau) w(\tau - t) e^{-j2\pi f \tau} \, d\tau \tag{2.35}$$

The output of the STFT is a complex-valued function of time and frequency as shown in the above formula and its magnitude squared is known as the scalogram / spectrogram which can be seen in Equation 2.36 which represents the energy distribution of the signal [55].

$$E(t, f) = |\text{STFT}(t, f)|^2 \tag{2.36}$$

The STFT is used to generate time-frequency representations of the EEG signals which can then be input for a deep learning / transfer learning application to classify the healthy vs epileptic patients. Algorithm 3 processes epilepsy / non-epilepsy signals by

Algorithm 3 STFT Image Generation with Sobel Filtering

Input: Set of EEG signals `EEG_signals`, output directory `dir1`, STFT parameters `nperseg`, `noverlap`, `nfft`
Output: Sobel-filtered STFT images
Initialize `batch_size` to n EEGs
for each batch in `EEG_signals` with size `batch_size` **do**
    for each signal in batch **do**
        Compute STFT: $(f, t, Zxx) \leftarrow$ `stft(signal, fs, nperseg, noverlap, nfft)`
        Compute magnitude scalogram: `scalogram` $\leftarrow$ `|Zxx|`
        Apply Sobel filter along frequency and time axes:
          `sobel_x` $\leftarrow$ `sobel(scalogram, axis` $= 0$`)`
          `sobel_y` $\leftarrow$ `sobel(scalogram, axis` $= 1$`)`
        Compute gradient magnitude: `sobel_scalogram` $\leftarrow \sqrt{\texttt{sobel\_x}^2 + \texttt{sobel\_y}^2}$
        Save `sobel_scalogram` as STFT output image in `dir1`
        Free memory by deleting variables and calling garbage collection
    end for
end for

computing their Short-Time Fourier Transform (STFT) to generate scalograms using Sobel filtering to also detect edges. The resulting Sobel-filtered STFT images are saved to a directory with memory management steps to handle large datasets / EEG signals with too many sample points. Note that with multiple EEGs per patient, the EEGs for every channel are concatenated and then the STFT is applied to the flattened EEG signal.

2.5 Machine learning, Transfer learning & XAI

2.5.1 Machine learning (ML)

ML is a subset of artificial intelligence that focuses on creating algorithms to enable computers to learn from and make predictions based on data [56]. There is supervised and unsupervised learning which can be seen in Figure 2.3. One of the main types of ML is supervised learning which involves training a model on a labeled dataset which will be done in this thesis. In supervised learning, each training example has an input and a corresponding output label where the main aim of the model is to learn a mapping from inputs to outputs for accurate predictions on new data. Supervised learning can

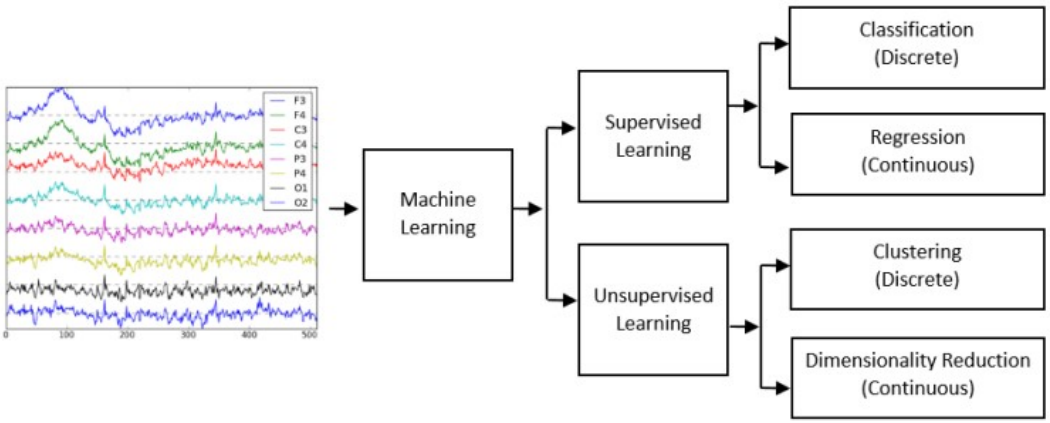


Figure 2.3: Machine learning applications on EEG [57].

be divided into regression and classification [57]. Here, we focus on the classification part in this thesis. Classification tasks predict discrete labels from predefined classes [58].
The dataset is usually split into a training, testing and validation set for seeing the performance of the classifier. Cross-validation is mainly used to assess model performance. This technique involves dividing the dataset into multiple folds, training, and testing the model multiple times with different folds to ensure reliability [59]. Moreover, hyperparameter tuning is important for optimizing model performance. Techniques like grid search and random search test multiple hyperparameter combinations to find the best configuration.
In classification tasks, performance is evaluated using metrics like accuracy, precision, recall, F1-score, and AUC-ROC. Accuracy is the ratio of correctly predicted instances, while precision and recall provide insights into the model's handling of true positives. The F1-score balances precision and recall, and AUC-ROC assesses the model's ability to distinguish between classes at different thresholds. Some of the classifiers that are used are SVM, XGBoost, and Random Forests.

2.5.2 Deep Learning

In the realm of machine learning, transfer learning is a technique that uses pre-trained models on datasets to apply knowledge to related but different tasks [60]. This approach is especially beneficial in domains with limited data as it enables the reuse of features learned from extensive datasets such as images, audio, and text [10, 61].
The architecture of CNNs typically comprises layers pooling layers and connected layers where after the pooling layers, there are feature maps that can be used in further classification processes. CNNs excel in tasks like image classification due to their capability to autonomously learn and extract features from data [62, 63]. The VGG16 Model is a trained deep learning model known for its success in image classification With a composition of 16 layers containing 13 layers and 3 connected layers VGG16 has been trained on extensive datasets such as ImageNet to effectively extract meaningful features from images [64].

2.5.3 Explainable AI (XAI)

The XAI method used in this thesis is SHAP (SHapley Additive exPlanations) values. The main use of this is to check and understand the feature importance in machine learning models. SHAP values, derived from cooperative game theory, provide a unified approach to explain the output of machine learning models by attributing the contribution of each feature to the model's prediction [65, 66]. Moreover, SHAP values provide insights into which features in the feature vector can have the most influence on the model's predictions. This is important in understanding complex models to check their interpretability. Besides this, SHAP values help to identify key EEG signal features that differentiate between healthy and epileptic states [67, 68].

SHAP values, also known as Shapley values, are formulated to distribute payouts fairly among players in a cooperative game [65]. For a given model f, the Shapley value for a particular feature i is computed using Equation 2.37 [67].

$$\phi_i = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!} \left[f(S \cup \{i\}) - f(S) \right] \tag{2.37}$$

In this formula, N represents the set of all features, S is a subset of N excluding i, and $|S|$ denotes the number of elements in S [66, 67]. To determine the contribution of each feature to the model's prediction, SHAP values are calculated by averaging the marginal contributions of a feature across all subset i.e. in this case the number of EEGs / patients S. This is mathematically represented as shown in Equation 2.38 [67].

$$\phi_i = \mathbb{E}_{S \subseteq N \setminus \{i\}} \left[f(S \cup \{i\}) - f(S) \right] \tag{2.38}$$

In practice, this expectation can be efficiently approximated using algorithms like Kernel SHAP and Tree SHAP mainly for models based on decision trees [66, 69]. For the implementation, the SHAP Python library is used [70]. The process involves steps like training the ML model, computing SHAP values for each feature in the feature vector, and then taking the mean of all these values to check overall feature importance across all patients / EEGs. Moreover, the generalized algorithm in terms of code to

Algorithm 4 Generalized SHAP Value Computation Process

```
Input: Feature Extraction Matrix, Model
Output: Top SHAP Values, Features index
for each Feature Matrix X in Dataset do
    Train Model M on X
    Compute SHAP values S using M
    Identify the top k features with the highest absolute SHAP values
    Store the top k SHAP values
end for
Combine and Analyze SHAP Results across all Feature Matrices
Visualize the top SHAP values with respect to their feature importance
```

compute the SHAP values is based on the feature matrix input and the model which in this case was the XGBoost due to its high performance. The process to get the SHAP values results is shown in Algorithm 4.

3 Method

This section explains the data collection process, the types of feature extraction methods selected, and the approach for classifying the patient data. In addition to this, the statistical tools used to compare MATLAB and Python implementations are introduced and later on compared. The metrics used in this study to analyze how the models work.

3.1 Datasets

For this thesis, routine scalp EEG recordings from two independent datasets are used: Temple University Hospital (TUH, USA) which is the largest publicly available EEG corpus [71] and Erasmus Medical Center Rotterdam (EMC, NL). The statistics of the data-set is shown in Table 3.1. For the TUH dataset, the information about the first seizure was not available in all the clinical reports [6]. The EEGs were recorded at various sampling frequencies according to the International 10–20 electrode placement scheme and they were annotated by at the individual centers independently in typical clinical settings [6]. For the EMC dataset, the diagnosis process is shown in Figure 3.1.

Table 3.1: Patient details of both datasets [Note: For the Age / Gender, (Mean ± Standard deviation) Age]

Dataset	**Type**	**Fs (Hz)**	**No of EEGs (No of patients)**	**Age/Gender**	
				Male	**Female**
TUH w IEDs	Normal	500	44 (30)	60 (48.8 ± 17.9)	39 (50.9 ± 20.5)
	Epileptic		259 (42)	32 (50.3 ± 20.2)	33 (56.4 ± 19.7)
TUH wo IEDs	Normal	500	44 (30)	55 (46.7 ± 18.2)	35 (49.5 ± 19.8)
	Epileptic		161 (33)	20 (34.5 ± 15.3)	24 (37.9 ± 16.1)
EMC	Normal	1000	105 (105)	69 (40.00 ± 17.85)	36 (44.78 ± 19.56)
	Epileptic		42 (42)	28 (49.75 ± 17.10)	14 (40.00 ± 17.85)

The procedure starts off with an initial clinical evaluation where the neurologist gathers detailed medical history and information about the patient's seizures. As mentioned before, EEG is used to record the electrical activity of the brain and detect abnormal patterns indicating if a patient could have epilepsy [72]. Whereas, the MRI provides detailed images of the brain's structure to identify any underlying conditions that might cause seizures [73]. In some special cases, additional tests are required to collect more information. The diagnosis may also involve long-term monitoring with video EEG to capture and analyze seizure activity over an extended period. The annotations are shown in Figure A.12 for the EMC dataset. R. van den Berg et al. explained the

overall procedure and the outcomes flowchart illustrates these steps showing the progression from initial evaluation to various diagnostic tests and procedures. It highlights the importance of a thorough and systematic approach to ensure accurate diagnosis and effective management of epilepsy. This detailed diagnostic process is essential for

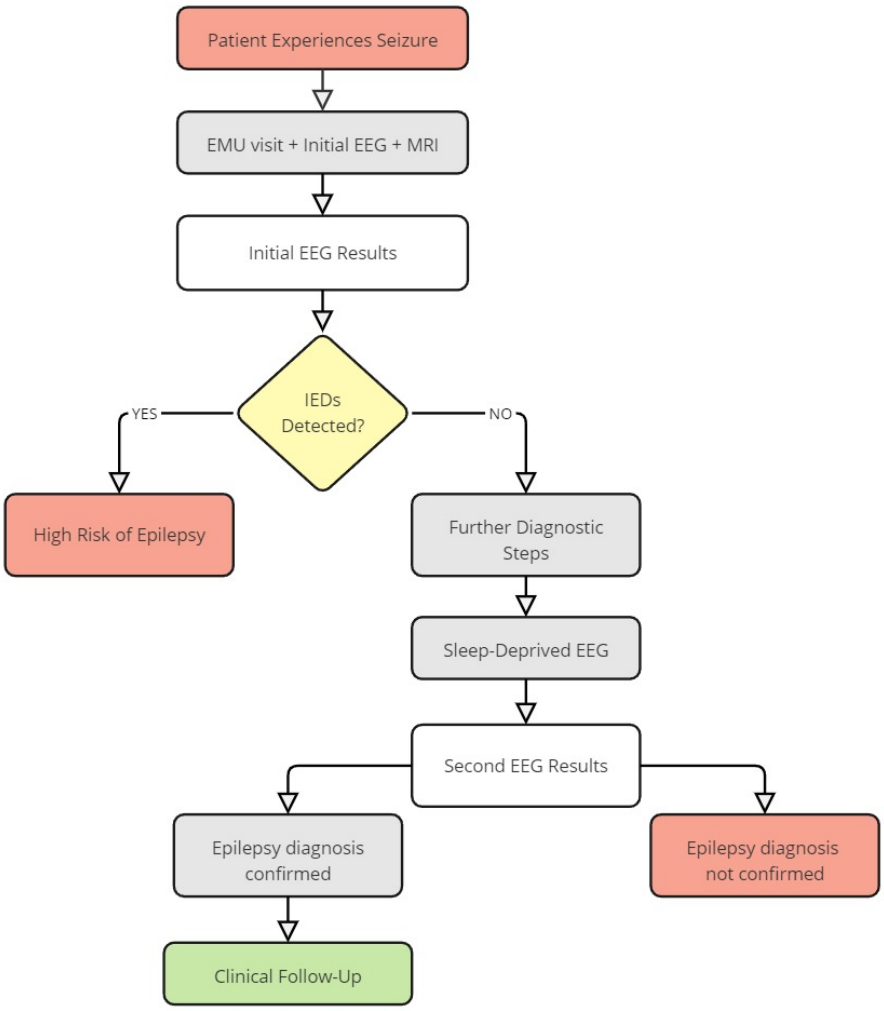


Figure 3.1: A comprehensive flowchart of the epilepsy diagnosis process of the EMC dataset provided based on feedback by R. van den Berg.

identifying patients at an elevated risk of epilepsy, allowing for timely interventions and personalized treatment plans. By following this structured approach, healthcare providers can boost the patient outcomes.

3.2 Overview of pipeline used for validation

3.2.1 Data analysis

The following pre-processing pipeline is used for both the TUH and EMC datasets as shown in Figure 3.2. These steps explained in the chapter 2 are used to pre-process and clean the EEG data.

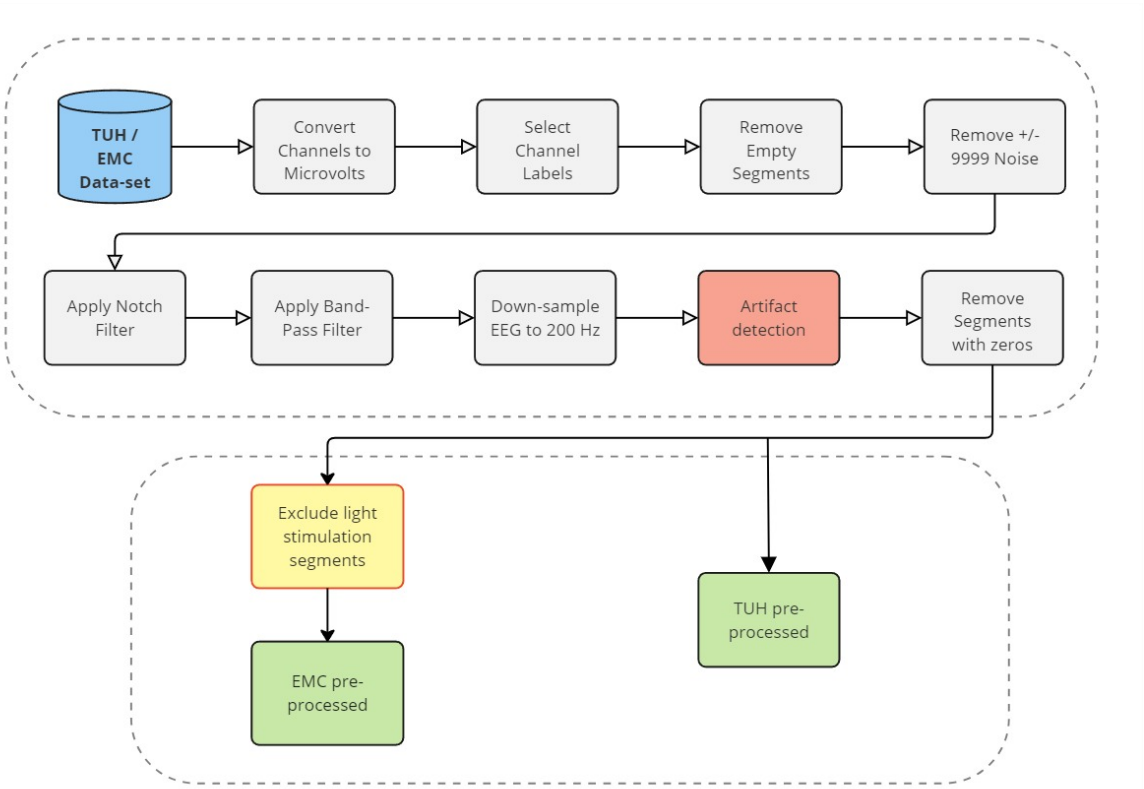


Figure 3.2: Pipeline used for pre-processing.

3.2.2 Feature selection

This section details the construction of feature vectors for each patient based on the extracted features. Each feature set is organized to create a comprehensive representation of the EEG data which can be used for further analysis and modeling. Spectral features are derived from the power spectral density (PSD) estimates of the EEG signals across the Given 19 EEG channels, the feature vector containing all the frequency bands (delta, theta, alpha, beta, and gamma) for each patient has a shape:

$$\text{S-F} = 19\,\text{channels} \times 5\,\text{bands} = 1 \times 95$$

UTMs capture the time-domain characteristics of the EEG signals. For each EEG channel, the mean, median, standard deviation, kurtosis, skewness, peak-to-peak amplitude,

number of zero crossings, number of peaks, nonlinear energy operators (NLEOED and NLEOTK), signal energy (time and frequency domain), and Shannon entropy are computed. This results in:

$$\text{UTM-F} = 19\,\text{channels} \times 13\,\text{features} = 1 \times 247$$

Using the DWT and CWT, the wavelet features are computed. For each channel,

Table 3.2: Feature Vectors for Each Patient used in XGBoost + LOSO CV classification

Feature Type	***Individual Features***	**Calculation**	**Size/Shape**
S-F	*Power in delta, theta, alpha, beta, and gamma bands for each ch*	19 ch × 5 bands	1 × 95
UTM-F	*Mean, median, std, kurtosis, skewness, peak-to-peak amplitude, zero crossings, peaks, NLEOED, NLEOTK, time and frequency domain energy, Shannon entropy*	19 ch × 13 features	1 × 247
DWT-F	*Mean and std of DWT coefficients at 6 levels*	19 ch × 6 DWT levels × 2 features	1 × 456
CWT-F	*Mean and std of CWT coefficients at 6 scales*	19 ch × 6 CWT scales × 2 features	1 × 494
CC-F	*Cross-correlation (C-C) between ch*	19 ch × 19 ch	1 × 171
CPLV-F	*Phase locking value (PLV) between ch*	171 × 6	1 × 1026
Graph Metrics	*Nodal and edge features from C-C and C-PLV networks*	Mean of all ch values	1 × 20 each
ST Features	*Mean square root of std (ST-SR), skewness of sum of powers (ST-P)*	19 ch × (5 bands + eeg signal) × 2 features	1 × 570 each

the mean and standard deviation of the square of the absolute values of the DWT and CWT coefficients are extracted. Given the multi-level decomposition in DWT and multi-scale analysis in CWT, the feature vectors are constructed as:

$$\text{DWT-F} = 19\,\text{channels} \times 12\,\text{DWT coefficients} \times 2\,\text{features} = 1 \times 456$$

$$\text{CWT-F} = 19\,\text{channels} \times 13\,\text{CWT scales} \times 2\,\text{features} = 1 \times 494$$

Connectivity features are derived from the cross-correlation (C-C) and phase locking value (PLV) between the EEG channels. The feature vectors are constructed as follows:

$$\text{C-C F} = 19\,\text{channels} \times 19\,\text{channels} = 1 \times 171$$

$$\text{C-PLV F} = 171 \times 6 = 1 \times 1026$$

Graph metrics are computed based on the connectivity features. For each EEG recording, the graph metrics include nodal features (e.g., degree, strength, centrality measures) and edge features (e.g., assortativity, efficiency). For each connectivity network (C-C and C-PLV), the mean of all channel values is taken, resulting in:

$$\text{C-C Network F} = 1 \times 20$$

$$\text{C-PLV Network F} = 1 \times 20$$

The ST features include the mean square root of standard deviations (ST-SR) and the skewness of the sum of powers (ST-P) of the ST matrix over different epochs. For each EEG channel and band, these features are computed as:

$$\text{ST Feature Vector} = 19\,\text{channels} \times (5\,\text{bands} + \text{eeg signal}) \times 2\,\text{features} = 1 \times 570$$

This structured approach ensures that all relevant characteristics of the EEG signals are captured providing a robust input for further analysis and machine learning models.
Besides this, there are a few clinical features that can be added to the data-set by concatenating these features with the feature vector mentioned above. This is shown in Table 3.3 which displays the one-hot encoding technique. One-hot encoding is a technique used in machine learning to convert categorical data into a format that can be provided to algorithms to improve predictions [74]. The categorical data is converted into binary vectors where each category is represented by a unique combination of binary values. This is particularly useful for categorical features such as age ranges, vigilance states, and gender as it allows the model to interpret the data without assuming any relationship between the categories. In the table, we have used binary encoding for age ranges and vigilance states [6] where each category is represented by a two-digit binary code. For gender, a simple binary encoding (0 for Female, 1 for Male) is used.

Table 3.3: Encoding for Age, Vigilance State, and Gender of the Patient

Feature	**Encoding**	**Feature Value**
Age	00	$18 \leq \text{age} < 30$
	01	$30 \leq \text{age} < 50$
	10	$50 \leq \text{age} < 70$
	11	$\text{age} > 70$
Vigilance State of Patient	00	Awake
	10	Drowsy
	11	Intermittent sleep
Gender	0	Female
	1	Male

3.2.3 Multiprocessing & CLI

Due to the high computation times for some features, multiprocessing is added to the pipeline. As shown in Figure B.2, the multiprocessing process for each montage,

segment length, and combiner is done. This improved the speedup to be approximately 10 times faster than with just serial processing besides parallel processing. For instance, the CWT computation time for the EMC dataset was approximately 55 hours with 1 core but with 16 cores, it reduces to 5-6 hours of computation time. Also, command line interface (CLI) of each function / method is done for the neurologist to have an easier time to run the code with integration of the tqdm tool to see the progress bar (%) of how many files have been created.

3.3 Model Development & Training

This section details the methodologies used for developing, training, and evaluating the models in this study. The process includes establishing a benchmark using a CNN architecture, performing Leave-One-Subject-Out Cross-Validation (LOSO CV) on various feature sets, and evaluating the models using different feature combinations.
To establish a benchmark, the same model selection pipeline is used as described in the recent paper [6]. The LOSO CV EEG classification (epileptic EEGs with IEDs vs. normal EEGs) was applied separately to each EEG dataset in this paper i.e. the TUH and EMC datasets. EEG classification was conducted across the two datasets in two stages:

3.3.1 Stage One: Evaluation of Individual Features

In the first stage, LOSO CV was performed on each feature set individually for each EEG dataset as shown in Figure 3.3. The process for each dataset and selected feature involved training the classifier on $N-1$ subjects and evaluating it on the Nth subject. This was repeated N times to evaluate all subjects.
For each dataset, a Receiver Operating Characteristics (ROC) curve was generated from the test results, and the LOSO CV Area Under the Curve (AUC) and Balanced Accuracy (BAC) were computed with other metrics explained later. This process was repeated across different datasets and the mean AUC and BAC results were saved. Given the various combinations of montages, segment lengths, and statistical measures, 144 LOSO CV EEG classification results were obtained for each feature set. Based on these results, the features were ranked according to their maximum LOSO CV AUC.

3.3.2 Stage Two: Evaluation of Combined Features

In the second stage, LOSO CV was performed by combining the IED rate with various combinations of IED-independent features to check the validation of the code made in Python. This method is shown in Figure 3.4. For each dataset with N subjects, data was split into training, validation, and testing sets. The data from the Nth subject is used for testing while the remaining $N-1$ subjects were split equally for training and validation.
A grid search was made to find the optimal combination of weights for the multiple features so that the sum of the weights equaled one. The best combination of weights was then applied to the test subject's data to predict the output. This step was

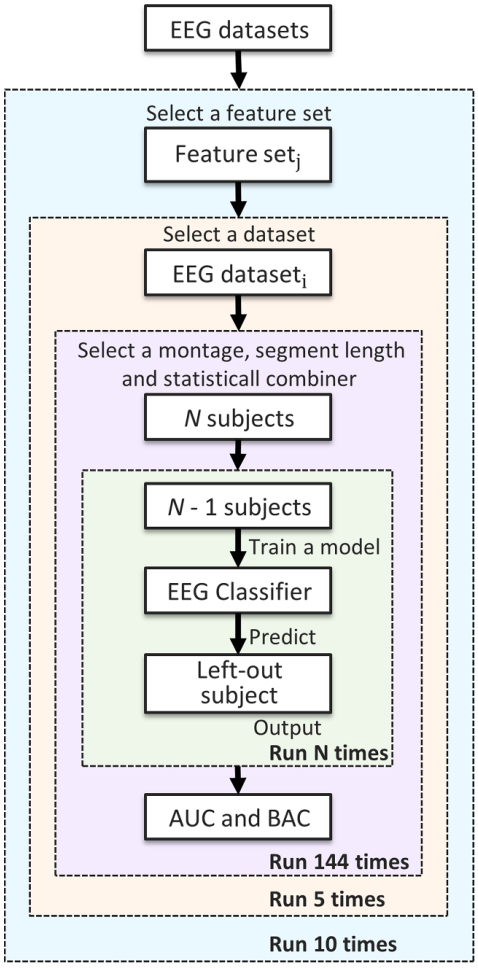


Figure 3.3: LOSO CV process on individual features [6].

repeated N times to evaluate all subjects, and the AUC and BAC were computed from the ROC curve. For the evaluation, the optimal weights were used to combine the predictions from the classifiers trained on different feature sets and the equation is shown in Equation 3.1 [6].

$$\text{Output} = \sum_{i=1}^{n} w_{\text{opt},i} \times o_i \tag{3.1}$$

where $w_{\text{opt},i}$ is the optimal weight for the output o_i from the i-th classifier, and n represents the total number of feature sets.

3.3.3 Hyperparameter Tuning and Model Evaluation

To make sure there is optimal performance of our classifiers, we employed a comprehensive hyperparameter tuning process using nested cross-validation combined with grid search. This approach is illustrated in Figure 3.5.

Nested Cross-Validation with Grid Search:

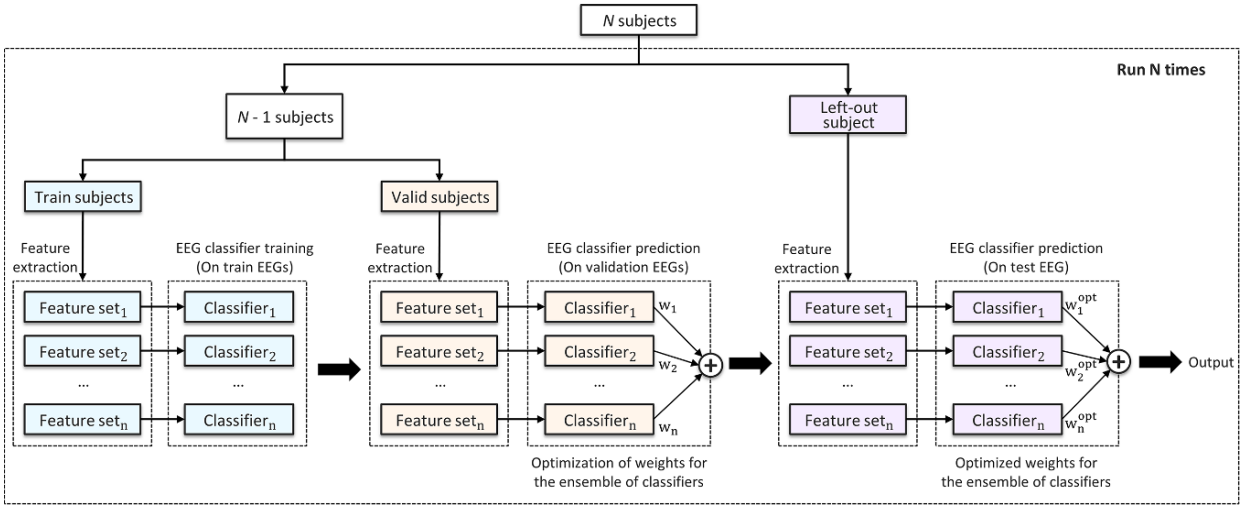


Figure 3.4: LOSO CV process on combination of features [6].

Nested cross-validation is a robust technique to simultaneously optimize hyperparameters and evaluate model performance [75]. It consists of two nested loops: an outer loop for model evaluation and an inner loop for hyperparameter tuning.

- **Outer Loop (Model Evaluation)**
  - The EEG dataset is divided into multiple combinations for cross-validation in which the training and testing datasets are stratified to maintain the proportion of classes.
  - We use stratified group 5-fold cross-validation (CV) for the outer loop. In each iteration, the dataset is split into five folds: four folds for training and one fold for testing.
  - The performance metrics, Area Under the Curve (AUC) and Balanced Accuracy (BAC), are computed based on the predictions on the test set.
- **Inner Loop (Hyperparameter Tuning)**
  - Within each training set of the outer loop, we further split the data using stratified group 4-fold cross-validation for hyperparameter tuning.
  - A grid search is performed over a predefined hyperparameter space which can be changed depending on the feature set that we extract. Each combination of hyperparameters is evaluated by training the model on two folds and validating it on the remaining fold.
  - The best combination of hyperparameters making the highest validation score is selected.
- **Grid Search**
  - The grid search explores various hyperparameter combinations within the specified grid space.

- Each hyperparameter combination is evaluated in the inner loop to identify the best configuration.

- **Hyperparameter Selection and Model Training**
  - The best hyperparameter combination from the inner loop is used to train the final model in the outer loop.
  - This process is repeated for all outer folds checking that each data point is used for both training and testing.

- **Performance Metrics**
  - The performance of the model is assessed using AUC and BAC metrics.
  - The mean AUC and BAC scores from the outer loop provide a robust estimate of the model's performance.

This nested cross-validation approach [76] is integrated with the LOSO CV process described earlier. For each dataset with the selected feature, we train the classifier on $N-1$ subjects and evaluate it on the Nth subject repeating this step N times to evaluate all subjects. The best hyperparameters are selected through the nested cross-validation process providing optimal model performance. This nested cross-validation

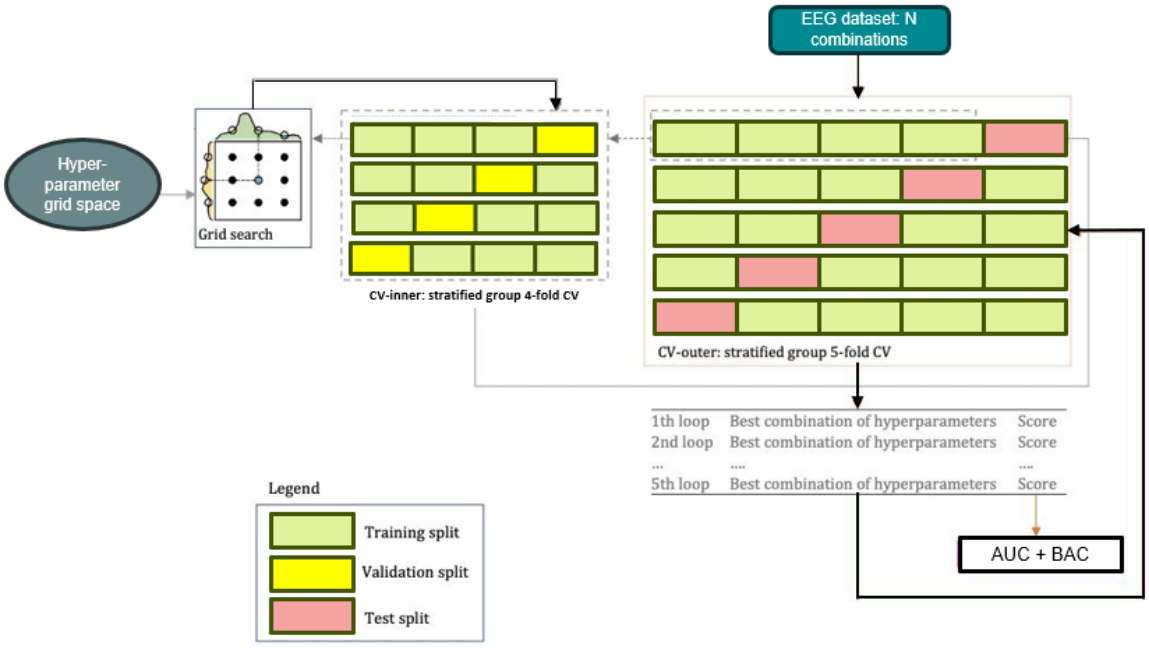


Figure 3.5: Nested cross-validation with grid search for hyperparameter tuning. The inner loop performs 4-fold cross-validation for hyperparameter selection while the outer loop performs 5-fold cross-validation for model evaluation. Algorithm is taken from [77]. [NOTE: The best hyperparameters are selected based on the highest AUC and BAC scores]

approach ensures that our model is validated and the selected hyperparameters lead to optimal performance reducing the risk of overfitting and improving the model to

Table 3.4: Default Parameters for XGBoost

Parameter	Default Value
n_estimators	100
learning_rate	0.1
max_depth	6
subsample	1
colsample_bytree	1
gamma	0
reg_alpha	0
reg_lambda	1

unseen data. XGBoost Default Parameters are shown in Table 3.4.
Hyperparameter Space for Grid Search in this thesis is shown in Table 3.5. The XG-

Table 3.5: Hyperparameter Space for Grid Search

Parameter	Grid
n_estimators	[50, 100, 150, 200]
learning_rate	[0.01, 0.1, 0.2, 0.3]
max_depth	[3, 4, 5, 6, 7, 8]
subsample	[0.6, 0.7, 0.8, 0.9, 1.0]
colsample_bytree	[0.6, 0.7, 0.8, 0.9, 1.0]
gamma	[0, 0.1, 0.2, 0.3, 0.4, 0.5]
reg_alpha	[0, 0.01, 0.1, 1]
reg_lambda	[1, 1.5, 2, 2.5, 3]

Boost classifier was chosen due to its robustness and ability to provide feature relevance [78]. The 'sample_pos_weight' parameter in XGBoost was adjusted to handle class imbalance while other hyperparameters were set to their default values.
In both stages, BAC was reported for 80% sensitivity to standardize the results [6]. The mean LOSO results are reported for both datasets providing a comprehensive evaluation of the model's performance to compare and check on which dataset this model has the best performance.

3.4 Transfer learning

The transfer learning method used in this thesis is a combination of using STFT images extracted from each EEG file which is then fed into a pre-trained model and a classifier [55]. The raw EEG data loaded from EDF files was preprocessed to remove light stimulation segments showing that only relevant brain activity was analyzed. After resampling the data to a uniform frequency (200 Hz), the signals were normalized to maintain consistency across different EEG recordings. STFT is used to convert

the EEG signals into a time-frequency representation i.e. in this case scalograms by segmenting the EEG signal into short windows. A Sobel filter was applied to the magnitude scalograms derived from the STFT for the edges of the image [79]. This method provides a 2D matrix where one axis represents time and the other represents frequency. The pre-trained weights from the ImageNet dataset are used in this case and also to make the image in the correct format i.e. 3 channel (RGB) image and resizing to 224 x 224 pixels input size. The epilepsy channels used are flattened to make a single epilepsy signal that is then converted into an STFT and the same goes for the non-epilepsy signals. A window size of 64 is used which gives more temporal resolution in this case.

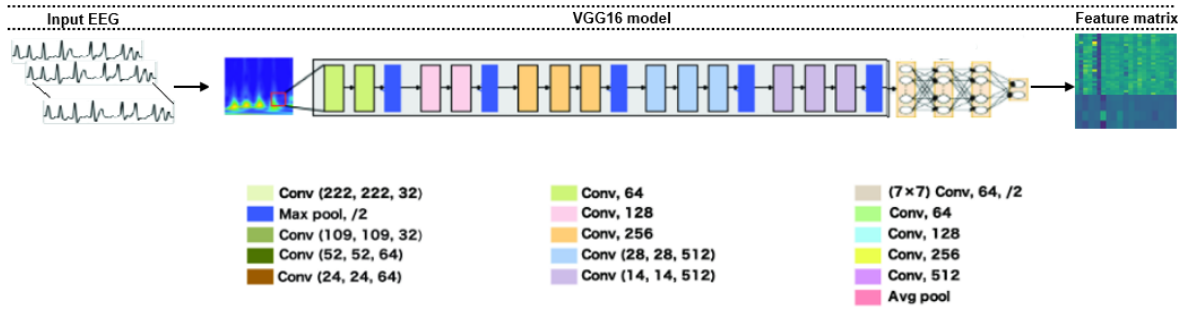


Figure 3.6: A transfer learning technique used to load the weights of the convolutional part of the network and features are extracted from the spectrogram/scalogram input [55].

STFT for Epilepsy detection: The STFT-transformed data is also processed using the VGG16 model. The features extracted by VGG16 are then used as input to an SVM classifier. This approach leverages the feature extraction capabilities of VGG16 and the robust classification performance of SVM using Mahfuz et al. approach [55]. In addition to this, besides using the SVM classifier, the XGBoost and Random forest classifiers with LOSO CV are also used to evaluate the model's performance. Besides this, regularization parameters are added to the XGBoost classifier to prevent overfitting. To do additional tests, a hold-out test set is made which contains 5 STFT's from each class. They are processed and not used in the training data where the model is trained excluding the hold-out test set. After this, the model is tested using the hold-out test set to evaluate the performance model.

3.5 Evaluation

3.5.1 Statistical analysis

To check the accuracy and consistency of the data processing pipeline, we compared the results of pre-processing and feature extraction between MATLAB and Python implementations. The comparison was based on two key metrics: Pearson correlation coefficient and Normalized Root Mean Square Error (NRMSE). In addition to this, after computing the features for both epileptic and healthy patients, the Mann-Whitney U test is used to compare the distributions of features between these two groups.

Pearson Correlation Coefficient

The Pearson correlation coefficient (r) measures how closely two datasets are related in a linear manner [80]. Further, the value of r is calculated for each feature by comparing the results obtained from MATLAB and Python. This coefficient varies between 1 and 1 with 1 showing a linear connection 1 indicating a strong negative linear relationship and 0 implying no linear association. The formula for the Pearson correlation coefficient is shown in Equation 3.2 [80].

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{3.2}$$

where x_i and y_i are the data points from MATLAB and Python, respectively, and $\bar{x}$ and $\bar{y}$ are their means. By calculating the correlation for each data channel, the degree of similarity between the two sets of results was analyzed.

Normalized Root Mean Square Error (NRMSE)

NRMSE is used to measure the difference between the values predicted by the Python implementation and the actual values from MATLAB. It is normalized by the range of the data to provide a relative measure of error. The formula for NRMSE is shown in Equation 3.3 [81, 82].

$$\text{NRMSE} = \frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}}{\max(y) - \min(y)} \tag{3.3}$$

where y_i are the observed values from MATLAB, $\hat{y}_i$ are the predicted values from Python, and n is the number of observations. NRMSE provides a normalized measure of the differences making it easier to compare across different scales of data.

The data is loaded in both formats and a unit test is done to check data sizes and min/max values. For each channel, pearson correlation and NRMSE is calculated providing a linear relationship and error metric to check the differences. The results are then aggregated accross all channels and analyzed. Files with low correlation values (below 0.9) were identified for further investigation to check any discrepancies. This comparison helped validate the integrity of the Python implementation so that it can be used in further analysis and research.

Mann-Whitney U Test for Feature Comparison

After computing the features for both epileptic and healthy patients, the Mann-Whitney U test is done to compare the distributions of these features between the two groups. It is used to determine whether there is some difference between the distributions of two independent samples. Unlike parametric tests, the Mann-Whitney U test does not assume a normal distribution, making it suitable for ordinal data or non-normal distributions and the test statistic U is calculated using Equation 3.4 [83, 84].

$$U = n_1 n_2 + \frac{n_1(n_1+1)}{2} - R_1 \tag{3.4}$$

where n_1 and n_2 are the sample sizes of the two groups, and R_1 is the sum of the ranks for the first group. The resulting U value is compared to a critical value from the Mann-Whitney U distribution to determine statistical significance.
By applying the Mann-Whitney U test, the distribution of features can then be analyzed and differed between epileptic and healthy patients which can providing insights into the distinguishing characteristics of these groups.

3.5.2 Metrics

The classification model is examined using performance metrics. Performance metrics are essential for evaluating and comparing machine learning models. For binary classification, these metrics are based on the ratios of four possible outcomes that is usually displayed in the confusion matrix when you visualize it: True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN).

Confusion Matrix: The confusion matrix shows the number of correctly and incorrectly classified instances for each class where the structure is as follows: As

Table 3.6: Confusion Matrix

	Predicted Positive	**Predicted Negative**
Actual Positive	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)

shown in Table 5.8, the definitions of True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) are important. TP refer to cases where the model correctly identifies patients who have the condition as positive and TN represents instances where the model accurately classifies patients without the condition as negative. FP occur when the model incorrectly identifies a healthy patient, whereas FN are cases where the model fails to detect the condition in patients who actually have it which could result in incorrect diagnosis.

Accuracy (ACC): This metric is the ratio of the correctly predicted EEGs / patients to the total number and is defined as Equation 3.5. Note that the mean accuracy is calculated across all folds.

$$\text{ACC} = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.5}$$

Balanced Accuracy at 80% Sensitivity ($\text{BAC}_{0.8}$): Balanced Accuracy ensures that sensitivity is at least 80%. It is calculated as shown in Equation 3.6.

$$\text{BAC}_{0.8} = \frac{0.8 + \frac{TN}{TN+FP}}{2} \tag{3.6}$$

Receiver Operating Characteristic (ROC) Curve: A typical ROC curve can be seen in Figure A.15. It plots the True Positive Rate (TPR) against the False Positive Rate (FPR) at various threshold settings. The True Positive Rate is also known as sensitivity and the False Positive Rate is calculated as shown in Equation 3.7 [6].

$$\text{FPR} = \frac{FP}{FP + TN} \tag{3.7}$$

Area Under the Curve (AUC): The AUC provides a single value to evaluate the classifier's performance across all threshold levels. A higher AUC value indicates better classifier performance for epilepsy diagnosis. This is computed using the scikit-learn package in Python. These evaluation metrics, including AUC and $\text{BAC}_{0.8}$, are important for assessing the performance of the machine learning models used in this thesis. They provide comprehensive results into the models' effectiveness in distinguishing between epileptic and healthy EEG patients so that the models achieve high sensitivity and specificity which is the ratio of $TP/(TP + FP)$ which is the positive predicted value. This gives information to the neurologist to see how many of his patients have epilepsy or not and the same goes for the negative predicted value which can be found from the confusion matrix.

4 Statistical analysis

4.1 Overview

This chapter presents the statistical analysis conducted on EEG features from both the TUH and EMC datasets. The Mann-Whitney U test is applied to the extracted features to check significant differences between healthy and epileptic EEGs. Research shows that this test is used in multiple research papers to see the involvement of features in the feature vector where the features (spectral & phase) in each feature vector was observed [85, 86]. Note that this is the statistical analysis of the data without IED's regarding the TUH dataset.

4.2 TUH Dataset

Table 4.1 provides a summary of the statistical analysis for the different feature sets in the TUH dataset. For each feature set, the best performing montage was identified based on the lowest p-value obtained. The percentage of significant features, the lowest p-value, the index of the feature with the lowest p-value, the mean p-value, and the standard deviation of p-values are displayed. The statistical analysis shows

Table 4.1: Summary of Statistical Analysis for Different Feature Sets

Feature	**Montage**	**%**	**Lowest [p]**	**Index**	**Mean [p]**	**Std [p]**
S	Laplacian	4.21	2.04×10^{-5}	73	0.39	0.31
CWT	BipolarDB	42.74	1.89×10^{-13}	11	0.16	0.26
DWT	BipolarDB	49.07	1.89×10^{-13}	7	0.13	0.25
CC	CAR	7.60	2.68×10^{-7}	46	0.28	0.31
PLV	Laplacian	9.84	1.03×10^{-9}	1000	0.32	0.32
UTM	BipolarDB	34.19	1.89×10^{-13}	169	0.22	0.30
GCC	CAR	15.00	N/A	N/A	N/A	N/A
GPLV	CAR	10.00	N/A	N/A	N/A	N/A
mST	Laplacian	16.67	3.50×10^{-12}	71	0.23	0.29
sST	BipolarDB	26.85	1.89×10^{-13}	72	0.21	0.29

several important effects in the epileptic EEGs from the TUH dataset. The feature sets CWT, DWT, UTM, and sST showed the highest percentage of significant features with more than 25% of their features being significant. Statistical features derived from the Laplacian and BipolarDB montage also show significant differences which provides more variability and higher-order statistical properties that can distinguish healthy and epileptic brains. However, cross-correlation was not as significant as the other features but still provided valuable insights with the CAR montage. In addition to this, the

graph metrics have $p > 0.001$ so they were marked with N/A. This could mean that the results obtained from the graph metric features are not significant due to their high p-values. The box plot in Figure 4.1 shows the distribution of the most significant

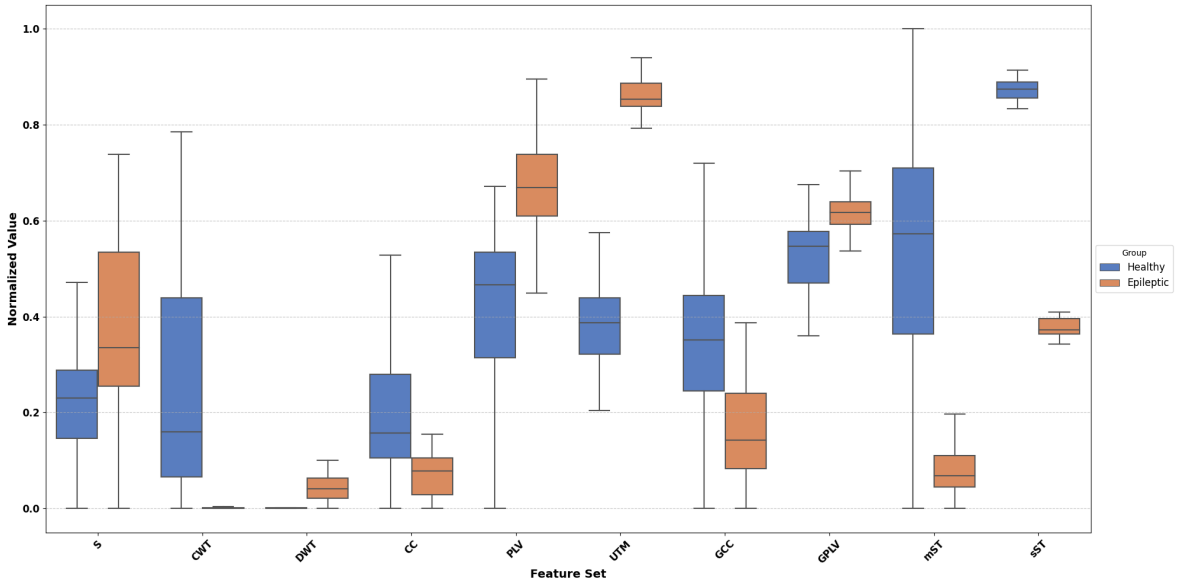


Figure 4.1: Distribution of the most significant features for each feature set in the TUH dataset comparing healthy and epileptic EEGs.

features for each feature set, comparing healthy and epileptic EEGs. The consistent differences across various feature sets underscore the multifaceted nature of epileptic EEGs, where both temporal, spectral, and spatial characteristics are affected.

4.3 EMC Dataset

Similarly, Table 4.2 provides a summary of the statistical analysis for the different feature sets in the EMC dataset with finding out best montages, feature index in the feature vector and the p-value statistics.

The statistical analysis reveals several important effects in the epileptic EEGs from the EMC dataset. The feature sets CWT, DWT, and sST showed the highest percentage of significant features meaning their involvement in distinguishing between healthy and epileptic EEGs. The statistical features derived from the BipolarDB and Cz montages were most effective highlighting their use in capturing the relevant EEG patterns. In contrast, cross-correlation and graph-based features (GCC and GPLV) were less significant suggesting that they may capture less different information in this dataset. The box plot in Figure 4.2 shows the distribution of the most significant features for each feature set comparing healthy and epileptic EEGs.

Figure A.19 provides a detailed observation of the top 10 significant features for each feature set further showing the specific features that differentiate healthy from

Table 4.2: Summary of Statistical Analysis for Different Feature Sets

Feature	**Montage**	**%**	**Lowest [p]**	**Index**	**Mean [p]**	**Std [p]**
S	Cz	5.56	$5.86e^{-6}$	6	0.30	0.31
CWT	BipolarDB	17.52	$8.52e^{-13}$	433	0.28	0.31
DWT	BipolarDB	15.97	$8.52e^{-13}$	394	0.29	0.31
CC	Laplacian	10.53	$1.64e^{-10}$	166	0.35	0.31
PLV	BipolarDB	7.30	$6.35e^{-11}$	867	0.31	0.31
UTM	CAR	19.43	$8.52e^{-13}$	192	0.25	0.30
GCC	BipolarDB	5.00	N/A	N/A	N/A	N/A
GPLV	BipolarDB	3.00	N/A	N/A	N/A	N/A
mST	Cz	4.63	$4.63e^{-10}$	82	0.31	0.29
sST	Cz	15.74	$8.52e^{-13}$	12	0.33	0.31

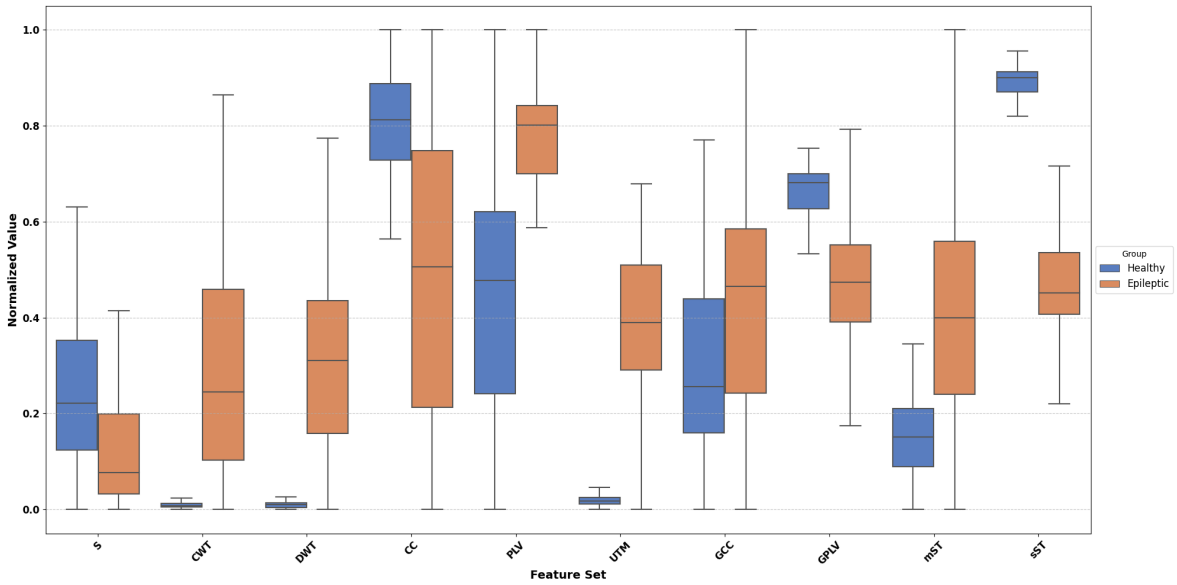


Figure 4.2: Distribution of the most significant features for each feature set in the EMC dataset comparing healthy and epileptic EEGs.

epileptic EEGs. Moreover, the analysis of both the TUH and EMC datasets reveals consistent patterns in significant features. Both datasets highlighted the importance of CWT, DWT, and sST features providing their effectiveness in distinguishing between healthy and epileptic EEGs.

4.4 Feature Extraction Results

Besides the Mann=Whitney test, the distribution of power in different frequency bands is looked into to observe how different the data is in both datasets. Figure 4.3 shows the distribution of power in different frequency bands for healthy and epileptic patients where here we compare both datasets i.e. TUH and EMC. The distribution of relative power in different EEG frequency bands (Delta, Theta, Alpha, Beta) across four groups: Epileptic with IEDs (TUH), Epileptic without IEDs (TUH), Healthy (EMC), and Epileptic (EMC) is calculated for all the patients. A few observations were noted.

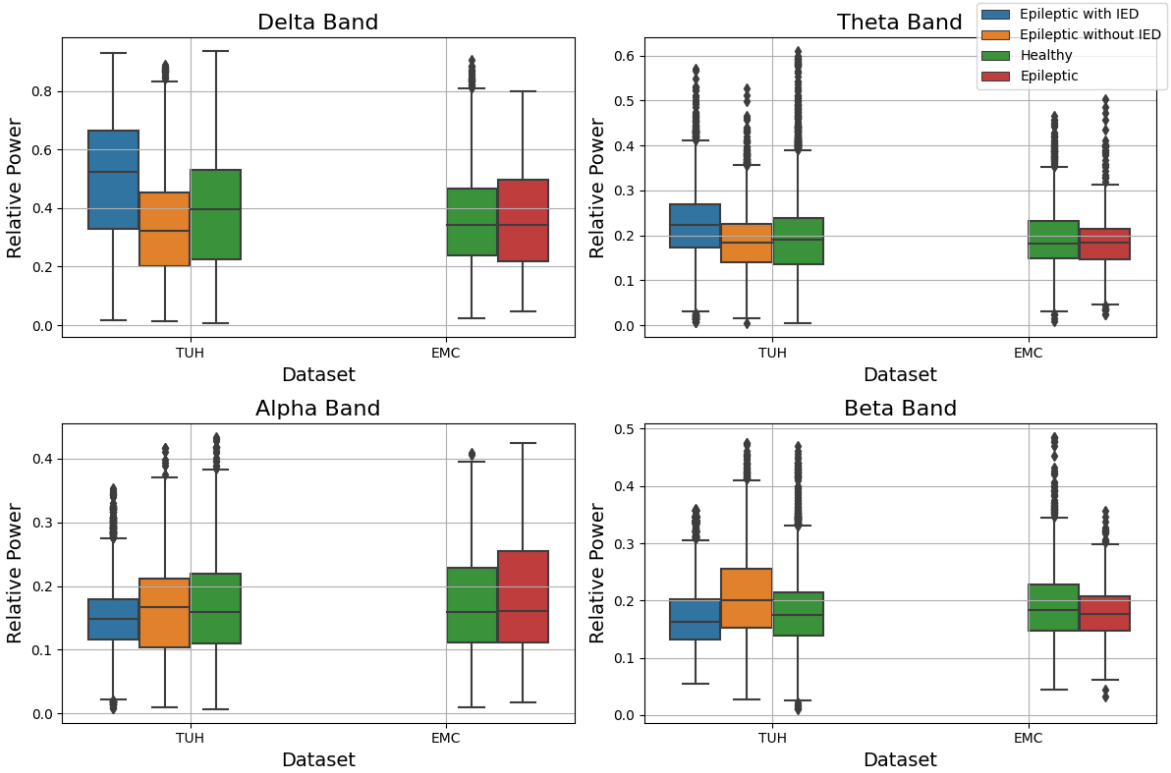


Figure 4.3: Distribution of Power in Different Frequency Bands.

The epileptic patients with IEDs have higher relative power in the Delta band compared to other groups in the TUH dataset suggesting a significant difference in slow-wave activity noted by J. Dauwels. The relative power in the Theta band is slightly higher in epileptic patients with IEDs in the TUH dataset but shows less variation in the EMC dataset. The Healthy group in the TUH dataset has lower Alpha power whereas the EMC dataset shows similar values of the Alpha power between Healthy and Epileptic groups. The Beta band shows similar values too with the highest variability observed in the TUH dataset's Epileptic without IEDs group.

4.5 VGG16 model features

After the VGG16 model processing, the feature matrix contain each feature vector of a patient is obtained. To compare the significant features, the Mann-Whitney U test is used for both the epileptic and healthy datasets. The top significant features for the TUH dataset with very low p-values are shown in Figure 4.4. Note that there are only 6 of the features shown in the figure that have very low p-values that meet the criteria. The other comparisons show that there are very similar traits. As shown in Figure 4.5,

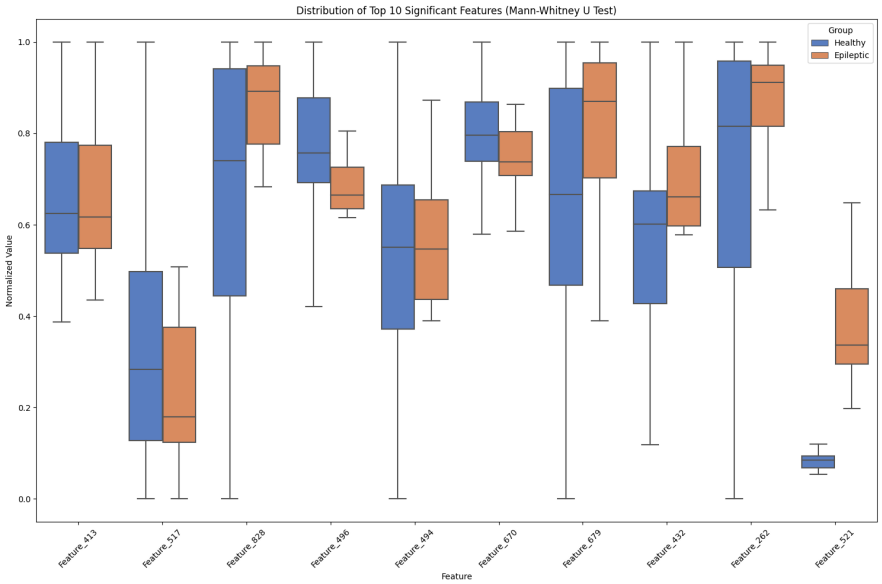


Figure 4.4: Distribution of Top 10 Significant Features (Mann-Whitney U Test) for the TUH dataset

the heatmap of the feature distribution from the Mann-whitney U test is displayed. The features i.e. 578, 743, and 776 show a lot of difference when comparing the mean

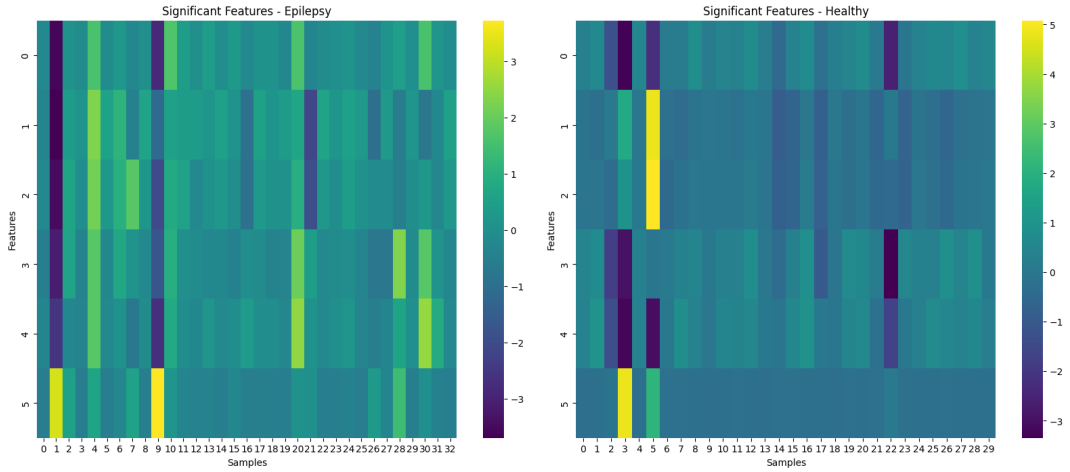


Figure 4.5: Heatmap of healthy vs epileptic significant feature distribution of the TUH dataset

values across each patient. These variations are important showing the distrubution of values with key differences that can influence the output of the classifiers. As there are many features that have very low p-values for the EMC dataset, the top 10 features that have the most differences between epilepsy and healthy class is shown in Figure 4.6. Here, it can be noted that there are some huge distinctions between both the classes compared the TUH dataset which shows that these features can be very useful for classification purposes. As shown in the box-plot, certain features i.e. at index 410 and

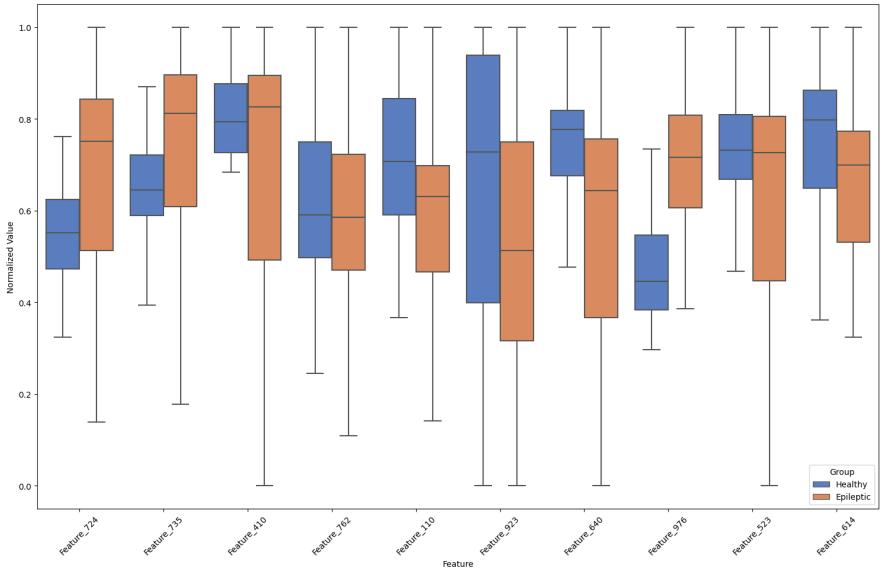


Figure 4.6: Distribution of Top 10 Significant Features (Mann-Whitney U Test) for the EMC dataset

762 shows distinct median differences between the two classes showing strong indicators. The variation across the features shows the importance of feature selection to develop improved classification results. The heatmap for the epileptic data shows a more varied

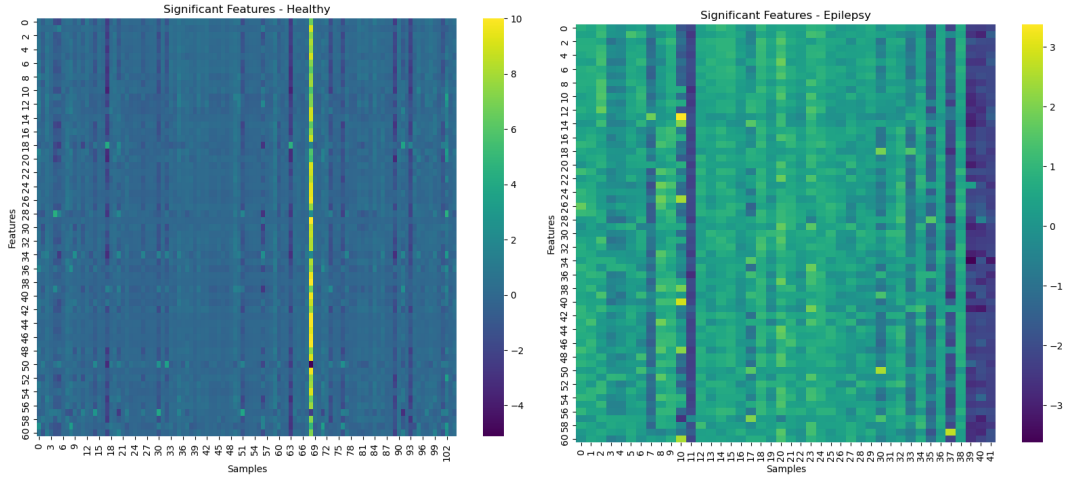


Figure 4.7: Heatmap of healthy vs epileptic significant feature distribution of the EMC dataset

and distributed pattern across different patients whereas for the healthy patient data, it is more uniform with less variance across the features showing consistency. Features show that there were more variations across the patients in epileptic EEGs which is important for differentiating with the healthy patient data. Further analysis of these features are important as there are a lot of features in the feature vector that has an influence on the classification results. Besides this, the feature maps shown in Figure A.20 and Figure A.21 shows the key differences in how the STFT images for both epileptic and non-epileptic patients change after each filter and layer in the VGG16 model.

5 Results

This chapter presents the results of the analysis on three EEG datasets: TUH (with IEDs and without IEDs datasets) and EMC. First, the results obtained from MATLAB and Python are compared to validate the consistency and reliability of the used methods. The findings are organized into sections corresponding to the different stages of the analysis used in this thesis including feature extraction, model performance, evaluation metrics, and hyperparameter tuning. We conclude the chapter with a comparison to baseline results and a discussion of the implications of our findings. Second, validating our method by comparing the results of Epileptic with IEDs vs. healthy patients to the baseline results reported by Thangavel et al is done as it is also one of the research questions for this thesis.

5.1 Correlation and NRMSE Analysis Between MATLAB and Python

To ensure the consistency and reliability of the pre-processing steps between MATLAB and Python, the mean correlation was calculated after each significant pre-processing step. As shown in Figure 5.1, the mean correlation remains consistently high across all steps, indicating a high level of agreement between the two platforms.

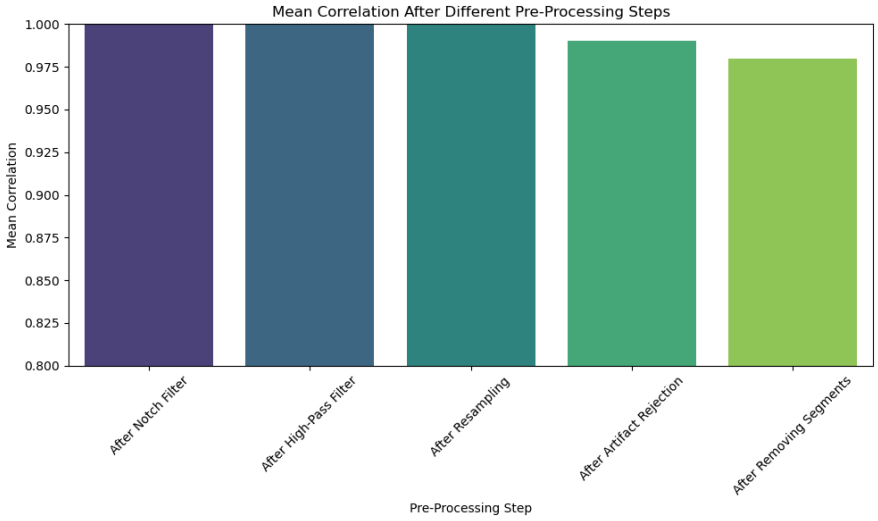


Figure 5.1: Mean Correlation After Different Pre-Processing Steps.

The accuracy of feature extraction methods was assessed using Pearson correlation co-efficients and Normalized Root Mean Square Error (NRMSE) between MATLAB and Python implementations. The mean correlation and NRMSE for different feature

sets are presented in Figure 5.2. The results show that most features have high mean correlation values showing that the features extracted from both versions are very similar. The NRMSE values are low for most features except with the CWT feature having the highest NRMSE due to scaling differences in Python. This is due to the 'morl' wavelet being slightly different in both versions.

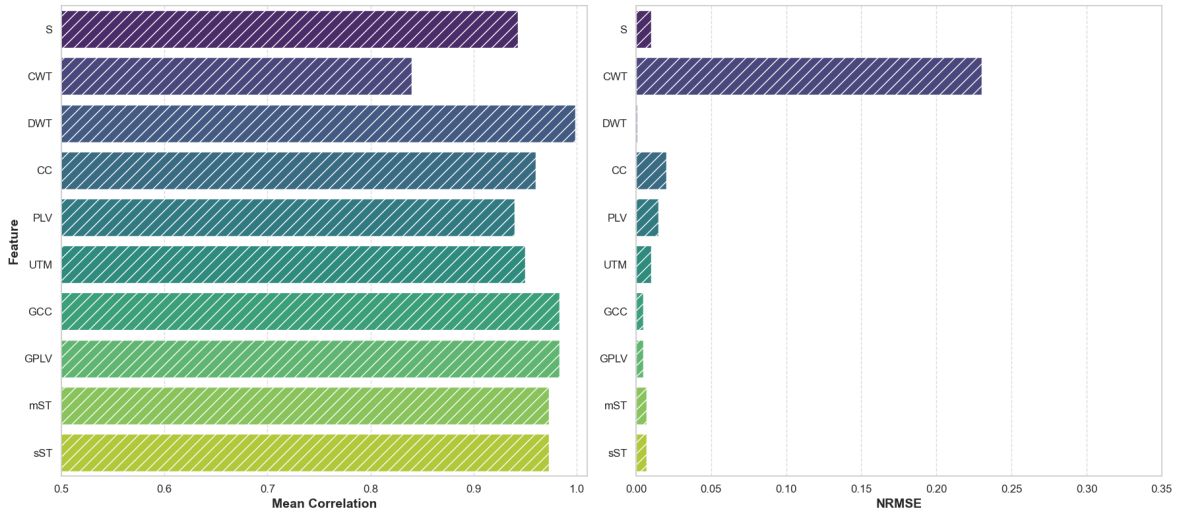


Figure 5.2: Mean Correlation and NRMSE for Different Feature Sets.

These evaluations confirm that the pre-processing steps and feature extraction methods used are consistent and reliable across MATLAB and Python implementations ensuring the robustness.

5.2 Model Performance

5.2.1 Epileptic with IEDs vs. Healthy (TUH Dataset)

First, a comparison of the Area Under the Curve (AUC) and Balanced Accuracy (BAC) metrics for various feature sets is evaluated under three conditions: MATLAB implementation (baseline i.e. Thangavel et al. paper), without age and vigilance features, and with age and vigilance features. The feature sets include Spectral (S), Cross-Correlation (CC), Phase Locking Value (CPLV), Stockwell Transform features (ST SR and ST P), Discrete Wavelet Transform features (db4 and morl), Univariate Temporal Measures (UTM) , and Graph-based features (Cnetwork and Pnetwork). As shown in Figure 5.3, the individual features show very less deviations between the AUC and BAC values across all conditions indicating their robustness in distinguishing between epileptic and healthy EEG signals. This can also be seen in Table C.1.

Cross-Correlation (CC) and Phase Locking Value (CPLV) show lower AUC and BAC values compared to Spectral features with CC benefiting slightly from the inclusion of age and vigilance features while CPLV remains relatively unaffected. The Stockwell Transform features demonstrate a notable difference, with ST SR

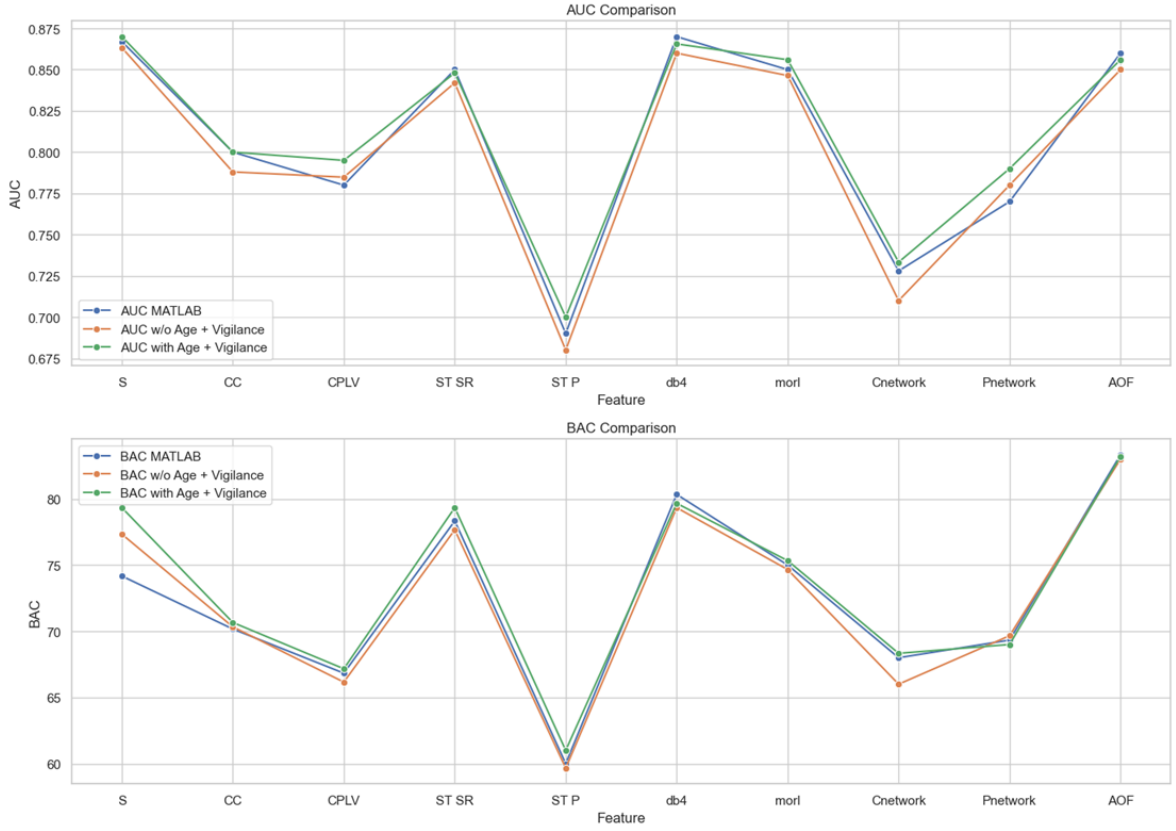


Figure 5.3: Comparison of AUC and BAC metrics for different feature sets across three conditions: MATLAB implementation (baseline), without age and vigilance features, and with age and vigilance features.

maintaining high AUC and BAC values, whereas ST P exhibits a significant drop especially without the additional features. Both Discrete Wavelet Transform feature sets (db4 and morl) perform reliably, with high AUC and BAC values, showing minimal impact from age and vigilance features.
Graph-based features (Cnetwork and Pnetwork) show variability, with Cnetwork generally outperforming Pnetwork, and both benefiting from the additional demographic features. The rest of the feature vector sets shows similar performance showing that the methods has been validated properly.

5.2.2 Epileptic without IEDs vs. Healthy (Both Datasets)

Next, we apply our validated methodology to both the TUH and EMC dataset which consists of epileptic patients without IEDs and healthy patients.

The performance of the models is evaluated using ROC curves and confusion matrices. Figure 5.4 & Figure 5.5 shows the ROC curves for the top-performing models on both datasets.

The plot shown includes multiple ROC curves for different feature sets. The `GPLV` feature stands out with the highest AUC of 0.72 showing it is the most effective feature

for class discrimination in this dataset. Features like PLV, S, and sST also show good performance with AUC values ranging from 0.65 to 0.71. Other features such as CWT, DWT, mST, CC, GCC, and UTM have lower AUC values suggesting they are less effective in distinguishing between classes. The plot includes multiple ROC curves, each repre-

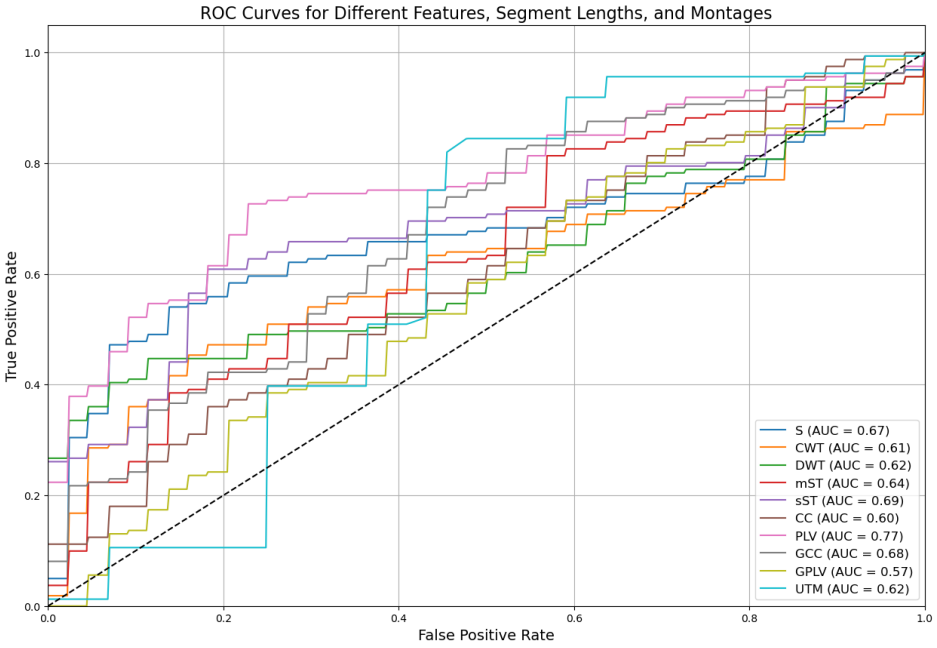


Figure 5.4: ROC Curves for Top-Performing combinations on TUH Dataset w/o IEDs.

senting a different feature set or combination of features. The PLV feature shows the highest AUC of 0.77 showing it has the best discriminatory power among the features tested. The S, sST, GCC, and GPLV features also show reasonable performance with AUC values around 0.67 to 0.71. Features like CWT, DWT, mST, CC, GPLV, and UTM show comparatively lower AUC values indicating less effective differences between classes.

The overall performance across features in the TUH dataset is somewhat lower than the EMC dataset suggesting possible differences in the dataset characteristics or quality. The CWT and mST improved with the addition of the vigilance states by 1 % whereas the other features didn't improve. Both datasets show that certain features such as PLV and sST consistently perform well in distinguishing between classes highlighting their robustness across different datasets. The higher AUC values in the EMC dataset compared to the TUH dataset shows that the EMC dataset has more distinct patterns or less noise, making classification easier.

Table 5.1 presents the BAC (Balanced Accuracy) and AUC (Area Under the Curve) mean differences for the TUH and EMC datasets. The comparisons include different pre-processing configurations comparing the results with and without age and vigilance state where

- **w1:** Difference between with age + state and without age + state
- **w2:** Difference between with age and without age + state

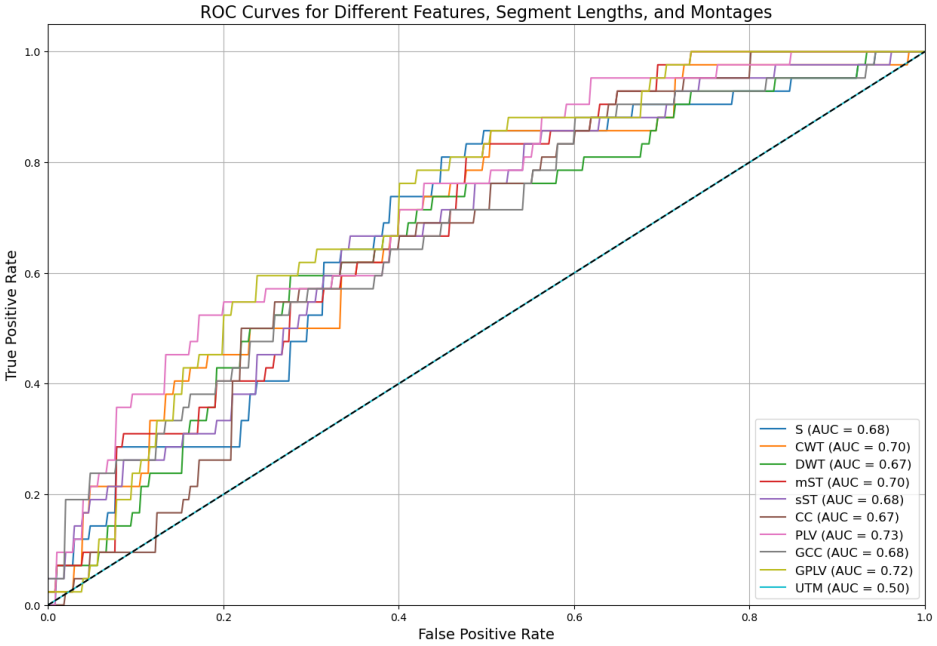


Figure 5.5: ROC Curves for Top-Performing combinations on EMC Dataset.

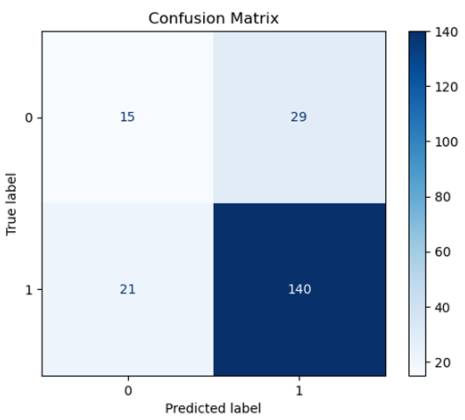
Table 5.1: AUC & BAC Mean Differences Comparing with and without Vigilance State + Age for TUH and EMC Datasets.

Dataset	**Metric**	**Comparison**	**CAR**	**Cz**	**BipolarDB**	**Laplacian**
TUH	BAC	w1	1.48	1.69	1.48	1.44
	BAC	w2	1.37	1.63	1.30	1.44
	AUC	w1	0.02	0.02	0.02	0.02
	AUC	w2	0.01	0.02	0.01	0.01
EMC	BAC	w1	1.44	1.66	1.48	1.46
	BAC	w2	1.38	1.64	1.46	1.47
	AUC	w1	0.02	0.02	0.02	0.02
	AUC	w2	0.01	0.02	0.01	0.01

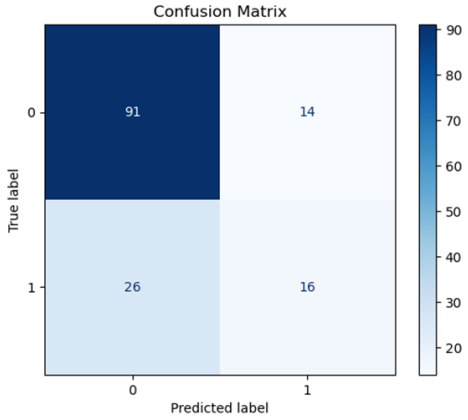
Note that for the EMC dataset, the gender is also added to the feature data.

For the TUH dataset (Figure 5.6a), these values indicate that the classifier was highly effective at identifying positive cases (epileptic) but struggled with negative cases (healthy). The high number of true positives suggests that the model is sensitive and performs well in detecting the epileptic EEGs with IEDs. For the EMC dataset (Figure 5.6b), the classifier displayed a high number of true negatives, indicating strong performance in identifying healthy subjects. However, it had difficulties correctly classifying positive cases (epileptic EEGs without IEDs), as evidenced by the higher number of false negatives.

The confusion matrices (Figure 5.6) reflect the model's performance differences between the two datasets. The TUH dataset shows higher sensitivity which leads to the model's ability to detect epileptic EEGs more effectively compared to the EMC dataset.



(a) TUH Dataset: Feature PLVmean, Segment length: 120, Montage: CAR



(b) EMC Dataset: Feature PLVskew, Segment length: 300, Montage: CAR

Figure 5.6: Confusion Matrices for highest AUC feature-set for TUH (left) and EMC (right) Datasets. (Note that the results of each class are flipped for both datasets as there are more epileptic EEGs in the TUH dataset and more non-epileptic EEGs for the EMC dataset)

The EMC dataset shows higher specificity showing that the model performs better in identifying healthy subjects. These results highlight the importance of dataset characteristics on model performance. The model's high sensitivity in the TUH dataset indicates its suitability for detecting clear epileptic signals. However, its lower sensitivity and higher specificity in the EMC dataset suggest that further improvements are needed for datasets without IEDs possibly through advanced feature extraction or ensemble methods.

5.2.3 Ensemble Method Results

The table below presents the results of the ensemble method for both the Thangavel et al. study and the current thesis version. The performance metrics used are Area Under the Curve (AUC) and Balanced Accuracy (BAC). The combinations of features for each method are listed alongside their corresponding metrics and includes the comparisons between the baseline results from Thangavel et al. and the results from the thesis version using both the TUH and EMC datasets.

The baseline results from Thangavel et al. show that the combination of IED, DWT, and ST-P achieved the best performance with an AUC of 0.787 and a BAC of 73%. Moreover, it shows that both the thesis version and baseline version do match. For the thesis version using the TUH dataset, the best results were obtained with the combination of IED, DWT, and ST-P, achieving an AUC of 0.76 and a BAC of 71.66%. Other combinations, such as S, DWT, and ST-P, also demonstrated competitive performance.

In the EMC dataset, the combination of S, DWT, and PLV yielded the highest AUC of 0.70 and a BAC of 64.33%. These results indicate the effectiveness of different

Table 5.2: Ensemble Method Results Comparison.

Type	Combination	AUC	BAC
Thangavel (baseline version)	IED + DWT + CWT	0.662	58.9
	IED + DWT + ST-P	0.787	73
Thesis version [TUH]	IED + DWT + CWT	0.645	57.43
	IED + DWT + ST-P	0.76	71.66
	S + DWT + ST-P	0.625	55.93
	S + DWT + ST-P	0.71	67.33
Thesis version [EMC]	S + DWT + PLV	0.68	63.55
	S + DWT + sST	0.68	65
	S + DWT + CWT	0.65	64.51
	S + DWT + PLV + sST	0.70	67
	CC + DWT + GCC + mST	0.67	61.38

feature combinations in enhancing epilepsy detection models. The results highlight the potential of various feature combinations to improve the performance of epilepsy detection models. This can be seen in Figure 5.7 Furthermore, the combination of

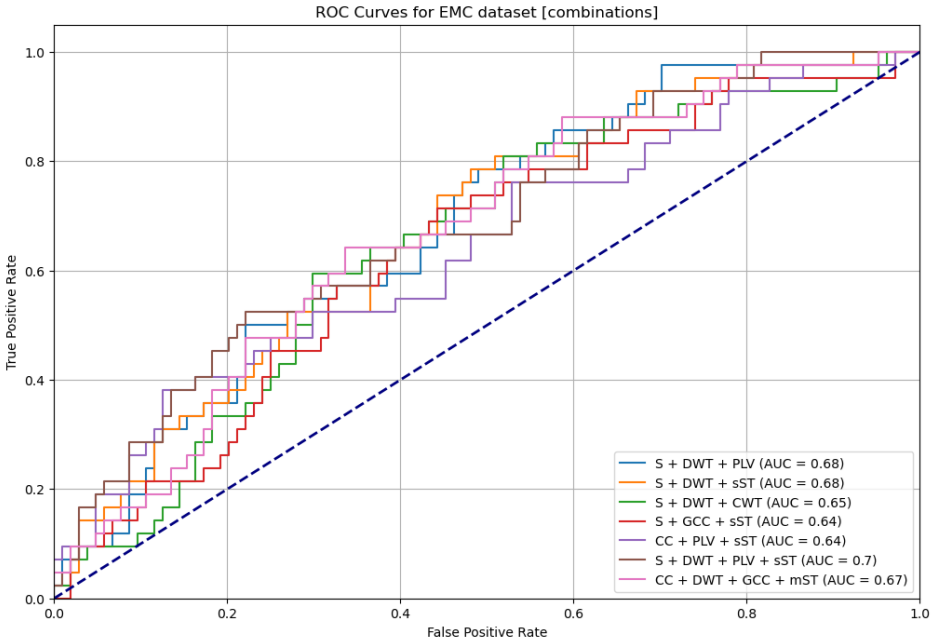


Figure 5.7: ROC Curves for Top-Performing combinations on EMC Dataset.

IED, DWT, and ST-P in both the baseline and the TUH dataset versions and the combination of S, DWT, and PLV in the EMC dataset shows that with hyperparameter tuning, the results i.e. AUC and BAC values can be increased.

5.2.4 Hyperparameter Tuning Results

The optimal hyperparameters were identified through a nested cross-validation with grid search approach as illustrated in Figure 3.5. Table 5.3 lists the best hyperparameters for the XGBoost classifier. There was only a 2 percent increase in the AUC and BAC values. Hence, this approach needs to be looked into again in the future. As shown in Table 5.4, the AUC improved by 0.01 and the BAC improved by 0.33.

Table 5.3: Optimal Hyperparameters for XGBoost Classifier.

Parameter	**Optimal Value**
learning Rate	0.01
max Depth	6
n_estimators	100
subsample	0.9
gamma	0.1

This shows that the algorithm needs to be applied on all combinations besides a single combination as proposed before. The hyperparameter tuning process is also applied to

Table 5.4: TUH Dataset Without IEDs: AUC and BAC Before and After Hyperparameter Tuning [All Feature Sets].

Feature Set	**Before Tuning**		**After Tuning**	
	AUC	**BAC**	**AUC**	**BAC**
S	0.67	65.0	0.68	65.33
CWT	0.61	55.50	0.62	58.0
DWT	0.62	56.0	0.62	58.5
CC	0.60	68.0	0.61	57.5
PLV	0.77	72.0	0.77	71.60
UTM	0.62	58.33	0.63	59.0
GCC	0.68	64.6	0.68	64.3
GPLV	0.57	53.0	0.57	53.0
mST	0.64	61.0	0.65	61.67
sST	0.69	65.0	0.7	67.0

the EMC dataset and similar results were obtained with an increase of AUC/BAC of 1 percent.

5.2.5 XAI method for both datasets

Moreover, Figure 5.8 shows the highest SHAP values for various feature types and montages across the TUH and EMC datasets using the feature importance scores using the Thangavel et al. method. The highest SHAP values for each feature type, segment length and montage combination are shown which provides the most important features. The PLVskewness feature in the BipolarDB montage showed high importance in the

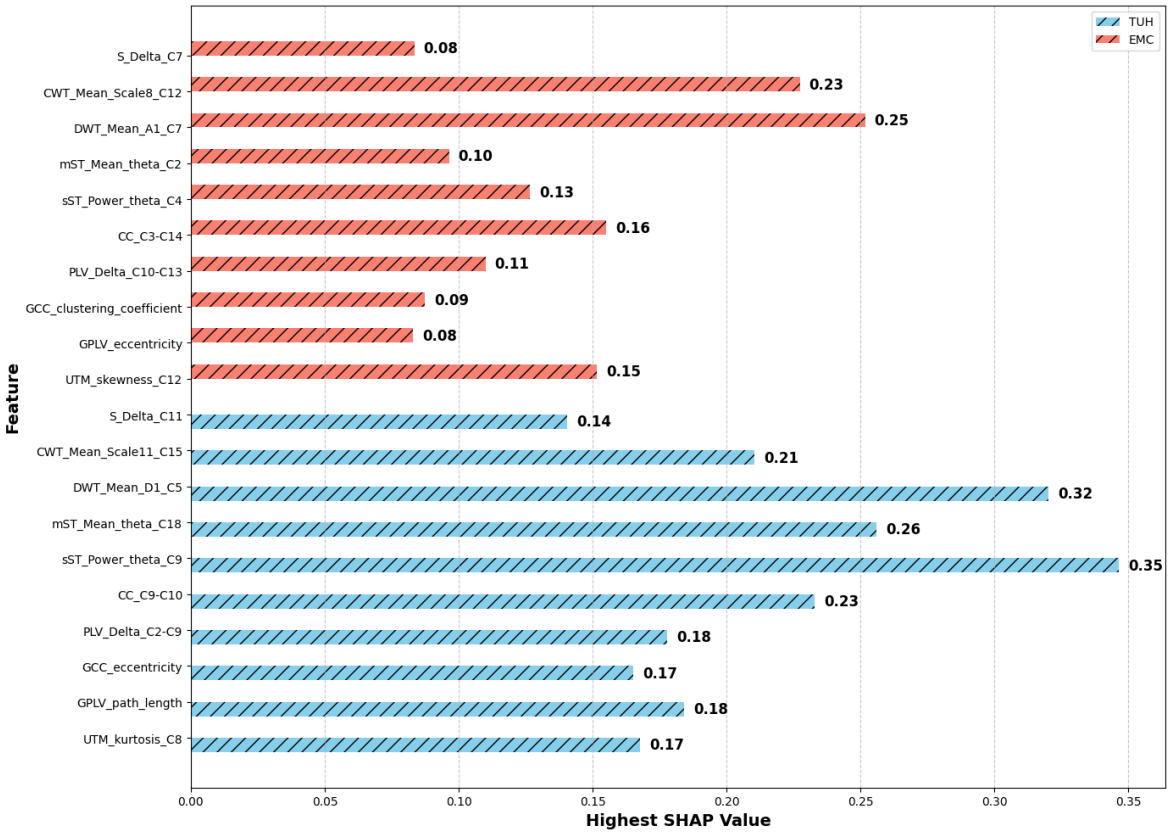


Figure 5.8: Comparison of Highest SHAP Values for Various Feature Types and Montages Across TUH and EMC Datasets.

EMC dataset as shown by the high SHAP value. Whereas, the CWT and DWT feature also gives significant importance in both datasets contributing to the model's outcome.

5.3 Transfer learning Results

5.3.1 TUH dataset

In this section, we present the comparison between the results obtained using Leave-One-Subject-Out Cross-Validation (LOSO CV). To better understand the feature space, the Principal Component Analysis (PCA) on the features extracted by the VGG16 model is performed and can be seen in Figure A.14. The plot shows that the features extracted by the VGG16 model maintain their ability to distinguish between classes even after the dimensionality reduction. The PCA plot shows that the features corresponding to epilepsy (blue dots) and non-epilepsy (orange dots) are distinguishable, however, with some overlap showing that there could be misclassifications. Based on the transfer learning results for the TUH dataset, the following metrics are observed across the three models: XGBoost, SVM, and Random Forest in Table 5.5.

Table 5.5: Metrics comparison using LOSO CV on the TUH dataset

Model	**Accuracy**	**Precision**		**Recall**		**F1-Score**	
		Epileptic	**Healthy**	**Epileptic**	**Healthy**	**Epileptic**	**Healthy**
XGBoost	0.70	0.58	0.61	0.58	0.73	0.63	0.67
SVM	0.52	0.52	0.0	1.0	0.0	0.69	0.0
Random Forest	0.67	0.63	0.61	0.64	0.63	0.63	0.63

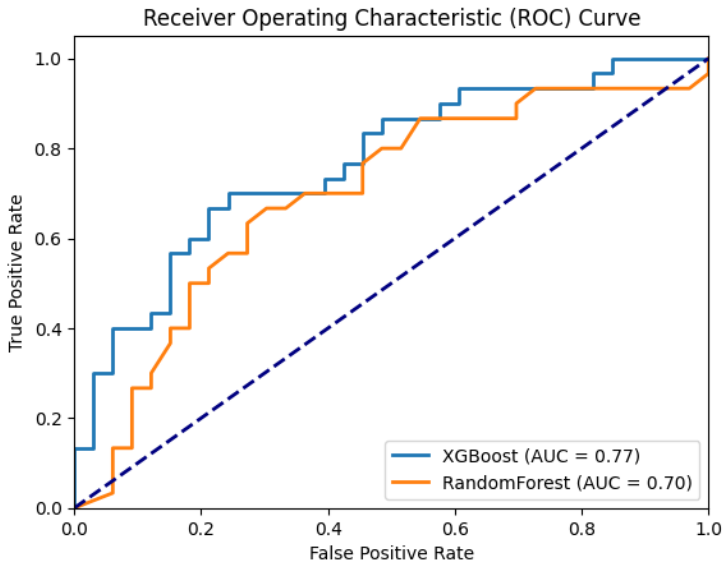


Figure 5.9: ROC curve of multiple classifiers on the TUH dataset using LOSO CV.

As shown in the table, the LOSO CV cross-validation with the XGBoost yields results in higher accuracy, precision, recall, and F1-Score compared to the other models. One possible reason for the poorer performance of the LOSO CV method could be the varying sizes of EEG segments across subjects that can lead to inconsistencies when generating images from these signals. This difference in the content in each image can negatively impact the performance of the models particularly in deep learning contexts where image consistency is critical which is why segmentation might be key for the TUH dataset. This can be considered as a future work.

Figure 5.9 illustrates the Receiver Operating Characteristic (ROC) curves for the classifiers. Note that the SVM classifier is not added to the results as it underperformed with an AUC of 0.4. The poor performance could be due to the complexity of the feature matrix whereas the tree classifiers can handle these feature vectors from the VGG16 model. The ROC curves indicate that the XGBoost and Random Forest models achieve the highest Area Under the Curve (AUC) values showing their robustness and reliability in classifying epileptic versus healthy EEG signals in the TUH dataset.

Furthermore, the ROC curve indicates that the model performs well with an AUC of 0.77. The confusion matrix reveals that the model correctly identifies a majority of the epileptic and non-epileptic cases, although there are some that are not classified correctly that needs to be investigated in the future.

XAI results

As shown in Figure 5.10, the plot in the left provides an overview of the top 10 most important features in the XGBoost model's predictions for epilepsy detection. The bar plot shows the average magnitude of the SHAP values for each feature showing how much each feature has an influence to the model's output. Feature 841 has the most effect compared to the other features in the image array. The second plot in

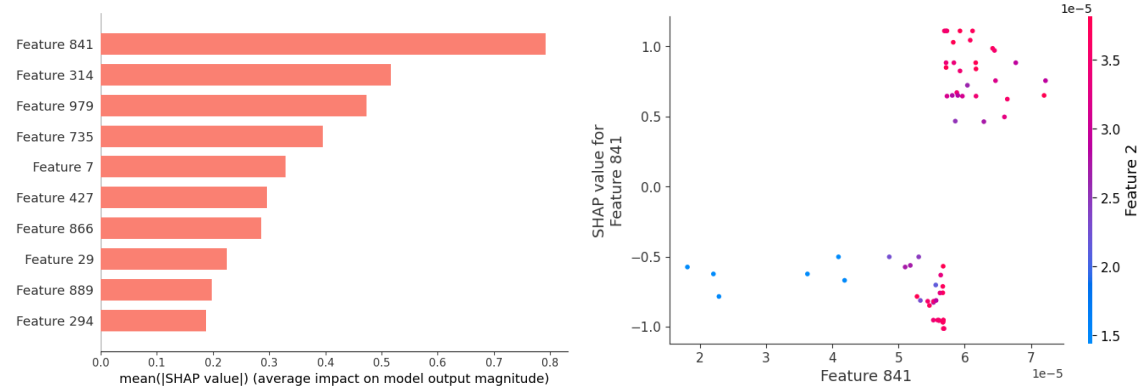


Figure 5.10: Comparison of the top 10 SHAP values for the features extracted using the VGG16 model (left) and highest SHAP value dependence plot for the TUH dataset.

Figure 5.12 is a SHAP scatter plot for Feature 841 showing the relationship between the SHAP value of this feature and its actual value in the dataset. Points higher on the Y-axis give a stronger positive impact on the model's prediction suggesting that higher values of Feature 841 are linked with a higher probability of an EEG being classified as epileptic. The same goes for the lower probabilities in the bottom of the y-axis. The clustering of points also could mean that possible patterns in how these features interact can have a huge impact on the model's performance.

5.3.2 EMC dataset

The results of the epilepsy detection models demonstrate multiple levels of performance across the different classifiers using the VGG16 as the integration of transfer learning. As stated before, the LOSO CV was used for the cross-validation method. The XGBoost model achieved the highest AUC of 0.86 showing a strong ability to distinguish between epileptic and healthy patients. This model also demonstrated

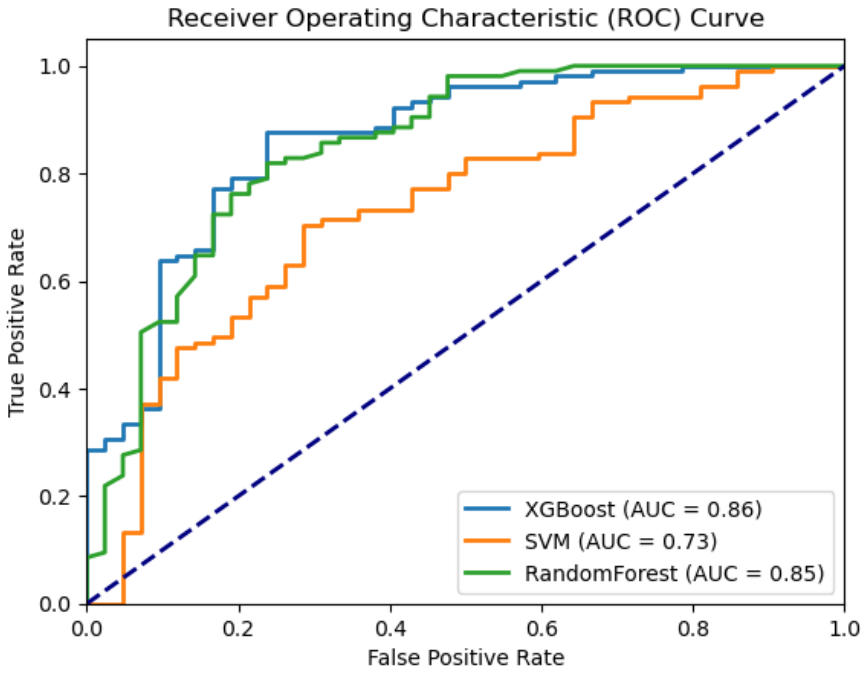


Figure 5.11: ROC curve of multiple classifer on the EMC dataset using LOSO CV.

a balanced accuracy and high precision for detecting epileptic EEGs meaning that it is effective at correctly identifying patients with epilepsy when looking at the positive and negative predicted values. The Random Forest model also performed

Table 5.6: Detailed analysis of LOSO CV results for EMC dataset.

Model	**TP**	**TN**	**FP**	**FN**	**AUC**	**BAC**
XGBoost	92	29	13	13	0.86	83.2
SVM	98	12	30	7	0.74	71.67
Random Forest	96	25	17	9	0.85	82.33

well with an AUC of 0.85 showing similar performance when compared to XGBoost, however, with lower precision and recall for healthy cases. On the other hand, the SVM model showed a lower AUC of 0.74 showing lower performance which struggles with differentiating between the two classes. With false positives being the problem as indicated by its lower precision and recall scores for healthy EEGs, it can be seen that there is lower performance. Table 5.6 gives a further detailed analysis of how many patients were classified properly and how many were not which

could be useful for the neurologist. The true positives and true negatives were very high for both the epilepsy and healthy patients i.e. multi-seizure vs single seizure classification in this case especially for the XGboost classifier. 87.6 percent of the non-epileptic patients and 69.05 percent of the epileptic patients were classified correctly in terms of a neurologist perspective. Overall, while all models performed

Table 5.7: Metrics comparison using LOSO CV on the EMC dataset

Model	**Accuracy**	**Precision**		**Recall**		**F1-Score**	
		Epileptic	**Healthy**	**Epileptic**	**Healthy**	**Epileptic**	**Healthy**
XGBoost	0.82	0.69	0.88	0.69	0.88	0.69	0.88
SVM	0.75	0.63	0.77	0.29	0.93	0.39	0.84
Random Forest	0.82	0.74	0.85	0.60	0.91	0.66	0.88

well, XGBoost and Random Forest are the most robust classifiers for automated epilepsy detection using Transfer learning. This was achieved with handling the data imbalance similar to Thangavel et al.'s approach and optimizing the identification of epileptic EEG patterns through the features extracted from the VGG16 model. To prevent data leakage, several checks were done to check for duplicates / near duplicates.

Additionally, the use of Leave-One-Subject-Out Cross-Validation (LOSO CV) played a vital role in mitigating overfitting by making sure that the model was tested on entirely unseen data from different subjects. This approach to validation not only improves the model's relevance but also provides a more reliable assessment of its true performance in clinical settings.

5.3.2.1 Evaluation on the Hold-Out Test Set

The model was evaluated using a hold-out test set to check its performance on completely unseen data. This evaluation provides insight into the model's ability and its effectiveness when using new patient data.

The model achieved an accuracy of 0.8 (80%) on the hold-out test set showing that it correctly classified 80% of the EEG data. This accuracy suggests that the model performs reasonably well on data it has not encountered during training.

The confusion matrix for the hold-out test set is shown in Table 5.8. The model

Table 5.8: Confusion Matrix for Hold-Out Test Set.

	Predicted Epileptic (0)	**Predicted Non=Epileptic (1)**
Epileptic (0)	TP: 3	FP: 0
Non-Epileptic (1)	FN: 2	TN: 5

correctly identified 3 out of 5 epileptic patients and all 5 non-epileptic samples.

Table 5.9 summarizes the precision, recall, and F1-score for both classes. The

Table 5.9: Performance Metrics for Hold-Out Test Set

Metric	Epileptic	Non-epileptic
Precision	1.00	0.71
Recall	0.60	1.00
F1-Score	0.75	0.83

model shows perfect precision (100%) meaning all predicted Healthy cases were correct. However, the recall was lower at 60%, meaning that the model missed 2 actual epileptic cases. Regarding the healthy patients i.e. non-epileptic, the model achieved perfect recall (100%), correctly identifying all actual cases. Moreover, the model performs better on the healthy class due to more samples present in the training data. This class imbalance costs some false positives in the epileptic class and hence, the trade-off between precision and recall must be considered in the future. In the context of epilepsy diagnosis, incorrectly classifying an epileptic case can have severe consequences. Therefore, the model's performance is important to meet the criteria of the neurologist.

5.3.2.2 XAI results

As shown in Figure 5.12, the plot in the left provides an overview of the top 10 most important features in the XGBoost model's predictions for epilepsy detection. The bar plot shows the average magnitude of the SHAP values for each feature showing how much each feature has an influence to the model's output. Feature 722 has the most effect followed by Features 917 and 6. These features have the largest mean absolute SHAP values suggesting they are the most influential in distinguishing between epileptic and healthy EEG signals. The second plot in Figure 5.12 is a SHAP scatter plot for

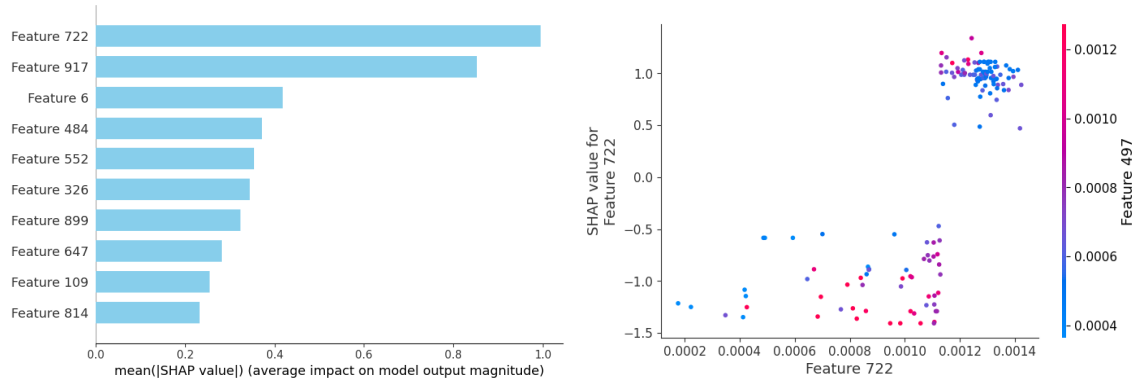


Figure 5.12: Comparison of the top 10 SHAP values for the features extracted using the VGG16 model (left) and highest SHAP value dependence plot for the EMC dataset.

Feature 722 showing the relationship between the SHAP value of this feature and its actual value in the dataset. The color gradient indicates the value of another feature which in this case is randomly chosen due to its lower SHAP value i.e. Feature 497 which gives more information on how Feature 722 interacts with Feature 497 in the model's

decision-making process. The analysis shows similar effects which was discussed in the XAI results for the TUH dataset.

6 Discussion

In this chapter, the results obtained are discussed connecting to the initial research questions and exploring implications of the field fo epilepsy diagnosis. The validation of the pre-processing and feature extraction steps is done by comparing results between MATLAB and Python implementations. As shown in the correlation and NRMSE plots, there is a high degree of agreement between the two versions with mean correlations greater than 0.9 for most features and minimal NRMSE values. This successful correlation shows that the data processing pipeline across different software environments is consistent.

In this study, two methodologies are used, XGBoost with Leave-One-Subject-Out Cross-Validation (LOSO CV) and VGG16-based transfer learning, to analyze and classify epileptic EEG signals from the TUH and EMC datasets. The findings indicate that both methods performed decent with the VGG16 model achieving a high (AUC) of 0.86 and an accuracy of 82%.

Table 6.1: Summary of comparison of our work with other state-of-the-art methods developed using EEG signals.

Study	**Method**	**Dataset**	**Split ratio**	**Results (%)**
Thomas et al. [12]	CNN, Template Matching, S-F classifier	Ep: 33, Np: 30	LOSO CV	AUC = 0.812, Balanced Accuracy = 74.8
Thangavel et al. [87]	1D ConvNet, 2D ConvNet	Ep: 93, Np: 461	LOIO CV	AUC = 0.839, Balanced Accuracy = 78.10
Yang et al. [29]	Time-domain features	Ep: 27, Np: 17	LOSO CV	Accuracy = 79.55, Recall = 81.44, Specificity = 76.47
Aristizabal et al. [88]	CNN & LSTM	Ep: 1360, Np: 240	5 fold CV	Accuracy = 72.54
Syamsundararao et al. [89]	CNN	TUH dataset	10-fold CV	Accuracy: 85.48, Precision: 76.65, Recall: 78.20, F1-score: 69.12

Continued on next page

Table 6.1: Summary of comparison of our work with other state-of-the-art methods developed using EEG signals (continued).

Study	**Method**	**Dataset**	**Split ratio**	**Results (%)**
Mahfuz et al. [55]	STFT or CWT, VGG16, SVM	Multiple datasets	5-fold CV	Mean Accuracy = 87%
Taski et al. [90]	Hypercube, MDWT, NCA, kNN, IMV	TUH dataset	train-test split	Accuracy = 86.23, Sensitivity = 80.16, Specificity = 91.33, Precision = 88.60, F1-score = 84.17
Our method	XGBoost + LOSO CV	TUH dataset / EMC dataset	LOSO CV	AUC = 0.77, BAC = 72.0 / AUC = 0.7, BAC = 67
Our method	VGG16 method	TUH dataset / EMC dataset	LOSO CV	AUC = 0.77, F1-score = 65, AUC = 0.86, F1-score = 79

Our methods were compared against several state-of-the-art techniques from recent studies. Thomas et al. [12] used CNN and template matching achieving an AUC of 0.812 and a balanced accuracy of 74.8%. Thangavel et al. [87] applied 1D and 2D ConvNets obtaining an AUC of 0.839 and a balanced accuracy of 78.10%. Yang et al. [29] employed time-domain features with LOSO CV achieving an accuracy of 79.55%, recall of 81.44%, and specificity of 76.47%. The comparative analysis reveals several important facts. The higher AUC and accuracy of our methods suggest that transfer learning, particularly using pre-trained models like VGG16, can effectively capture complex patterns in EEG signals. This is proven by the PCA visualization which shows distinct clustering of features extracted by VGG16 showing major differences after dimensionality reduction.

The results from the transfer learning approach using the VGG16 model for feature extraction shows significant variations in performance across different classifiers on both TUH and EMC datasets. The EMC dataset outperforms the TUH dataset with these STFT + VGG16 features + XGBoost + LOSO CV. The XGBoost model was the most effective with an AUC of 0.86. This indicates a strong capability in distinguishing between epileptic and healthy EEGs without clear markers i.e. IEDs. The Random Forest model closely followed XGBoost with an AUC of 0.85 showing that it is also highly effective even with lower precision and recall for the epileptic class. On the other hand, the SVM model gives a lower AUC of 0.74 showing a relative difficulty in distinguishing between the two classes with the high number of false positives for healthy EEGs. This outcome highlights the challenges SVM faces when

applied to this type of complex, imbalanced data which other models can perform better.

Future work will focus on expanding the dataset to include more diverse patient demographics and exploring additional deep learning architectures to further improve classification accuracy as shown Table 6.1. In addition to this, XAI techniques such as SHAP values provide better understanding of model predictions showing their acceptance and use in clinical practice.

Despite the above results, several limitations must be discussed. The primary limitation is the potential for overfitting due to the relatively small sample size. While LOSO CV mitigates this risk, larger datasets are necessary to validate the the findings. Furthermore, this study shows the usefulness of advanced machine learning and deep learning techniques in classifying epileptic patients. Future research will aim to address the identified limitations and further refine these models to improve their diagnostic accuracy and reliability.

Future research directions & Conclusion

7

7.1 Research Findings, Limitations and Significance

In this section, the findings from the methods used in this report are discussed by connecting the results to the research questions to check their significance in improving epilepsy diagnosis. The main aim of this research was to create and validate machine learning algorithms for classifying people with epilepsy and those without using different techniques. This research focuses on testing the new techniques against well-known diagnostic standards using the TUH and EMC dataset. We checked how patients with IEDs compared to healthy individuals where the frequency band powers in Figure 4.3 shows the distributions impact. Our findings showed the use of the methods from the initial study conducted by Thangavel et al. which leads to the right direction for the epilepsy diagnosis research. The AUC and BAC values for the TUH dataset containing IEDs closely resembled the results documented in the research paper suggesting that this method of extracting features and classifying them is reliable and successful.

After achieving these results with the TUH dataset, the same method is applied to the EMC dataset too for analyzing and distinguishing epileptic patients without IEDs from healthy individuals. This was done to evaluate how well our approach could be adapted to other datasets. The results showed that although the performance metrics (AUC and BAC) were slightly lower for the EMC dataset compared to the TUH dataset, the models still displayed similar results. This discovery implies that the features we identified are pertinent and efficient across different patient groups and recording settings.The results of this research have some implications. Being able to classify EEG signals from patients with epilepsy compared to those without can still be challenging. In addition to this, the adapting these techniques across datasets highlights the potential for increased use. Additional data collection is still important to improve the EEG analysis on the EMC dataset.

Although this paper showed different outcomes, there are limitations to take into account. After applying the model on the EMC dataset with Thangavel et al. method, it didn't reach the same level as on the TUH dataset. This difference might come from variations in recording environments, patient characteristics, or the presence of distinct epileptic patterns in the EMC dataset. Future studies could explore alternative feature extraction methods, data augmentation techniques, and different machine learning algorithms to further improve classification accuracy [76].

7.2 Future Research Directions

7.2.1 Expanding and Merging Datasets

Expanding the datasets to include a more varied patient population and a broader range of epilepsy types that occur in the future could provide more validation of the proposed methods. Merging both datasets could also be a new solution to increase the sample size that can help with training the models from scratch in addition to using pre-trained models. In addition to this, based on past experience in an internship, re-training the models after finding out the mis-classified features can also be a new approach in epilepsy diagnosis as this can for sure improve the accuracy of the model by removing the bad features that are used for training.

7.2.2 Hybrid Models and Deep Learning Approaches

Using a hybrid model approach such as combining XGBoost with an LSTM model could improve classification accuracy. Deep learning approaches could have a significant impact by using the known feature vectors implemented in this thesis. The integration of STFT or wavelet transforms with 1D or 2D-CNNs depending on the complexity is recommended for future work as these methods have proven to be successful in other studies. However, it is important to note that not all studies use datasets without IEDs which can have an impact on classification performance as highlighted in this study as there is a significant performance drop between dataset with and without IEDs. Future advanced deep learning research could focus on 1D CNNs on time-domain signals, 2D CNNs on STFT-domain signals or transformers (impact in image classification tasks) [91, 92].

7.2.3 Statistical Analysis and alternatives

Further research should also involve statistical analysis of all extracted features to examine the main effects in epileptic EEGs when using the ensemble method. This could give more information in the feature selection process identifying the most significant features and improving model performance. Looking into alternatives to wavelet and Stockwell transforms such as multi-frequency band functional connectivity analysis could provide more insights into the brain's functional networks with additional feature extractions related to mean or median power / energy [12, 33, 90]. This could give a better understanding of the EEG data and improve classification accuracy. Besides this, looking into how the artefact removal algorithm influences the statistical properties of the EEG signal should be looked into to see if there are better techniques other than the one used in this thesis.

7.2.4 Analysis of External Factors

The analysis of the effects of external factors such as hyperventilation, photic stimulation, and sleep deprivation on EEG patterns, as shown in the annotations in the EMC dataset in Figure A.12, is another important direction. These factors are known to influence EEG readings, and their inclusion or exclusion could improve model robustness

and accuracy. In this thesis, the photic stimulation part is removed from the dataset due to too many spikes as mentioned in the Background chapter.

7.2.5 Including Patient-Specific Information

Including patient-specific information such as age, gender, and medical history in the epilepsy diagnosis models could provide more detailed differences between the 2 classes improving the model's ability to distinguish between epilepsy types and other neurological conditions especially when expanding the dataset.

7.2.6 Segmentation and Transfer Learning for Data Augmentation

Provided this limited size of the current dataset, training deep learning models from scratch may not be feasible or reliable i.e. there could be overfitting when trying the STFT and a 2D CNN. Using segmentation techniques to generate an increased number of training samples and applying transfer learning from pre-trained models could help overcome these data limitations and improve model performance.

7.2.7 Other feature extraction methods

There are many other features other than the ones used in this thesis that can influence the outcomes of the model. In terms of connectivity features, coherence can be a new feature that can be looked into as mentioned in some papers [93]. So, the cross-correlation and phase locking value from before can be compared to this feature set. Different combiners can be added and looked into in the future besides the current 5 combiners.

Conclusion

The application of transfer learning using the VGG16 model has proven to be highly effective for the diagnosis of epilepsy using EEG data from the EMC dataset. The model that performed best is the VGG16 + XGBoost + LOSO CV which shows that it can be a strong candidate for future automated epilepsy detection systems. Other than this, Thangavel et al.'s approach on both datasets prove that there are significant results when classifying EEGs without IEDs which was one of the main research challenges of this thesis.

The results suggest that integrating deep learning-based feature extraction with traditional machine learning classifiers such as XGBoost can improve the accuracy and reliability of automated epilepsy diagnosis systems a lot. Future work could also explore the integration of these models into clinical workflows which could have a significant impact on patient outcomes.

In conclusion, this study successfully developed, validated, and compared different machine learning models for the classification of EEG signals in epileptic and healthy patients. The results show the efficiency of the proposed methods and their potential for clinical application. Future research should aim to address the limitations which was mentioned and continue to refine and expand the capabilities of these models for epilepsy diagnosis. With not many papers focusing on EEGs without IEDs, many techniques can be tested to check whether the classification results can improve.

Supplementary Figures & Statistical analyses

A

A.1 Pre-processing

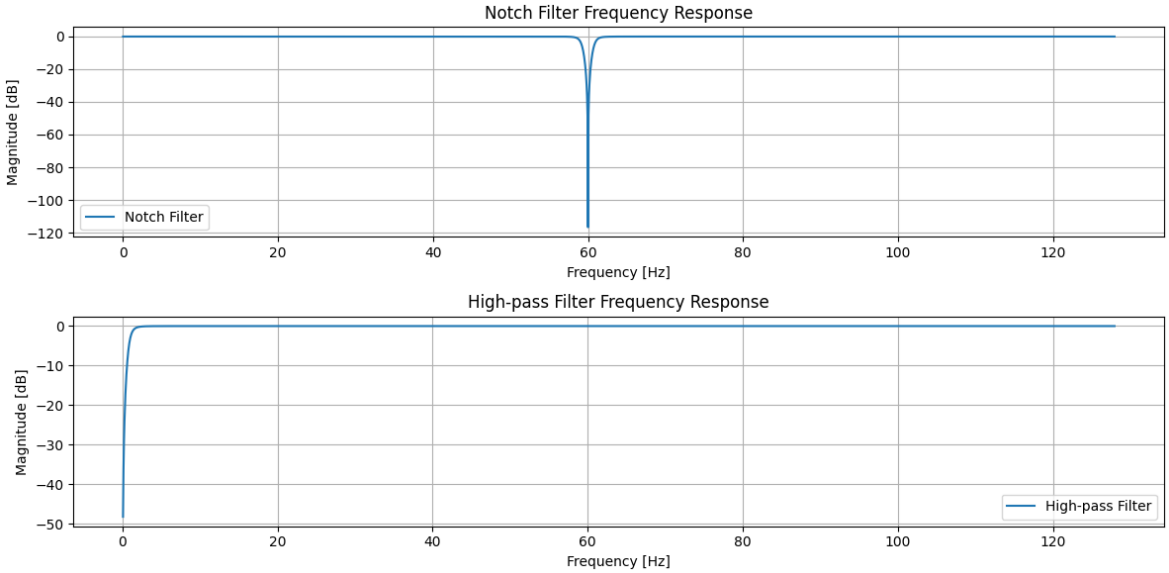


Figure A.1: Frequency responses of the notch & high pass filter.

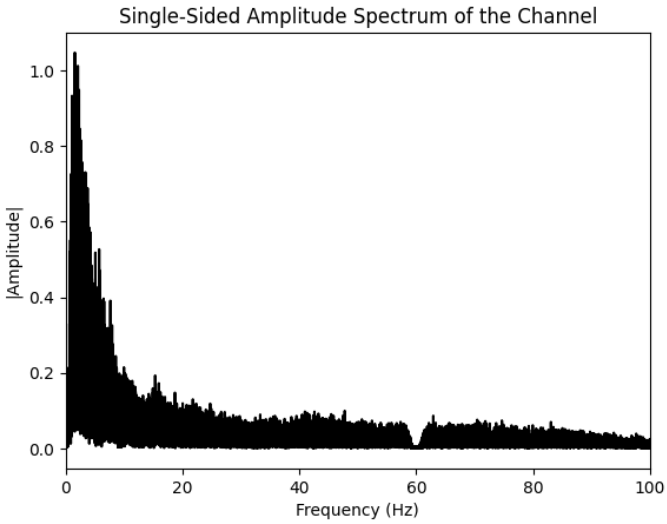


Figure A.2: Single-sided amplitude spectrum of channel F3.

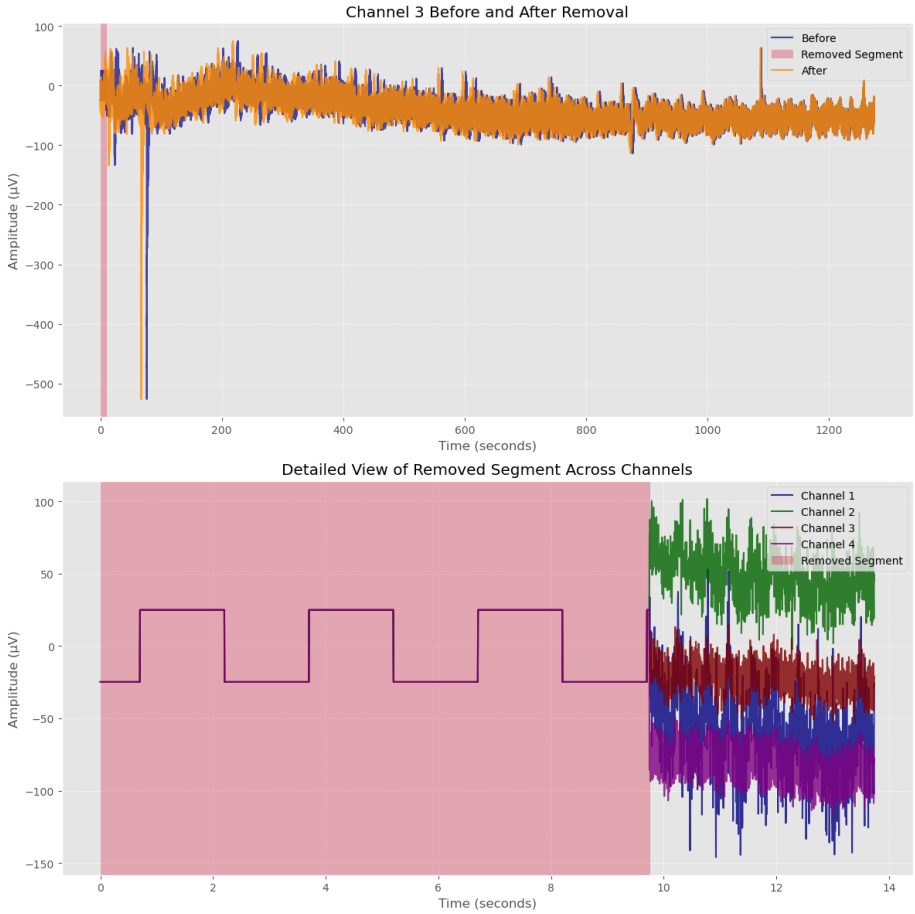


Figure A.3: Pre-processing of TUH dataset showing raw EEG data noise & artifact removals.

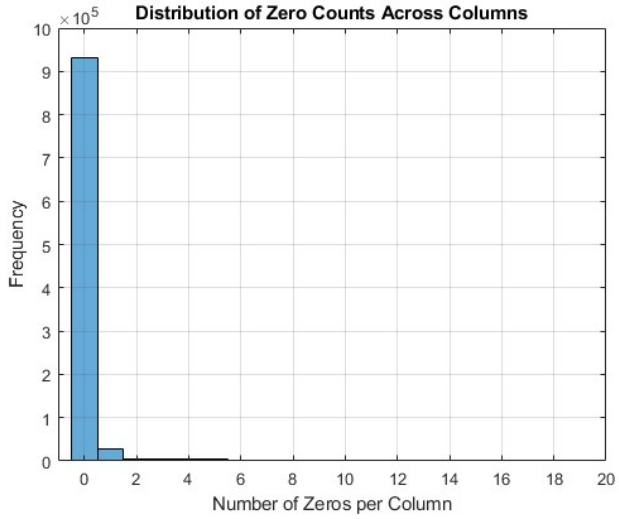


Figure A.4: Distribution of zero counts across all channels before the removing segments stage.

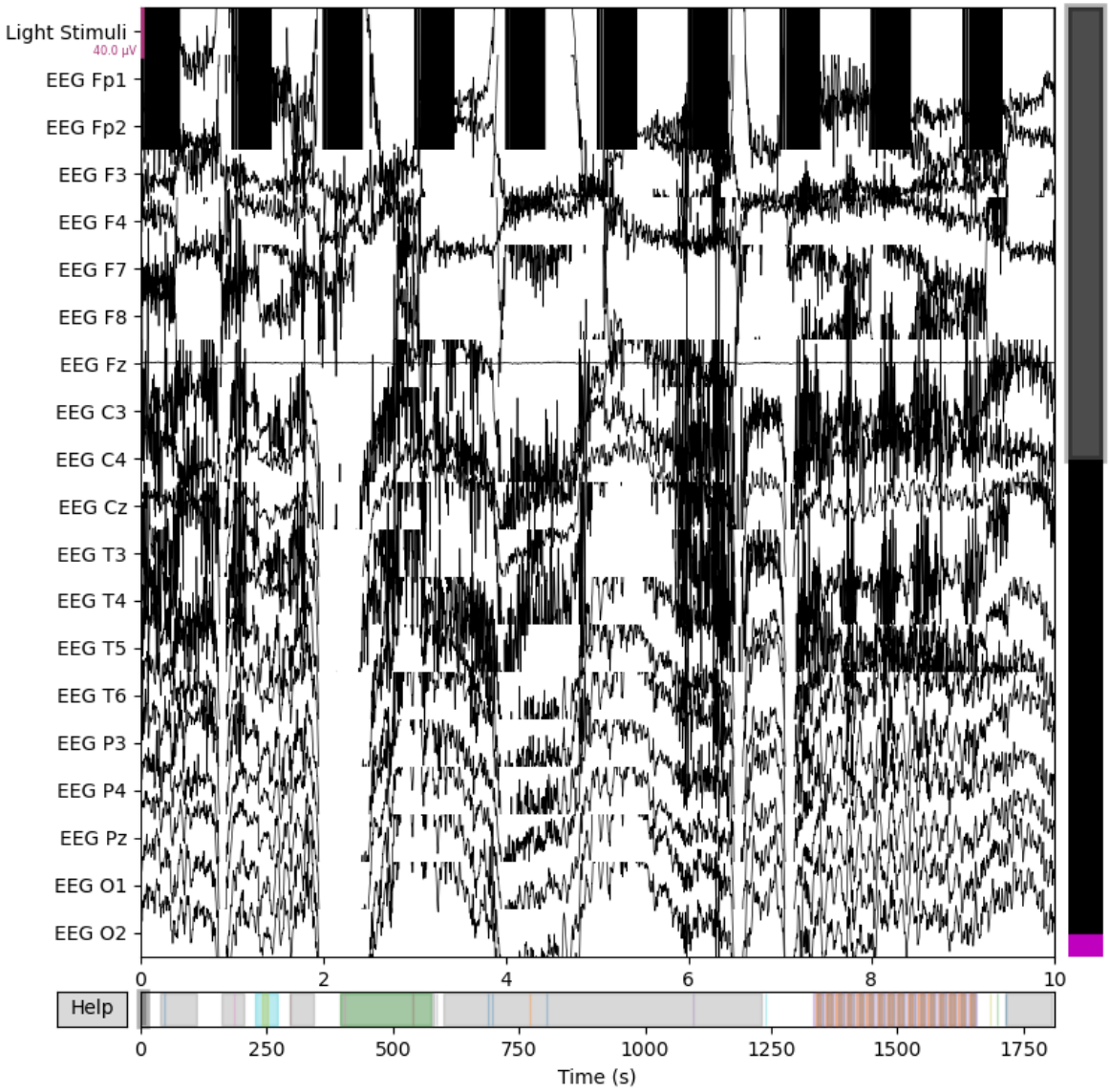


Figure A.5: Plot of the EEG signals from different channels of the EMC dataset.

A.2 Feature extraction analysis

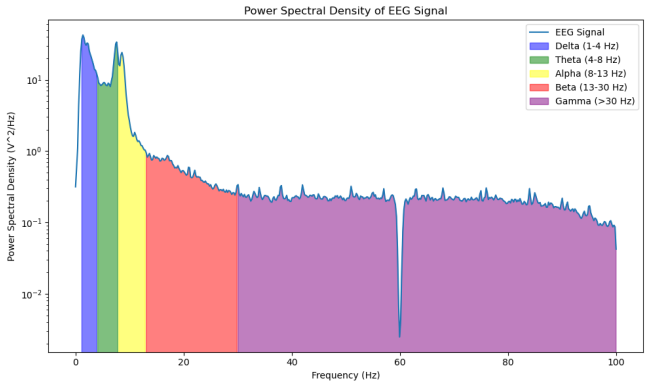


Figure A.6: Spectral bands of EEG signals showing the relative power of delta, theta, alpha, beta, and gamma bands.

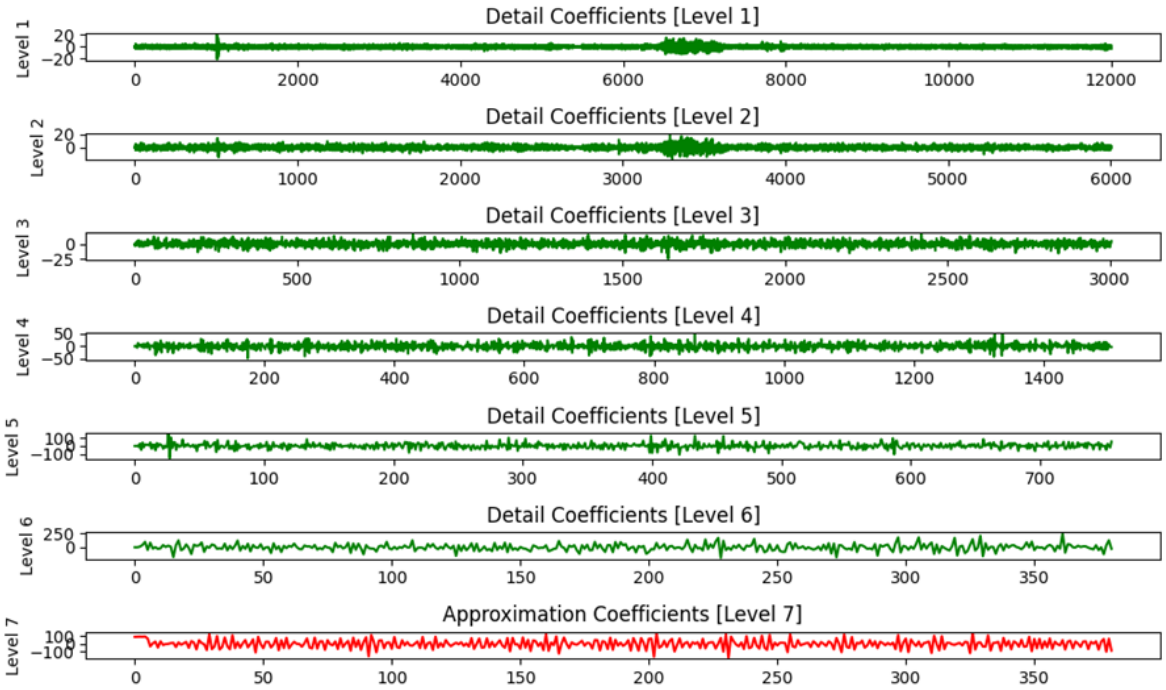


Figure A.7: Approximate and detail coefficients of a sample EEG signal segment taken from a epileptic patient.

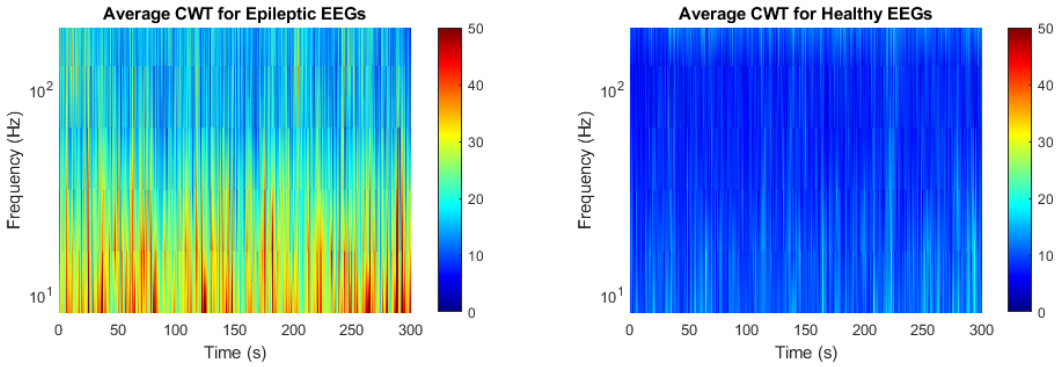


Figure A.8: Average CWT coefficients for epileptic and healthy EEGs using Morlet (morl) wavelet. The plot shows the time on the x-axis and frequency on the y-axis with the color bar indicating the squared magnitude (power).

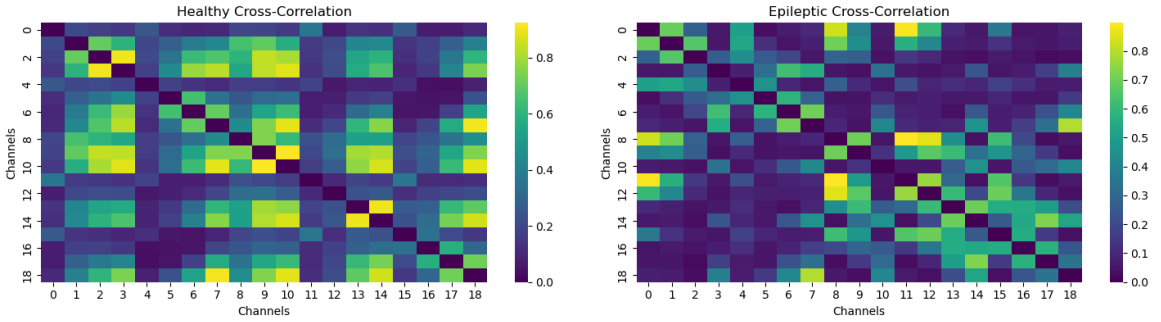


Figure A.9: Heatmap of CC showing the strength of connections between different EEG channels.

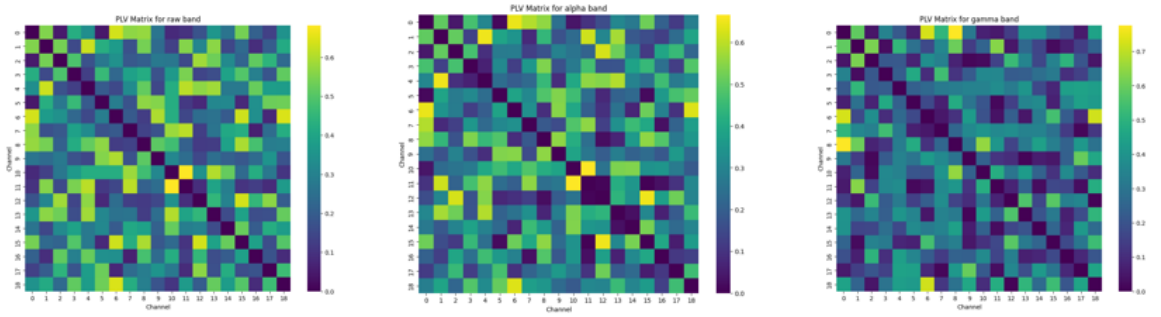


Figure A.10: CPLV connectivity matrices for different bands.

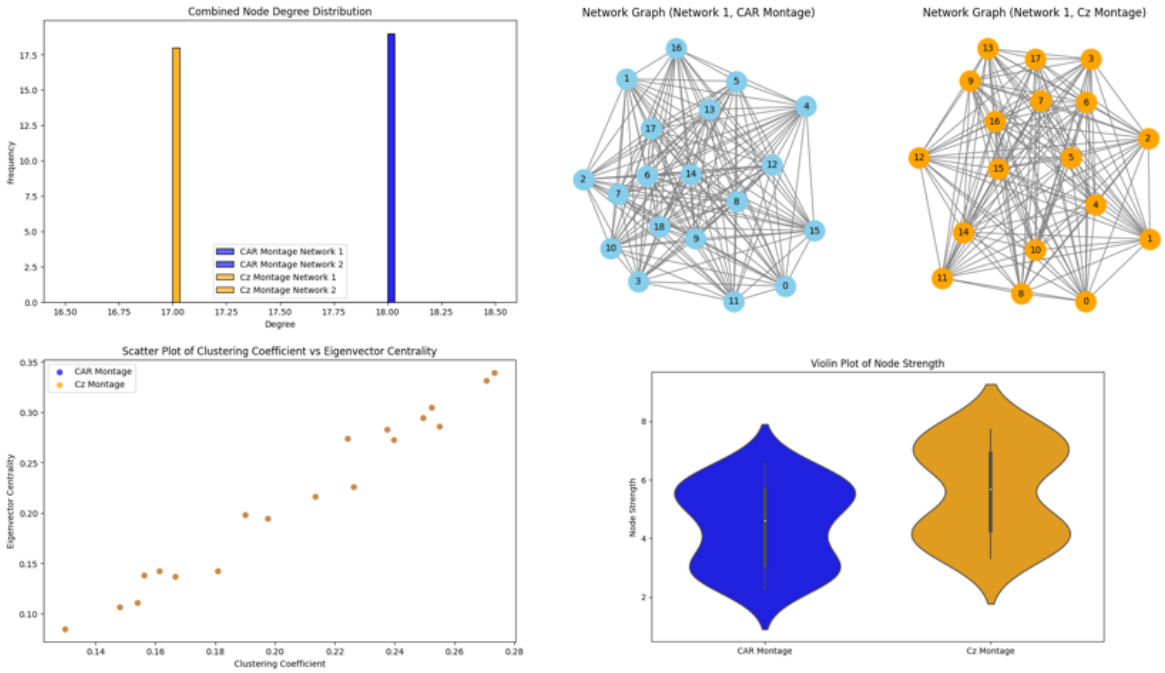


Figure A.11: Graph metrics visualisation using the networkx toolbox [94].

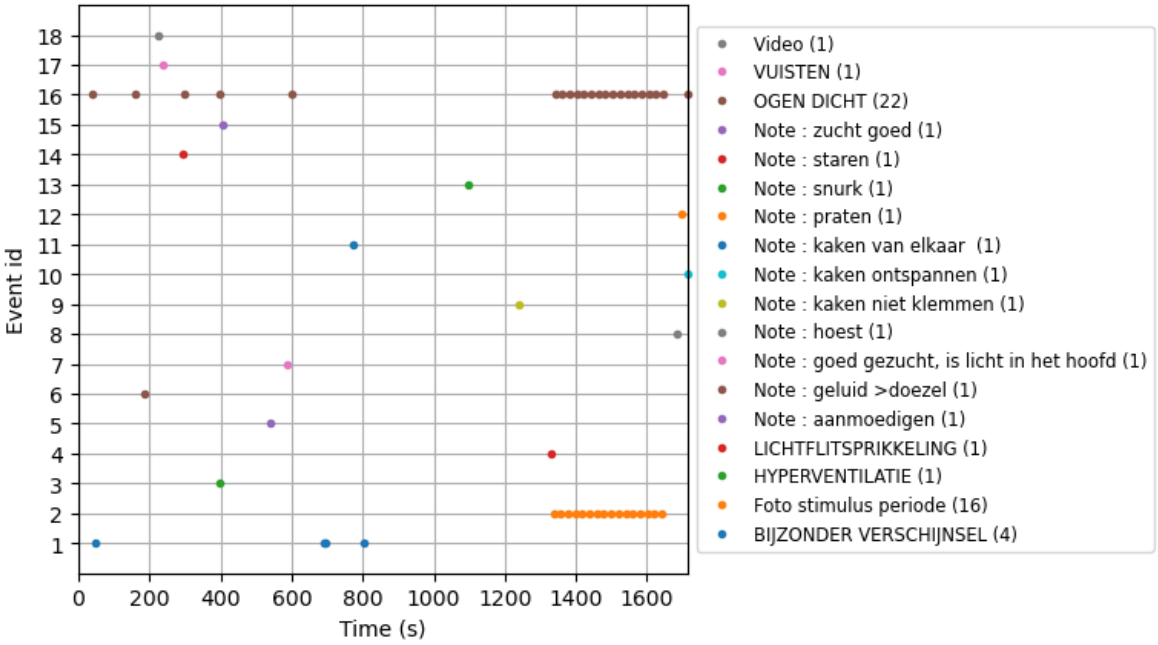


Figure A.12: EMC dataset annotations.

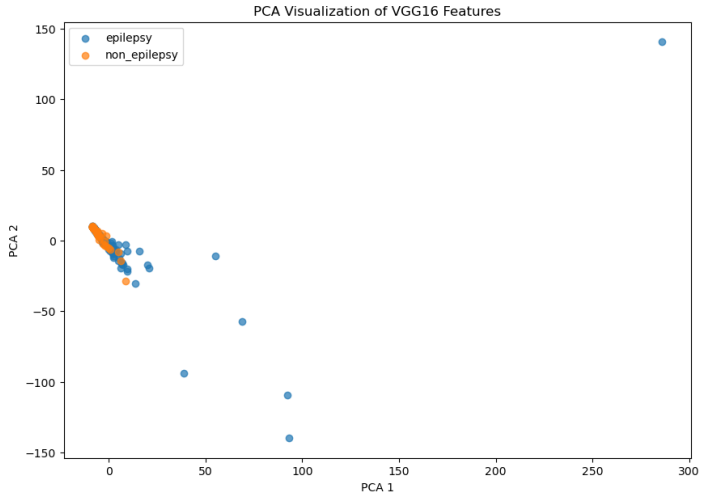


Figure A.13: PCA Visualization of VGG16 Features (TUH dataset).

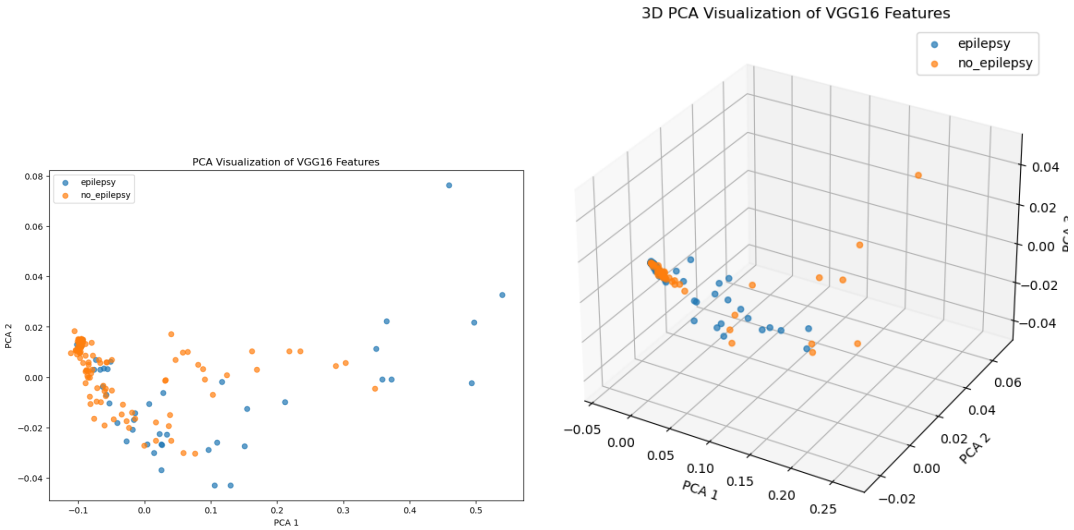


Figure A.14: PCA Visualizations 2 components (left) & 3 components (right) of VGG16 Features (EMC dataset).

A.3 Statistical analysis

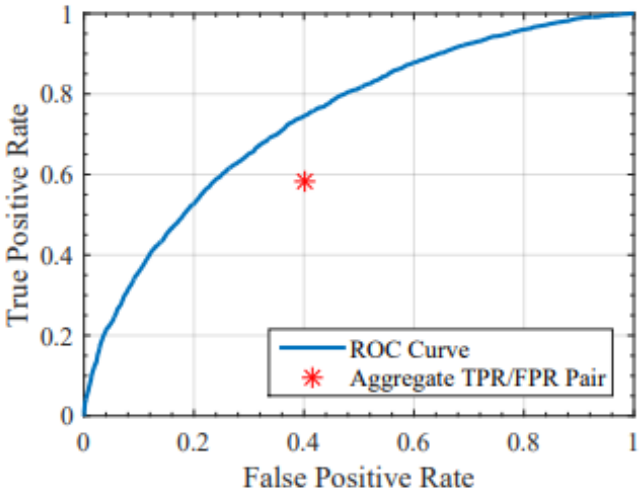


Figure A.15: A typical ROC curve [95].

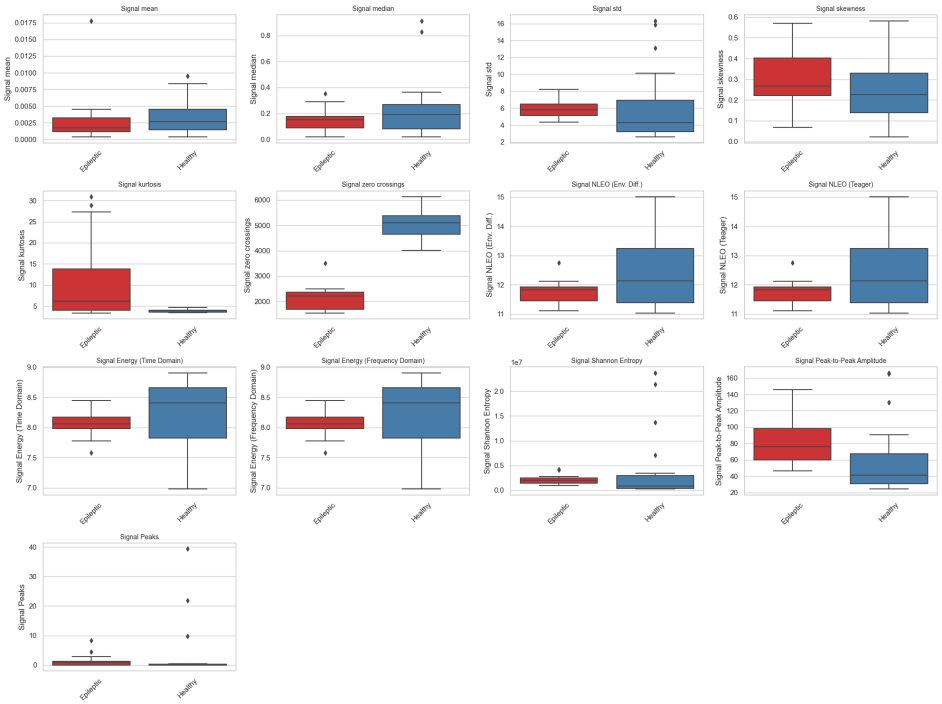


Figure A.16: Box-plots of each UTM feature comparing Epileptic vs Healthy patients [N = 10] for TUH dataset.

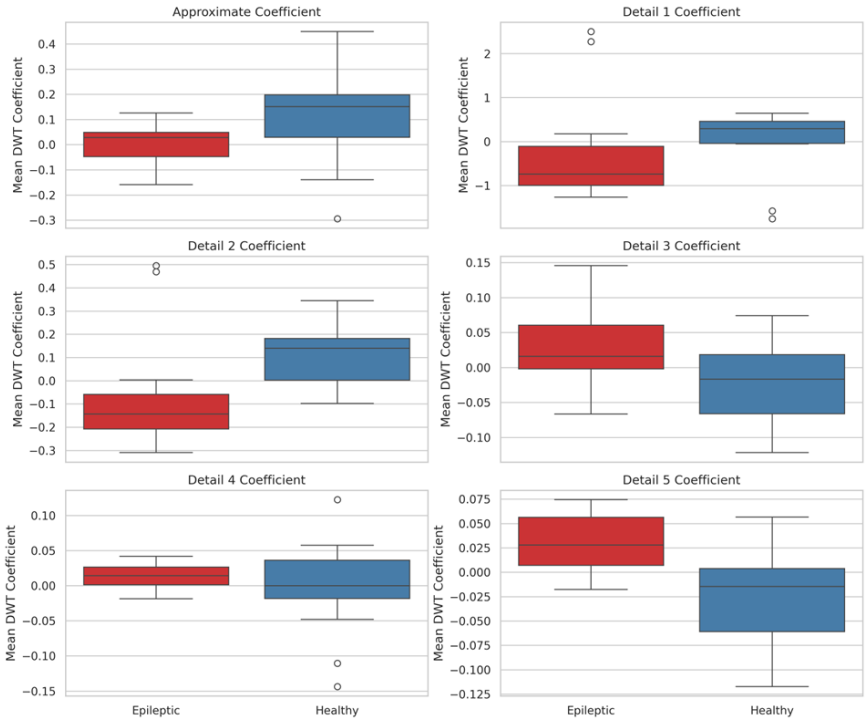


Figure A.17: Box-plots of DWT features comparing Epileptic vs Healthy patients [N = 10] for EMC dataset.

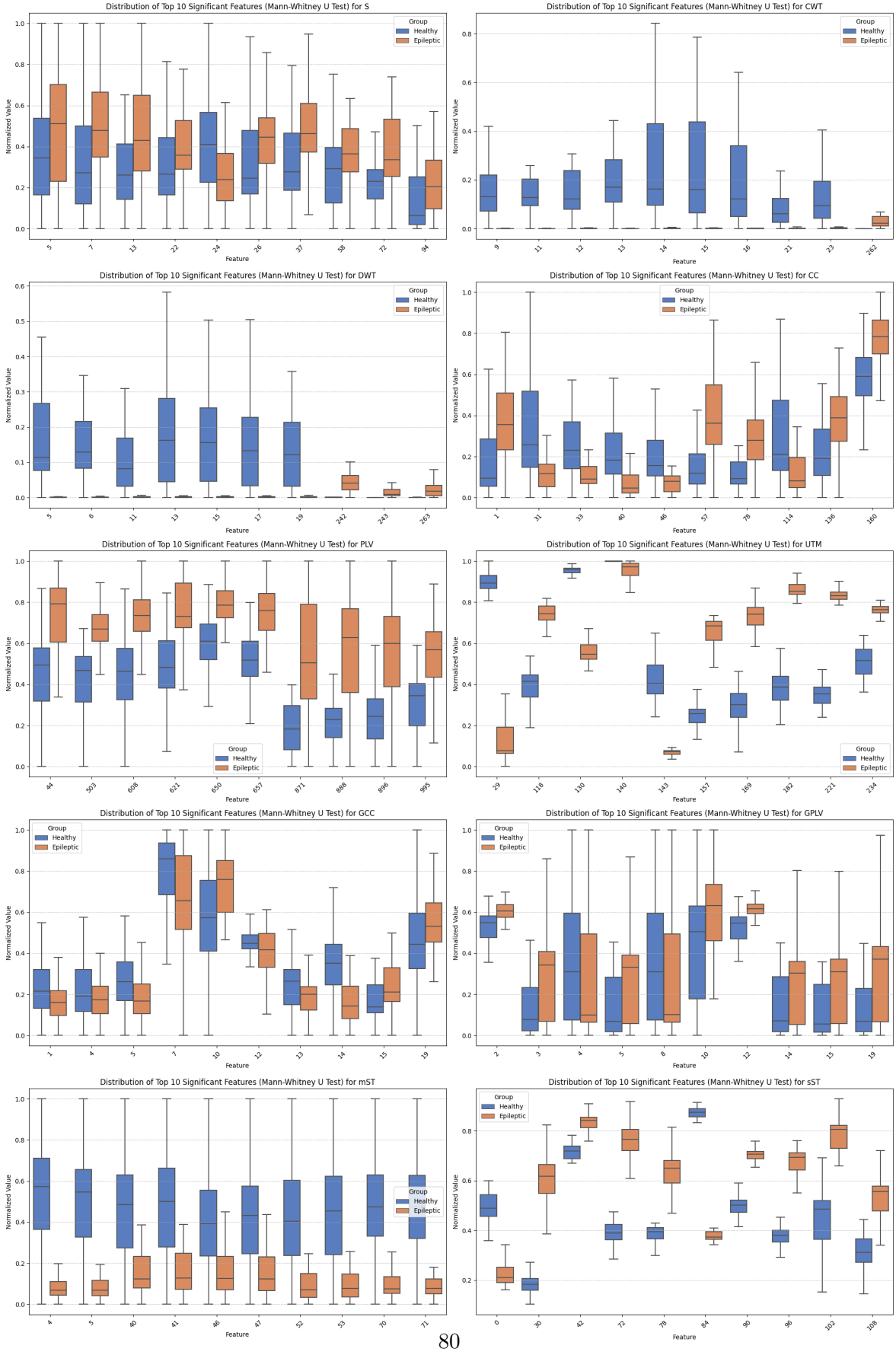


Figure A.18: Distribution of Top 10 Significant Features (Mann-Whitney U Test) for Each Feature Set (TUH).

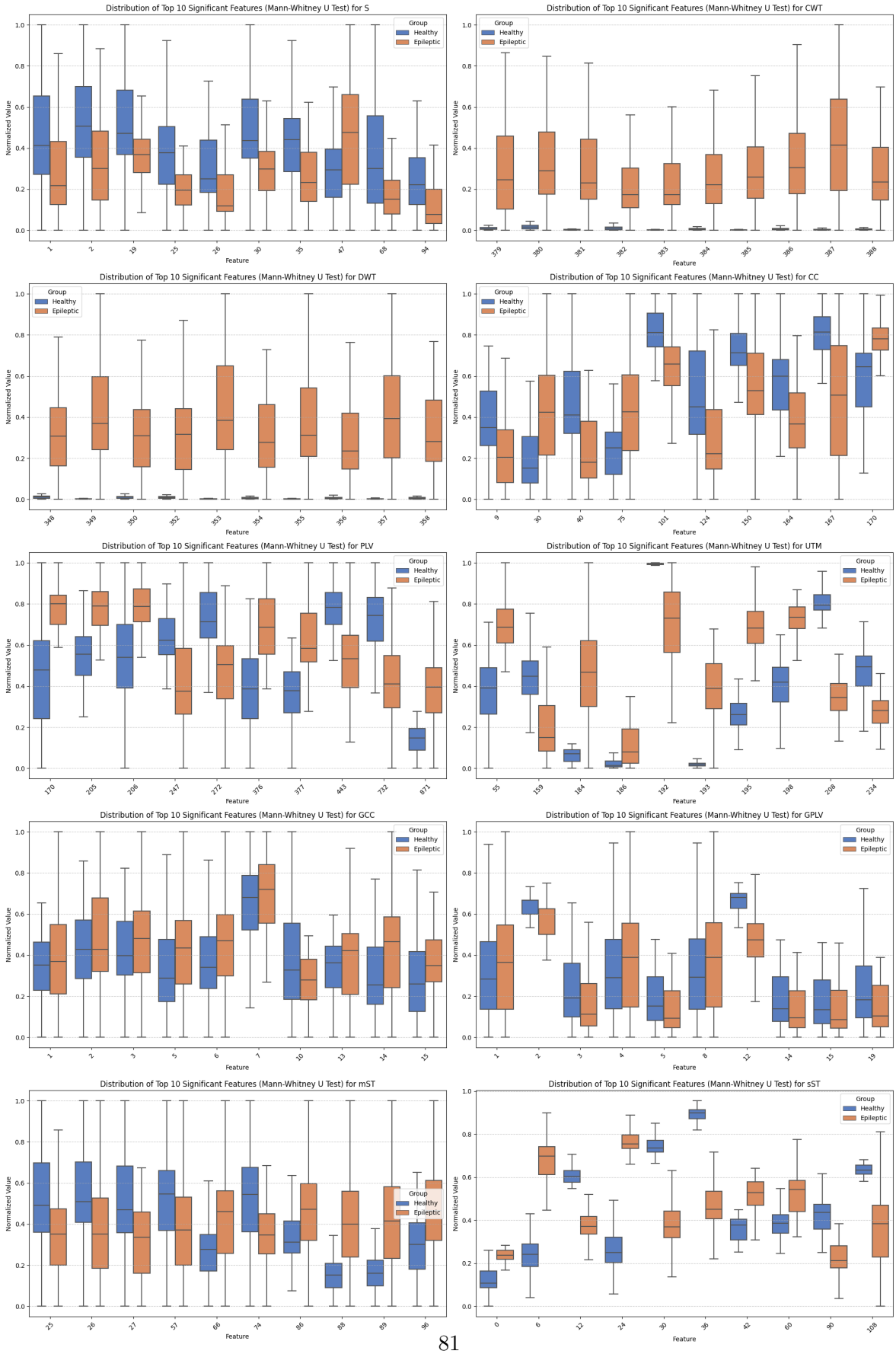


Figure A.19: Distribution of Top 10 Significant Features (Mann-Whitney U Test) for Each Feature Set(EMC).

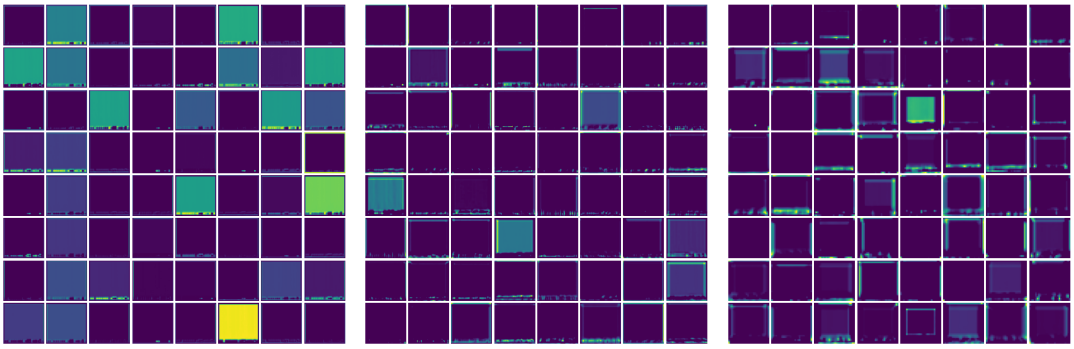


Figure A.20: Feature maps in first, third and fourth convolutional blocks for epileptic patient (sub-0001) EMC dataset.

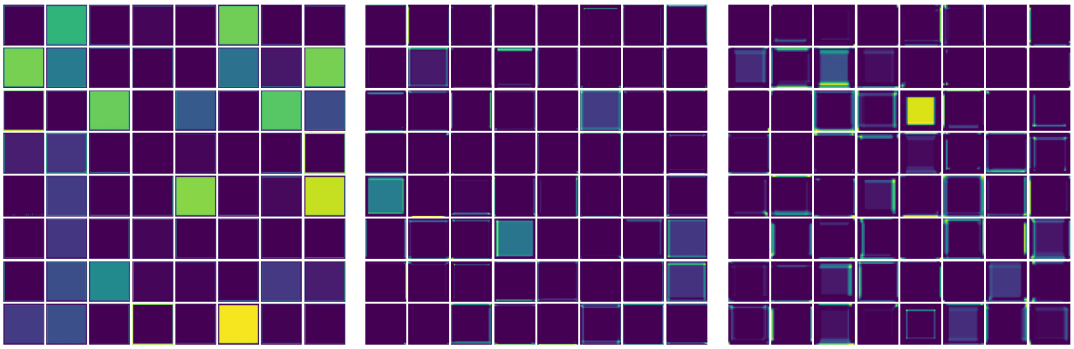


Figure A.21: Feature maps in first, third and fourth convolutional blocks for non-epileptic patient (sub-0001) EMC dataset.

Flowcharts

B

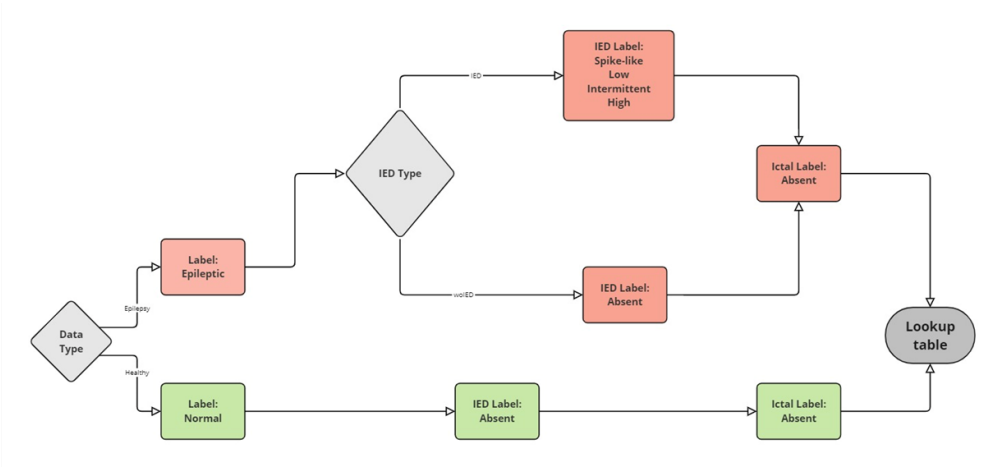


Figure B.1: Labelling & lookup table for TUH dataset configuration.

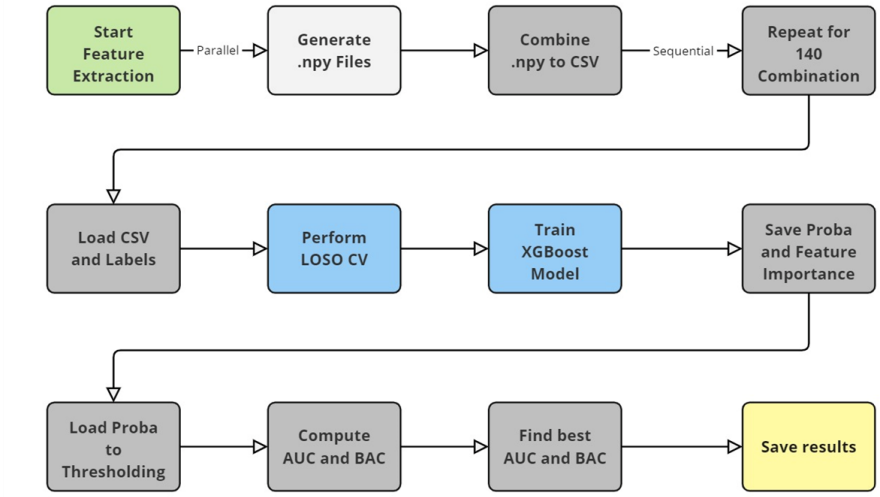


Figure B.2: Overall process with multiprocessing using Delftblue cluster.

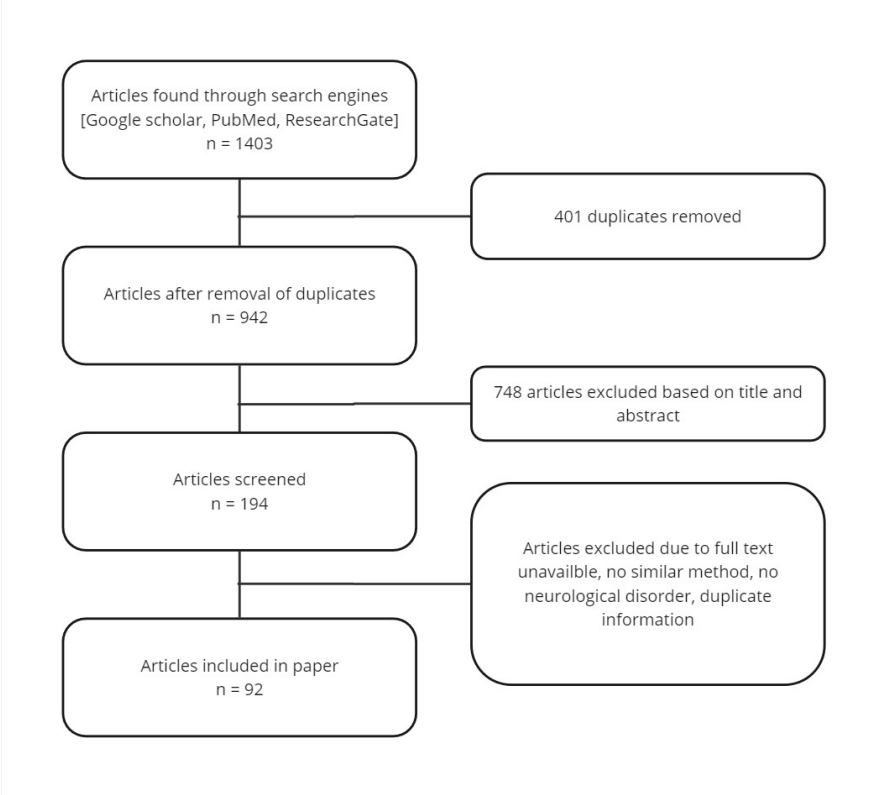


Figure B.3: Flowchart describing the different steps of article collection.

Comparison b/w MATLAB and Python algorithms

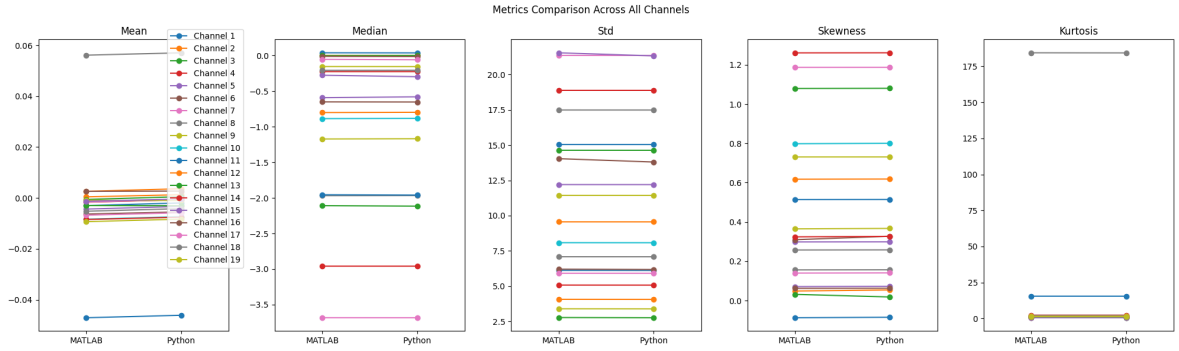


Figure C.1: Pre-processing channel metrics results b/w MATLAB and Python.

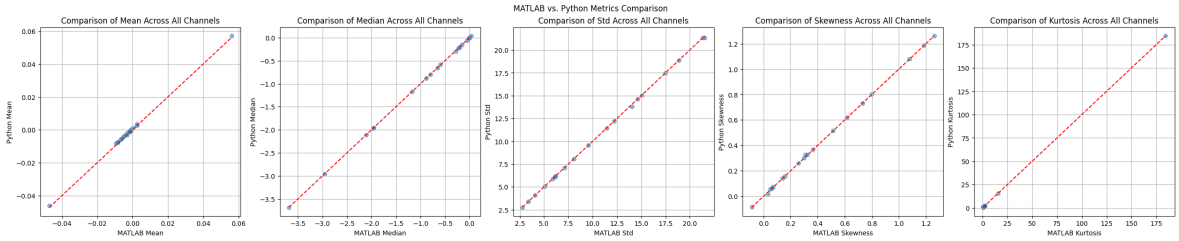


Figure C.2: Pre-processing Correlation line plot b/w MATLAB and Python.

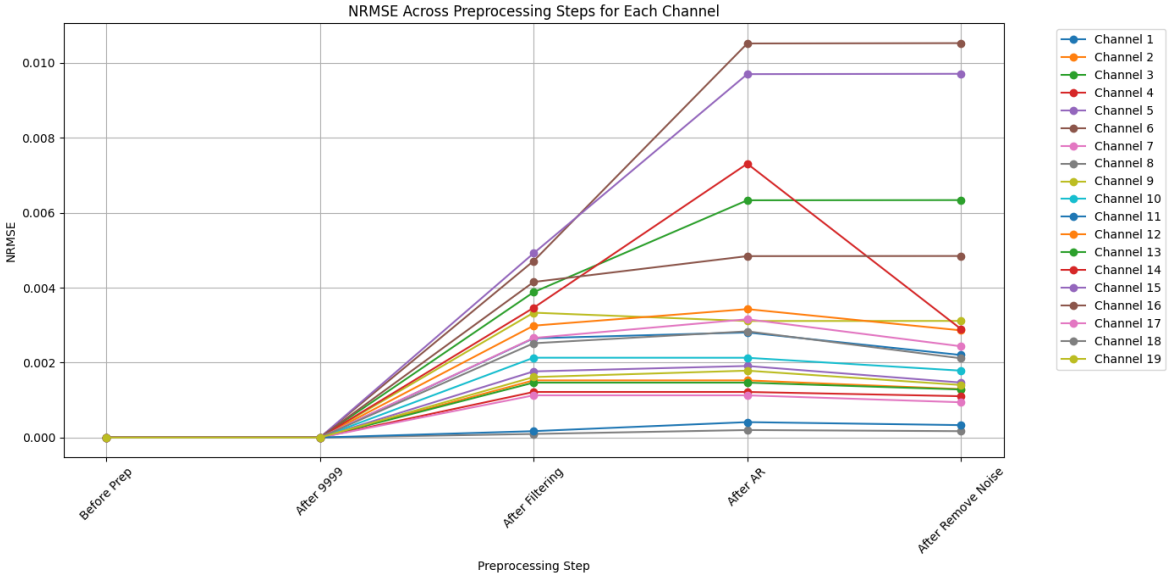


Figure C.3: Pre-processing NRMSE results b/w MATLAB and Python.

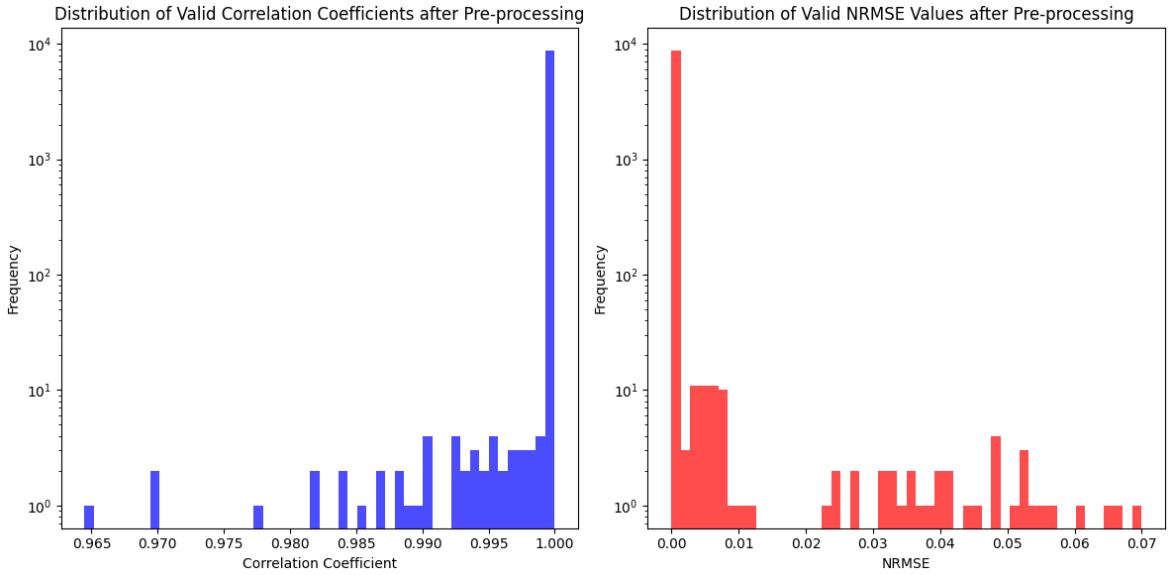


Figure C.4: Pre-processing Correlation & NRMSE results b/w MATLAB and Python.

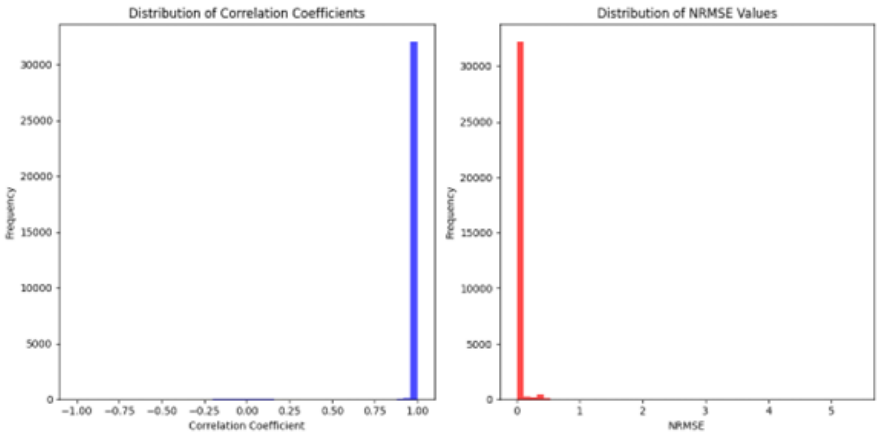


Figure C.5: S Correlation results & NRMSE b/w MATLAB and Python.

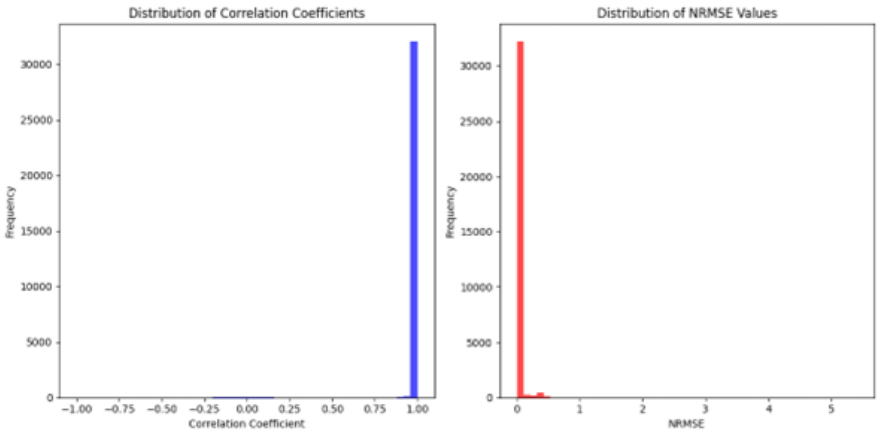


Figure C.6: CC Correlation results & NRMSE b/w MATLAB and Python.

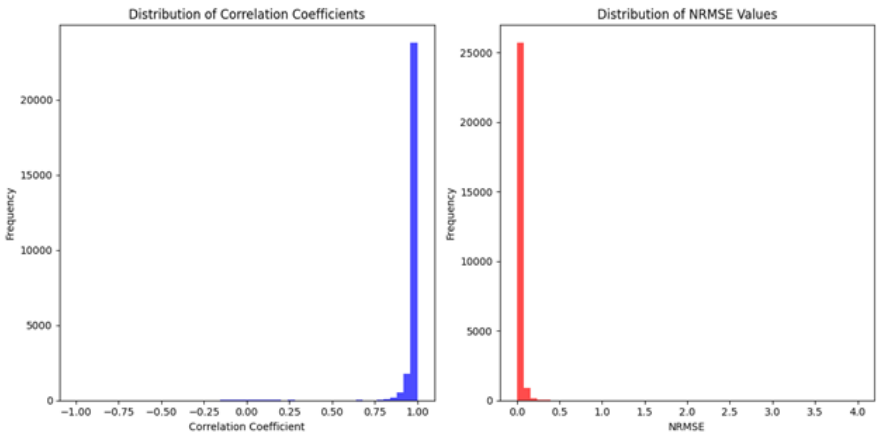


Figure C.7: CPLV Correlation results & NRMSE b/w MATLAB and Python.

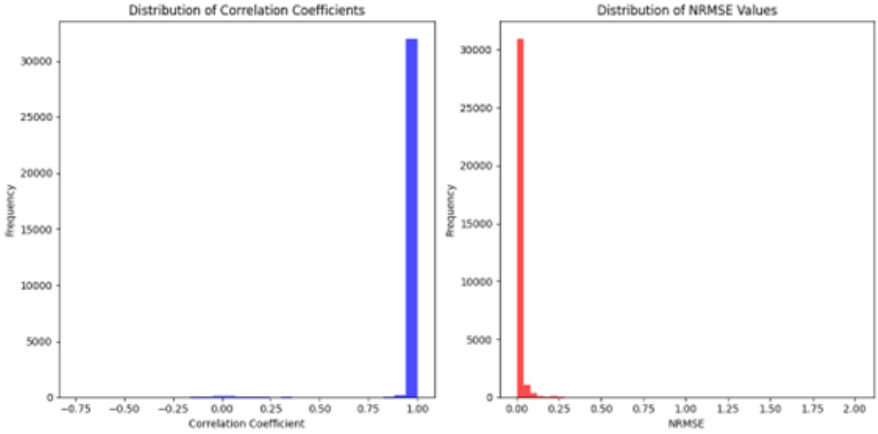


Figure C.8: DWT Correlation results & NRMSE b/w MATLAB and Python.

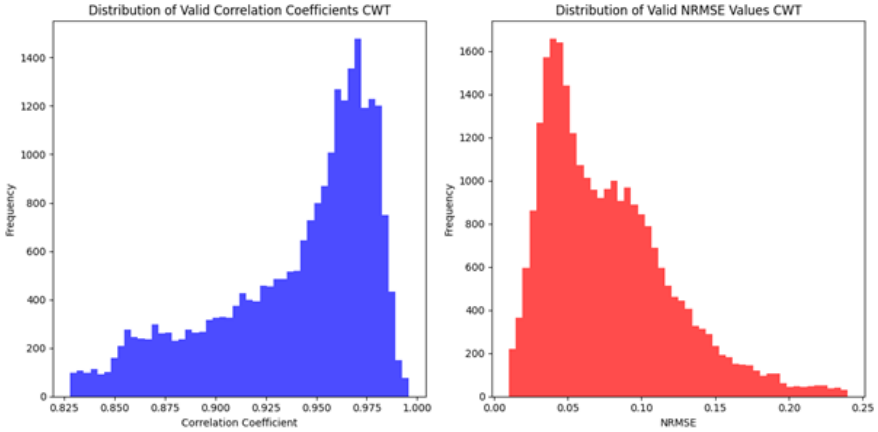


Figure C.9: CWT Correlation results & NRMSE b/w MATLAB and Python.

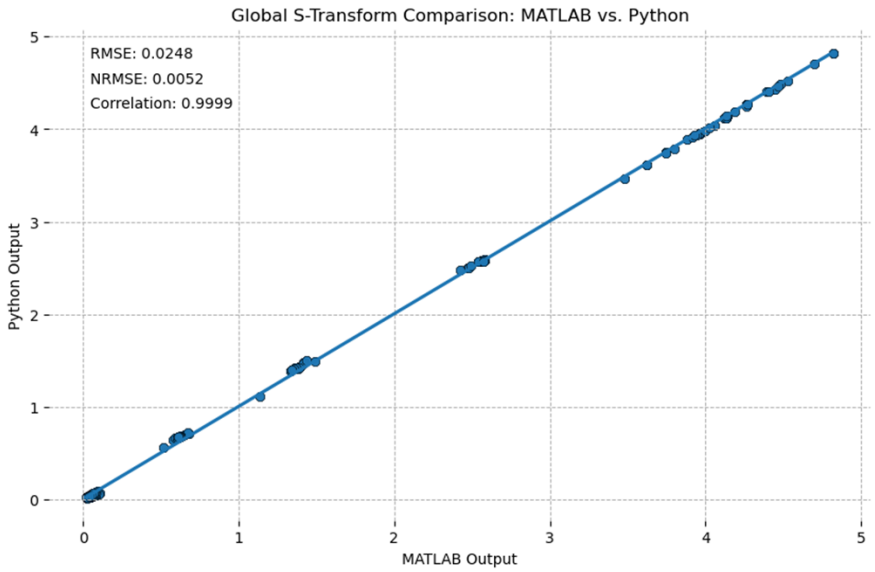


Figure C.10: Global S-transform MATLAB vs Python.

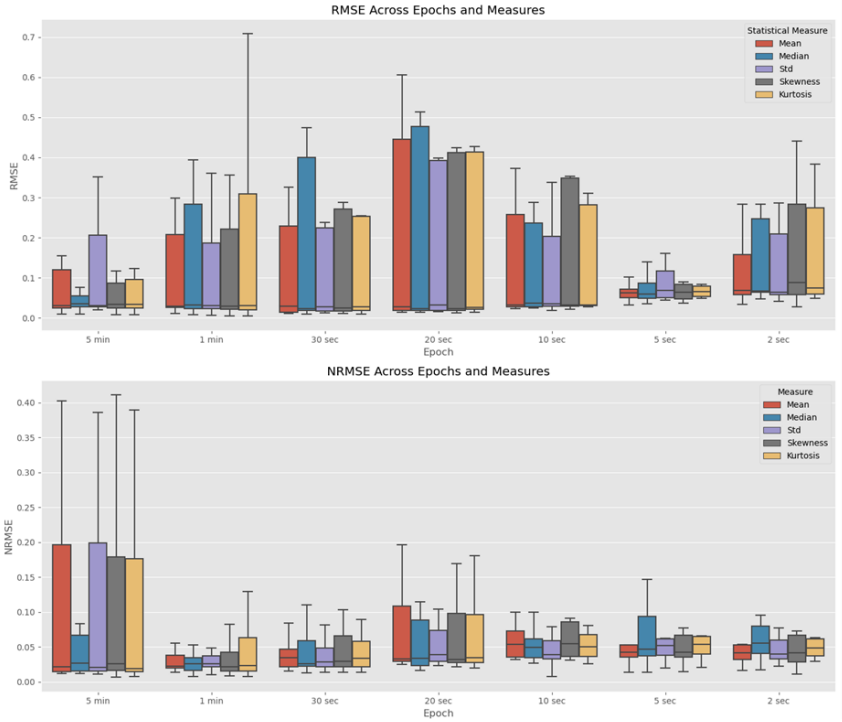


Figure C.11: RMSE and NRMSE stockwell MATLAB vs Python.

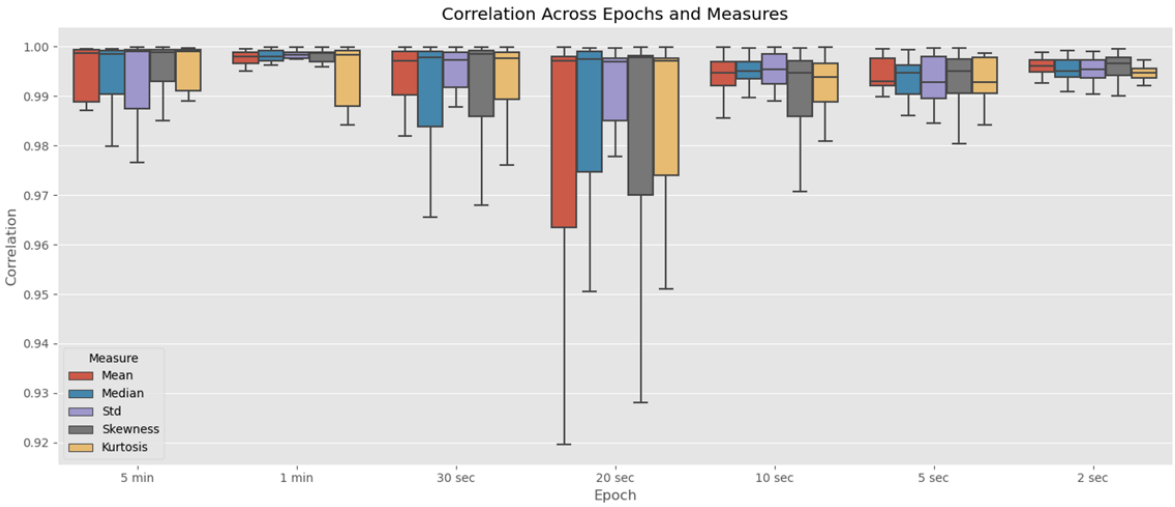


Figure C.12: Correlation analysis stockwell MATLAB vs Python.

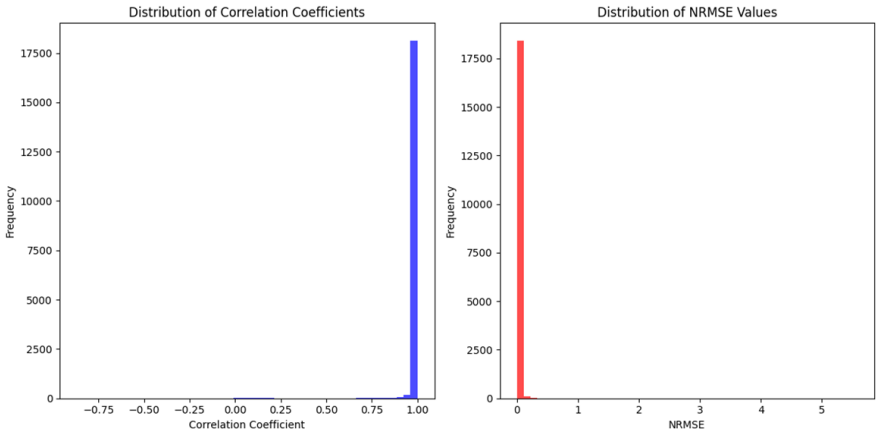


Figure C.13: Graph metrics Correlation results & NRMSE b/w MATLAB and Python.

C.1 Tabular results

Table C.1: Individual features comparison of the TUH dataset with IEDs.

Feature	Thangavel Paper		w/o Age + Vigilance		with Age + Vigilance	
	AUC	**BAC**	**AUC**	**BAC**	**AUC**	**BAC**
S	0.867	74.167	0.8633	77.33	0.87	78.33
CC	0.8	70.167	0.798	70.33	0.8	70.66
CPLV	0.78	66.833	0.7848	66.16	0.79	67.166
ST SR	0.85	78.33	0.842	77.66	0.848	79.33
ST P	0.69	60	0.68	59.66	0.7	61
DWT	0.87	80.33	0.86	79.33	0.8656	79.66
CWT	0.85	75	0.8464	74.66	0.856	75.33
Cnetwork	0.72	68	0.71	66	0.7233	68.33
Pnetwork	0.77	69.33	0.78	69.66	0.79	71
UTM	0.86	83.33	0.85	83	0.856	83.2

Algorithms re-implemented after bug-fixes in Python library

D

D.1 Pre-processing

Algorithm 5 Buffer Data into Segments of 1 Second

Input: *data* (signal), *sample_rate* (samples per second)
Output: *padded_data* (buffered data in segments)
$buffer\_len \leftarrow sample\_rate$
$num\_segments \leftarrow \lceil \frac{length(data)}{buffer\_len} \rceil$
$padded\_len \leftarrow num\_segments \times buffer\_len$
$padded\_data \leftarrow zeros(padded\_len)$
$padded\_data[: length(data)] \leftarrow data$
return $padded\_data.reshape(num\_segments, buffer\_len)$

D.2 Graph metrics

Algorithm 6 Eigenvector Centrality for Undirected Networks

Input: *network* (adjacency matrix)
Output: v (eigenvector centrality)
$n \leftarrow$ length of *network*
if $n < 1000$ **then**
    $vals, vecs \leftarrow \text{eig}(network)$
else
    $vals, vecs \leftarrow$ eigs(csr_matrix($network$), k=1, which='LM')
end if
$idx \leftarrow \text{argmax}(\text{real}(vals))$
$ec \leftarrow \text{abs}(vecs[:, idx])$
$v \leftarrow ec.reshape(len(ec), 1)$
return v

Algorithm 7 Autofix Matrix Adjustments

Input: W (matrix)
Output: W (modified matrix)
Clear diagonal elements of W
$\text{diag}(W) \leftarrow 0$
Remove Infs and NaNs from W
$W[\text{isinf}(W) \text{ or } \text{isnan}(W)] \leftarrow 0$
Ensure exact binariness of W
$U \leftarrow \text{unique}(W)$
if $\text{len}(U) > 1$ **then**
    $idx\_0 \leftarrow |W| < 10^{-10}$
    $idx\_1 \leftarrow |W - 1| < 10^{-10}$
    if all(idx_0 or idx_1) **then**
        $W[idx\_0] \leftarrow 0$
        $W[idx\_1] \leftarrow 1$
    end if
end if
Ensure exact symmetry of W
if not array_equal(W, W^T) **then**
    if $\max(|W - W^T|) < 10^{-10}$ **then**
        $W \leftarrow \frac{W+W^T}{2}$
    end if
end if
return W

Algorithm 8 Custom Local Assortativity

$n \leftarrow$ length of W
Set diagonal of W to 0
$r_{pos} \leftarrow$ assortativity of $W \times (W > 0)$
$r_{neg} \leftarrow$ assortativity of $-W \times (W < 0)$
$str_{pos}, str_{neg} \leftarrow$ strengths of W
Initialize loc_assort_pos, loc_assort_neg as NaN arrays
for $curr\_node = 0$ to $n - 1$ **do**
    $j_{pos} \leftarrow$ indices of $W[curr\_node, :] > 0$
    if $str_{pos}[curr\_node] \neq 0$ **then**
        $loc\_assort\_pos[curr\_node] \leftarrow \sum |str_{pos}[j_{pos}] - str_{pos}[curr\_node]| / str_{pos}[curr\_node]$
    end if
    $j_{neg} \leftarrow$ indices of $W[curr\_node, :] < 0$
    if $str_{neg}[curr\_node] \neq 0$ **then**
        $loc\_assort\_neg[curr\_node] \leftarrow \sum |str_{neg}[j_{neg}] - str_{neg}[curr\_node]| / str_{neg}[curr\_node]$
    end if
end for
if sum of loc_assort_pos is not NaN **then**
    $loc\_assort\_pos \leftarrow ((r_{pos} + 1)/n) - (loc\_assort\_pos / \sum(loc\_assort\_pos))$
end if
if sum of loc_assort_neg is not NaN **then**
    $loc\_assort\_neg \leftarrow ((r_{neg} + 1)/n) - (loc\_assort\_neg / \sum(loc\_assort\_neg))$
end if
return $loc\_assort\_pos, loc\_assort\_neg$

Algorithm 9 Modularity Calculation for Weighted Networks

Input: A (connection matrix), γ (resolution)
Output: Ci (community indices), Q (modularity)
Initialize N, K, m, B, Ci, cn, U
while U not empty **do**
    Compute eigenvalues and eigenvectors of B
    Identify maximum eigenvalue and corresponding eigenvector
    Compute S and modularity contribution q
    if $q > 0$ **then**
        Fine-tune S by iteratively adjusting signs
        if no split possible **then** Remove current community from U
        elseDivide community and update Ci and U
        end if
    elseRemove current community from U
    end if
    Update B for next community
end while
Compute final modularity Q based on Ci

D.3 CWT algorithm with modified checks

Algorithm 10 Feature Extraction Using CWT

Input: EEG data *data*, Sampling frequency *Fs*, Segment length *sec*, Wavelet type *WAVELET_TYPE*
Initialize: Number of segments *seg_num*, scales *scales*, output matrix *out*
Checks: Check frequencies greater than 2 Hz and segment lengths
if *seg_num* == 0 **then**
    out ← CWT of the entire data
else
    Adjust scales dynamically to ensure sufficient scales
    for *each segment* and *each channel* **do**
        Apply CWT on the segment
        Extract mean and standard deviation of squared coefficients
        Store features in *out*
    end for
end if
Output: Feature matrix *out*

Bibliography

[1] World Health Organization, "Epilepsy fact sheet." Available: https://www.who.int/news-room/fact-sheets/detail/epilepsy, Feb. 2024. Accessed: 2024-04-06.

[2] Mayo Clinic, "Eeg (electroencephalogram)." Available: https://www.mayoclinic.org/tests-procedures/eeg/about/pac-20393875, 2024. Accessed: 2024-07-06.

[3] Nederlandse Vereniging voor Neurologie, "Zorgevaluatie en kennisvragen." Available: https://www.neurologie.nl/wetenschap/zorgevaluatie/kennisvragen/, 2024. Accessed: 2024-07-06.

[4] B. Frauscher and J. Gotman, "Sleep, oscillations, interictal discharges, and seizures in human focal epilepsy," *Neurobiology of Disease*, vol. 127, pp. 545–553, 2019.

[5] E. K. S. Louis, L. C. Frey, J. W. Britton, J. L. Hopp, P. J. Korb, M. Z. Koubeissi, W. E. Lievens, and E. Pestana-Knight, *Electroencephalography (EEG): An Introductory Text and Atlas of Normal and Abnormal Findings in Adults, Children, and Infants*. 2016.

[6] P. Thangavel, J. Thomas, N. Sinha, W. Y. Peh, R. Yuvaraj, S. S. Cash, R. Chaudhari, S. Karia, J. Jing, R. Rathakrishnan, V. Saini, N. Shah, R. Srivastava, Y.-L. Tan, B. Westover, and J. Dauwels, "Improving automated diagnosis of epilepsy from eegs beyond ieds," *Journal of Neural Engineering*, vol. 19, p. 066017, Nov. 2022.

[7] X. Montoya, F. Díaz, J. Félix, J. Paucar, J. Ferrer, and P. Fonseca, "Seizure detection by analyzing the number of channels selected by cross-correlation using tuh eeg seizure corpus," in *18th International Symposium on Medical Information Processing and Analysis* (M. G. Linguraru, L. Rittner, N. Lepore, E. Romero Castro, J. Brieva, and P. Guevara, eds.), SPIE, Mar. 2023.

[8] K. K. Dutta, P. Manohar, and K. Indira, "Time and frequency domain preprocessing for epileptic seizure classification of epileptic eeg signals," *Journal of Intelligent & Fuzzy Systems*, vol. 45, p. 8217–8226, Nov. 2023.

[9] E. Lemoine, D. Toffa, G. Pelletier-Mc Duff, A. Q. Xu, M. Jemel, J.-D. Tessier, F. Lesage, D. K. Nguyen, and E. Bou Assi, "Machine-learning for the prediction of one-year seizure recurrence based on routine electroencephalography," *Scientific Reports*, vol. 13, Aug. 2023.

[10] K. Han, C. Liu, and D. Friedman, "Artificial intelligence/machine learning for epilepsy and seizure diagnosis," *Epilepsy & Behavior*, vol. 155, p. 109736, 2024.

[11] W. T. Kerr and K. N. McFarlane, "Machine learning and artificial intelligence applications to epilepsy: a review for the practicing epileptologist," *Current Neurology and Neuroscience Reports*, vol. 23, p. 869–879, Dec. 2023.

[12] P. Thangavel, J. Thomas, W. Y. Peh, J. Jing, R. Yuvaraj, S. S. Cash, R. Chaudhari, S. Karia, R. Rathakrishnan, V. Saini, N. Shah, R. Srivastava, Y.-L. Tan, B. Westover, and J. Dauwels, "Time–frequency decomposition of scalp electroencephalograms improves deep learning-based epilepsy diagnosis," *International Journal of Neural Systems*, vol. 31, no. 08, p. 2150032, 2021. PMID: 34278972.

[13] J. S. Kreutzer, J. DeLuca, and B. Caplan, eds., *Encyclopedia of Clinical Neuropsychology*. Springer New York, 2011.

[14] J. W. Britton, L. C. Frey, J. L. Hopp, P. Korb, M. Z. Koubeissi, W. E. Lievens, E. M. Pestana-Knight, and E. K. S. Louis, *Electroencephalography (EEG): An Introductory Text and Atlas of Normal and Abnormal Findings in Adults, Children, and Infants*. Chicago, IL: American Epilepsy Society, 2016. [Online].

[15] D. J. McFarland, L. M. McCane, S. V. David, and J. R. Wolpaw, "Spatial filter selection for eeg-based communication," *Electroencephalography and Clinical Neurophysiology*, vol. 103, no. 3, pp. 386–394, 1997.

[16] D. Nhu, M. Janmohamed, P. Perucca, A. Gilligan, P. Kwan, T. O'Brien, C. W. Tan, and L. Kuhlmann, "Graph convolutional network for generalized epileptiform abnormality detection on eeg," in *2021 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, pp. 1–6, 2021.

[17] G. Demoulin, E. Pruvost-Robieux, A. Marchi, C. Ramdani, J.-M. Badier, F. Bartolomei, and M. Gavaret, "Impact of skull-to-brain conductivity ratio for high resolution eeg source localization," *Biomedical Physics & Engineering Express*, vol. 7, p. 055014, Aug. 2021.

[18] O. Mecarelli, *Electrode Placement Systems and Montages*, pp. 35–52. Cham: Springer International Publishing, 2019.

[19] Syam, Syahrull Hi-Fi, Lakany, Heba, Ahmad, R.B., and Conway, Bernard A., "Comparing common average referencing to laplacian referencing in detecting imagination and intention of movement for brain computer interface," *MATEC Web Conf.*, vol. 140, p. 01028, 2017.

[20] A. Widmann, E. Schröger, and B. Maess, "High-pass filtering artifacts in erp studies," *Psychophysiology*, vol. 52, no. 8, pp. 1140–1153, 2015.

[21] S. W. Smith, *Introduction to Digital Signal Processing: A Computer Laboratory Textbook*. Newnes, 1997.

[22] W. Peng, *EEG Preprocessing and Denoising*, pp. 71–87. Singapore: Springer Singapore, 2019.

[23] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*. Prentice Hall, 1999.

[24] V. Shah, M. Smith, and A. Delorme, "Resampling methods in eeg analysis," *NeuroImage*, vol. 198, pp. 656–670, 2019.

[25] S. K. Mitra, *Digital Signal Processing: A Computer-Based Approach*. McGraw-Hill, 2007.

[26] T. Kreuz, F. Mormann, R. G. Andrzejak, A. Kraskov, H. Stögbauer, C. E. Elger, and K. Lehnertz, "Measuring synchronization in multivariate recordings from an epileptic patient," *Physiological Measurement*, vol. 28, no. 6, p. 459, 2007.

[27] S. Sanei and J. A. Chambers, *EEG Signal Processing*. Wiley, 2nd ed., 2013.

[28] A. Othmani, A. Q. M. Sabri, S. Aslan, F. Chaieb, H. Rameh, R. Alfred, and D. Cohen, "Eeg-based neural networks approaches for fatigue and drowsiness detection: A survey," *Neurocomputing*, vol. 557, p. 126709, 2023.

[29] L. Yang, J. He, D. Liu, W. Zheng, and Z. Song, "Eeg microstate features as an automatic recognition model of high-density epileptic eeg using support vector machine," *Brain Sciences*, vol. 12, no. 12, 2022.

[30] J. M. O'Toole, A. Temko, and N. Stevenson, "Assessing instantaneous energy in the eeg: A non-negative, frequency-weighted energy operator," in *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, IEEE, Aug. 2014.

[31] A. Harati, M. Golmohammadi, S. Lopez, I. Obeid, and J. Picone, "Improved eeg event classification using differential energy," in *2015 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, IEEE, Dec. 2015.

[32] D. Q. Phung, D. Tran, W. Ma, P. Nguyen, and T. Pham, "Using shannon entropy as eeg signal feature for fast person identification," in *The European Symposium on Artificial Neural Networks*, 2014.

[33] N. Bajaj, "Wavelets for eeg analysis," in *Wavelet Theory* (S. Mohammady, ed.), ch. 5, Rijeka: IntechOpen, 2020.

[34] A. K. Singh and S. Krishnan, "Trends in eeg signal feature extraction applications," *Frontiers in Artificial Intelligence*, vol. 5, Jan. 2023.

[35] Ö. Türk and M. S. Özerdem, "Epilepsy detection by using scalogram based convolutional neural network from eeg signals," *Brain Sciences*, vol. 9, no. 5, 2019.

[36] B. Gosala, P. Dindayal Kapgate, P. Jain, R. Nath Chaurasia, and M. Gupta, "Wavelet transforms for feature engineering in eeg data processing: An application on schizophrenia," *Biomedical Signal Processing and Control*, vol. 85, p. 104811, Aug. 2023.

[37] E. A. Khoursheed and A. S. Essa, "Eegs feature extraction by multi-level dwt with different numbers of principal components," in *2019 International Conference on Advanced Science and Engineering (ICOASE)*, IEEE, Apr. 2019.

[38] I. Guler and E. D. Ubeyli, "Adaptive neuro-fuzzy inference system for classification of eeg signals using wavelet coefficients," *Journal of Neuroscience Methods*, vol. 148, pp. 113–121, Oct. 2005.

[39] K. Indiradevi, E. Elias, P. Sathidevi, S. Dinesh Nayak, and K. Radhakrishnan, "A multi-level wavelet approach for automatic detection of epileptic spikes in the electroencephalogram," *Computers in Biology and Medicine*, vol. 38, no. 7, pp. 805–816, 2008.

[40] M. A. Hadj-Youcef, M. Adnane, and A. Bousbia-Salah, "Detection of epileptics during seizure free periods," in *2013 8th International Workshop on Systems, Signal Processing and their Applications (WoSSPA)*, IEEE, May 2013.

[41] G. Lee, R. Gommers, K. Wohlfahrt, *et al.*, "Pywavelets: A python package for wavelet analysis." Available: https://github.com/PyWavelets/pywt, 2019. Accessed: 2024-07-18.

[42] D. B. Percival and A. T. Walden, *Wavelet Methods for Time Series Analysis*. Cambridge Series in Statistical and Probabilistic Mathematics, Cambridge University Press, 2000.

[43] P. Stoica and R. Moses, *Spectral Analysis of Signals*. Pearson Prentice Hall, 2005.

[44] J.-P. Lachaux, E. Rodriguez, J. Martinerie, and F. J. Varela, "Measuring phase synchrony in brain signals," *Human Brain Mapping*, vol. 8, no. 4, pp. 194–208, 1999.

[45] R. Stockwell, L. Mansinha, and R. Lowe, "Localization of the complex spectrum: the s transform," *IEEE Transactions on Signal Processing*, vol. 44, no. 4, pp. 998–1001, 1996.

[46] H. K. Lee and Y.-S. Choi, "Application of continuous wavelet transform and convolutional neural network in decoding motor imagery brain-computer interface," *Entropy*, vol. 21, p. 1199, Dec. 2019.

[47] S. B. Theodore Alexandrov and T. S. McElroy, "A review of some modern approaches to the problem of trend extraction," *Econometric Reviews*, vol. 31, no. 6, pp. 593–624, 2012.

[48] A. A. Badr and M. Abdulmunim, "Hilbert transform and its applications: A survey," *International Journal of Scientific and Engineering Research*, vol. Volume 8, pp. 699–704, 02 2017.

[49] R. Stockwell, L. Mansinha, and R. Lowe, "Localization of the complex spectrum: the s transform," *IEEE Transactions on Signal Processing*, vol. 44, p. 998–1001, Apr. 1996.

[50] A. Chakraborty, S. Chatterjee, and R. Mandal, "Power quality recognition in noisy environment employing deep feature extraction from cross stockwell spectrum time–frequency images," *Electrical Engineering*, vol. 106, p. 443–458, Sept. 2023.

[51] M. Rubinov and O. Sporns, "Complex network measures of brain connectivity: Uses and interpretations," *NeuroImage*, vol. 52, p. 1059–1069, Sept. 2010.

[52] N. Tools and R. C. (NITRC), "bctpy: The brain connectivity toolbox for python." https://github.com/aestrivex/bctpy, 2023. Accessed: 2024-08-05.

[53] A. Saxena and S. Iyengar, "Centrality measures in complex networks: A survey," 2020.

[54] D. A. Lee, T. Jang, J. Kang, S. Park, and K. M. Park, "Functional connectivity alterations in patients with post-stroke epilepsy based on source-level eeg and graph theory," *Brain Topography*, Apr. 2024.

[55] M. Rashed-Al-Mahfuz, M. A. Moni, S. Uddin, S. A. Alyami, M. A. Summers, and V. Eapen, "A deep convolutional neural network method to detect seizures and characteristic frequencies using epileptic electroencephalogram (eeg) data," *IEEE Journal of Translational Engineering in Health and Medicine*, vol. 9, p. 1–12, 2021.

[56] I. H. Sarker, "Machine learning: Algorithms, real-world applications and research directions," *SN Computer Science*, vol. 2, Mar. 2021.

[57] M.-P. Hosseini, A. Hosseini, and K. Ahi, "A review on machine learning for eeg signal processing in bioengineering," *IEEE Reviews in Biomedical Engineering*, vol. 14, p. 204–218, 2021.

[58] C. Vens, J. Struyf, L. Schietgat, S. Džeroski, and H. Blockeel, "Decision trees for hierarchical multi-label classification," *Machine Learning*, vol. 73, p. 185–214, Aug. 2008.

[59] T.-T. Wong and P.-Y. Yeh, "Reliable accuracy estimates from k-fold cross validation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 8, pp. 1586–1594, 2020.

[60] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Journal of Big Data*, vol. 3, no. 1, pp. 1–40, 2016.

[61] P. P. Shinde and S. Shah, "A review of machine learning and deep learning applications," in *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, pp. 1–6, 2018.

[62] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[63] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, vol. 25, pp. 1097–1105, 2012.

[64] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[65] L. S. Shapley, "A value for n-person games," *Contributions to the Theory of Games*, vol. 2, pp. 307–317, 1953.

[66] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems*, vol. 30, (Long Beach, CA, USA), pp. 4765–4774, 2017. DOI: 10.5555/3295222.3295404.

[67] C. Rudin, C. Chen, Z. Chen, H. Huang, L. Semenova, and C. Zhong, "Interpretable machine learning: Fundamental principles and 10 grand challenges," 2021. Available: https://arxiv.org/abs/2103.11251, Accessed: 2024-08-05.

[68] H. Chen, H. Wang, and H. Liu, "Explaining neural network predictions on eeg data using shap," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 28, no. 9, pp. 1968–1978, 2020.

[69] M. Sundararajan, A. Taly, and Q. Yan, "Many shapley values for model explanation," in *Advances in Neural Information Processing Systems*, vol. 33, (Vancouver, BC, Canada), 2020. DOI: 10.5555/3454287.3454544.

[70] S. Lundberg, "Shap python library." Available: https://github.com/slundberg/shap, Accessed: 2024-08-05, 2021.

[71] I. Obeid and J. Picone, "The temple university hospital eeg data corpus," *Frontiers in Neuroscience*, vol. 10, 2016.

[72] S. J. M. Smith, "Eeg in the diagnosis, classification, and management of patients with epilepsy," *Journal of Neurology, Neurosurgery & Psychiatry*, vol. 76, no. suppl 2, pp. ii2–ii7, 2005.

[73] A. Bernasconi, N. Bernasconi, B. C. Bernhardt, and D. Schrader, "Advances in mri for "cryptogenic" epilepsies," *Nature Reviews Neurology*, vol. 7, p. 99–108, Jan. 2011.

[74] J. T. Hancock and T. M. Khoshgoftaar, "Survey on categorical data for neural networks," *J. Big Data*, vol. 7, Dec. 2020.

[75] C. Cooney, A. Korik, R. Folli, and D. Coyle, "Evaluation of hyperparameter optimization in machine and deep learning methods for decoding imagined speech eeg," *Sensors*, vol. 20, no. 16, 2020.

[76] T. Azevedo, L. Passamonti, P. Lio, and N. Toschi, "A machine learning tool for interpreting differences in cognition using brain features," 02 2019.

[77] S. Raschka, "Model evaluation, model selection, and algorithm selection in machine learning," 2018.

[78] L. Torlay, M. Perrone-Bertolotti, E. Thomas, and M. Baciu, "Machine learning–xgboost analysis of language networks to classify patients with epilepsy," *Brain Informatics*, vol. 4, p. 159–169, Apr. 2017.

[79] W. Yan, G. Xu, Y. Du, and X. Chen, "Ssvep-eeg feature enhancement method using an image sharpening filter," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 30, pp. 115–123, 2022.

[80] K. Pearson, "Mathematical contributions to the theory of evolution. iii. regression, heredity, and panmixia," *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, vol. 187, pp. 253–318, 1896.

[81] S. H. To, "Normalized root mean square error (nrmse)." https://www.statisticshowto.com/nrmse/, 2021. Accessed: 2024-08-05.

[82] S. A. Otto, R. Plonus, and S. Funk, "Normalized root mean square error." https://saskiaotto.github.io/INDperform/reference/nrmse.html, 2022. Accessed: 2024-08-05.

[83] P. Wilson, "An updated wilcoxon–mann–whitney test," in *Developments in Statistical Modelling* (J. Einbeck, H. Maeng, E. Ogundimu, and K. Perrakis, eds.), Springer, Cham, 2024.

[84] J. Frost, "Mann whitney u test explained." https://statisticsbyjim.com/mann-whitney-u-test/, 2023. Accessed: 2024-08-05.

[85] J. Mai, X. Wang, Z. Li, J. Liu, Z. Zhang, and H. Fu, "Eeg signal classification of tinnitus based on svm and sample entropy," *Computer Methods in Biomechanics and Biomedical Engineering*, vol. 26, no. 5, pp. 580–594, 2023. PMID: 35850561.

[86] A. Goshvarpour and A. Goshvarpour, "Novel high-dimensional phase space features for eeg emotion recognition," *Signal, Image and Video Processing*, vol. 17, p. 417–425, May 2022.

[87] P. Thangavel, J. Thomas, W. Y. Peh, J. Jing, R. Yuvaraj, S. S. Cash, R. Chaudhari, S. Karia, R. Rathakrishnan, V. Saini, N. Shah, R. Srivastava, Y.-L. Tan, B. Westover, and J. Dauwels, "Time–frequency decomposition of scalp electroencephalograms improves deep learning-based epilepsy diagnosis," *International Journal of Neural Systems*, vol. 31, no. 08, p. 2150032, 2021. PMID: 34278972.

[88] D. Ahmedt-Aristizabal, T. Fernando, S. Denman, J. E. Robinson, S. Sridharan, P. J. Johnston, K. R. Laurens, and C. Fookes, "Identification of children at risk of schizophrenia via deep learning and eeg responses," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 1, pp. 69–76, 2021.

[89] T. Syamsundararao, A. Selvarani, R. Rathi, N. Vini Antony Grace, D. Selvaraj, K. M. A. Almutairi, W. B. Alonazi, K. S. A. Priyan, and R. Mosissa, "An efficient signal processing algorithm for detecting abnormalities in eeg signal using cnn," *Contrast Media & Molecular Imaging*, vol. 2022, no. 1, p. 1502934, 2022.

[90] I. Tasci, B. Tasci, P. D. Barua, S. Dogan, T. Tuncer, E. E. Palmer, H. Fujita, and U. R. Acharya, "Epilepsy detection in 121 patient populations using hypercube pattern from eeg signals," *Information Fusion*, vol. 96, pp. 252–268, 2023.

[91] S. Mallick and V. Baths, "Novel deep learning framework for detection of epileptic seizures using eeg signals," *Frontiers in Computational Neuroscience*, vol. 18, Mar. 2024.

[92] O. S. Lih, V. Jahmunah, E. E. Palmer, P. D. Barua, S. Dogan, T. Tuncer, S. García, F. Molinari, and U. R. Acharya, "Epilepsynet: Novel automated detection of epilepsy using transformer model with eeg signals from 121 patient population," *Computers in Biology and Medicine*, vol. 164, p. 107312, 2023.

[93] F. Mormann, K. Lehnertz, P. David, and C. E. Elger, "Mean phase coherence as a measure for phase synchronization and its application to the eeg of epilepsy patients," *Physica D: Nonlinear Phenomena*, vol. 144, no. 3, pp. 358–369, 2000.

[94] A. A. Hagberg, P. Swart, and D. S. Chult, "Exploring network structure, dynamics, and function using networkx." Available: https://networkx.github.io/, 2008. Proceedings of the 7th Python in Science Conference (SciPy2008), Pasadena, CA USA, Aug 2008.

[95] K. Feng, H. Hong, K. Tang, and J. Wang, "Decision making with machine learning and roc curves." arXiv preprint, May 2019. Available at https://arxiv.org/abs/1905.02810.