ABSTRACT

Title of dissertation: ADVANCES IN QUANTITATIVE CHARACTERIZATIONS OF ELECTROPHYSIOLOGICAL NEURAL ACTIVITY

David O. Nahmias
Doctor of Philosophy, 2020

Dissertation directed by: Dr. Kimberly L. Kontson
Office of Science and Engineering Labs,
Center for Devices and Radiological Health,
U.S. Food and Drug Administration

Professor Jonathan Z. Simon
Department of Electrical and Computer Engineering,
University of Maryland - College Park

Disorders of the brain and nervous system result in more hospitalizations and lost productivity than any other disease group. Electroencephalography (EEG), which measures brain electrical signals from the scalp, is a common neuro-monitoring technique used for diagnostic, rehabilitative, and therapeutic purposes. Understanding EEG quantitatively and its neural correlates with patient characteristics could inform the safety and efficacy of technologies that rely on EEG. In this dissertation, a large clinical data set comprised of over 35,000 recordings as well as data from previous research experiments are utilized to better quantify characteristics of neurological activity.

We first propose non-parametric methods of evaluating consistency of quantitative EEG features (qEEG) by applying novel statistical approaches. These results

provide data-driven methods of identifying qEEG and their spatial characteristics ideal for various applications, and determining consistencies of novel features using existing data.

These qEEG are commonly used in feature-based machine learning applications. Further, EEG-driven deep learning has shown promising results in distinguishing recordings of subjects. To better understand the performance of these two machine learning approaches, we assess their ability to distinguish between subjects taking different anticonvulsants. Our methods could successfully discriminate between patients taking either anticonvulsant and those taking no medications solely from neural activity with similar performance from both feature-based and deep learning approaches.

With feature-based methods, it is easier to interpret which qEEG have the most impact on algorithm performance. However, deep learning applications in EEG can present difficulty in understanding and investigating underlying neurophysiological implications. We propose and validate a method to investigate frequency band importance in EEG-driven deep learning models. The easy perturbation EEG algorithm for spectral importance (easyPEASI) is simpler than previous methods and is applied to classifications investigated in this work.

Until this point, our work used well segmented EEG from clinical settings. However, EEG is usually corrupted by noise which can degrade its utility. We formulate and validate novel approaches to score electrophysiological signal quality based on the presence of noise from various sources. Further, we apply our method to compare and evaluate the performance of existing artifact removal algorithms.

ADVANCES IN QUANTITATIVE CHARACTERIZATIONS OF ELECTROPHYSIOLOGICAL NEURAL ACTIVITY

by

David O. Nahmias

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2020

Advisory Committee:
Professor Jonathan Z. Simon, Chair and Co-Advisor
Dr. Kimberly L. Kontson, Co-Advisor
Professor Ramalingam Chellappa
Professor Behtash Babadi
Professor Edward M. Bernat, Dean's Representative

Preface

This dissertation is mainly comprised of four research chapters. Three of these chapters have been published or submitted for publication and are presented exactly as submitted, with changes only made for formatting considerations.

To provide more context, Chapter 1 includes an extensive background of many of the methods used in the research chapters. Theory and proofs presented in this chapter are mainly from textbooks and foundational papers. Some derivations are included in detail while others omitted because of space constraints. A brief introduction on electroencephalography (EEG) is also provided. Topics covered in the introduction are: electroencephalography, power spectral density, quantitative EEG features, artifact removal algorithms, Kruskal-Wallis statistical tests, density estimation, and machine learning. Original contributions to the area of quantitative characterizations of electrophysiological neural activity are presented in Chapters 2, 3, 4, and 5; another smaller work is included in Appendix A. Each of these chapters begins with an overview outlining the chapter at a high level.

Chapter 2 presents a study on the consistency of quantitative electroencephalography features in a large clinical data set. This work has been published in the Journal of Neural Engineering in 2019 [95].

Chapter 3 presents a study on deep learning and feature based medication classifications from EEG in a large clinical data set. This work has been submitted for publication in a journal and is in peer review.

Chapter 4 presents a study on the development of easyPEASI, a simple method to identify important spectral features of EEG in deep learning models. This work

has been submitted for publication as a conference proceeding for a conference with a machine learning focus and is in peer review.

Chapter 5 presents a study quantifying signal quality from unimodal and multimodal sources with application to EEG with ocular and motion artifacts. This work has not yet been submitted for publication. It is expected some or all elements of this chapter will be submitted for publication in the future.

Chapter 6 briefly concludes and summarizes the works of this dissertation, their novel contributions to the field, and potential future directions.

Finally, Appendix A presents a smaller study on EEG spectral connectivity analysis in a large clinical population. This work applies existing methods to our larger data set and though presents interesting results is not included in the main text of this dissertation. Methods used exclusively in this appendix, mainly graph theory and spectral graph theory, are not discussed in the introduction. This work has been published as a conference proceeding as part of the 9th IEEE/EMBS Neural Engineering (NER) conference in 2019 [93].

Online resources, including software and future publications, for reproducibility and future applications of this dissertation can be found at `dnahmias.com`.

Dedication

Mami lapin, merci pour tout.

(This page intentionally left blank)

Acknowledgments

I owe gratitude to so many people who have made this dissertation possible and because of whom I will cherish this experience forever.

First and foremost I would like to thank my advisor Dr. Kimberly Kontson for giving me an invaluable opportunity to work on meaningful, challenging, and interesting projects over the past four years at the US Food and Drug Administration. Her support and faith in me as I chose projects and dove in is something I will always be thankful for. Of course as I inevitably ran into hurdles, she always made herself available for help and advice. It has been a pleasure to work with and learn from such a wonderful and extraordinary individual. I have always felt lucky to have her as an advisor and mentor. I look forward to continue working with her even as this chapter of my research career comes to a close as a colleague and friend. As I have said many times before, "Thank you Kim," for all your support throughout these years, I will be forever grateful.

I would also like to thank my co-advisor, Professor Jonathan Simon from the electrical and computer engineering department at UMD. Without his support throughout my graduate career, this dissertation would not have been possible. He was always available to discuss my progress and give guidance, and for that I am grateful. Thanks are also due to Professor Rama Chellappa, Professor Behtash Babadi, and Professor Ed Bernat for agreeing to serve on my dissertation committee and for sparing their invaluable time reviewing the manuscript.

I would be remiss if I did not mention Dr. Eugene Civillico. He has been

a great mentor throughout these past few years and guide through the field of neuroscience. No matter how mathematical I made things, he always found a way to keep us grounded in the application, importance, and impact of our work. And to my colleagues in the Human Device Interaction lab at the FDA, Sophie, J.J., and Chris, you have all enriched my experience in so many ways. I could not have asked for better lab-mates.

Above all, I owe my deepest thanks to my family, who have always been there for me and supportive through everything in my life. Je suis lá grâce á vous tous.

I would like to acknowledge financial support from the Center for Devices and Radiological Health, Office of Science and Engineering Laboratories, Division of Biomedical Physics and from the DARPA-FDA IAA 224-14-6009, DARPA Biological Technology Office through the Hand Proprioception and Touch Interfaces (HAPTIX) program.

There have been so many colleagues, classmates, friends, and mentors that have helped me along this journey and so, lastly, thank you all!

Table of Contents

List of Tables

List of Figures

List of Abbreviations

AAR	Automated Artifact Removal
ANOVA	Analysis of Variance
BCI	Brain Computer Interface
BMI	Brain Machine Interface
BSS	Blind Source Separation
CDF	Cumulative Density Function
COV	Coefficient of Variation
DARPA	Defense Advanced Research Projects Agency
dB	Decibels
DCNN	Deep Convolution Neural Network
DFT	Discrete Fourier Transform
DTFS	Discrete Time Fourier Series
DTFT	Discrete Time Fourier Transform
easyPEASI	easy Perturbation EEG Algorithm for Spectral Importance
ECG	Electrocardiography
EEG	Electroencephalography
ELU	Exponential Linear Unit
EM	Expectation-Maximization
EMG	Electromyography
EOG	Electrooculography
FBCSP	Filter-Bank Common Spatial Pattern
FDA	U.S. Food and Drug Administration
FS	Fourier Series
FT	Fourier Transform
FFT	Fast Fourier Transform
GMM	Gaussian Mixture Model
HAPTIX	Hand Proprioception and Touch Interfaces
Hz	Hertz
ICA	Independent Component Analysis
IMU	Inertial Measurement Unit
IQR	Inter-Quartile Range

KDE	Kernel Density Estimation
kSVM	Kernel Support Vector Machine
LU	Linear Unit
LZC	Lempel-Ziv Complexity
MAP	Maximum A Posterior
MARA	Multiple Artifact Removal Algorithm
MLE	Maximum Likelihood Estimator
NEDC	Neural Engineering Data Consortium
PCA	Principle Component Analysis
PDF	Probability Density Function
PSD	Power Spectrum Density
qEEG	Quantitative Electroencephalography
RBF	Radial Basis Function
ReLU	Rectified Linear Unit
SD	Standard Deviation
SGD	Stochastic Gradient Descent
SNR	Signal to Noise Ratio
SOBI	Second Order Blind Identification
SR	Sampling Rate
SVM	Support Vector Machine
TUH	Temple University Health
UMD	University of Maryland - College Park
V	Volts
ZCA	Zero Component Analysis

List of Notations

$\forall$	For all
$\in$	Is an element of
$\emptyset$	Empty set, null set
$\perp$	Perpendicular, independent
$[\cdot]$	Discrete index
$(\cdot)$	Continuous or discrete value
$\{\dots\}$	Set
$\mathbb{R}$	The field of real numbers
$\mathbb{C}$	The field of complex numbers
$\mathbb{N}$	The set of natural numbers $\{0, 1, 2, 3, \dots\}$
$\mathbb{N}^+$	The set of positive natural numbers $\{1, 2, 3, \dots\}$
$\mathbb{Z}$	The set of integer numbers $\{\dots, -2, -1, 0, 1, 2, \dots\}$
$M_{i,j}$	The $(i, j)^{\text{th}}$ element of the matrix M
$M_{R\times C}$	A matrix, M, with R rows and C columns
v_i	The i^{th} element of the vector v
$Re(\cdot)$	The real component of $\cdot$
$Im(\cdot)$	The imaginary component of $\cdot$
$\lvert \cdot \rvert$	Absolute value operator
$\lVert \cdot \rVert_p$	L^p-Norm operator, $(\sum_i \lvert \cdot_i \rvert^p)^{1/p}$
$\lVert \cdot \rVert_\infty$	L^∞-Norm operator, $\max_i \lvert \cdot_i \rvert$
$\#\cdot$	Cardinality operator
$\vec{\cdot}$	Vector array
$(\cdot)^T$	Transpose operator
$(\cdot)^*$	Complex conjugate operator
$(\cdot)^H$	Hermitian operator, the conjugate-transpose
$L^1(\mathbb{R}^d)$	Space of functions where $\int \lvert \cdot \rvert \mathrm{d}\cdot < \infty$, $\forall \cdot \in \mathbb{R}^d$ for $d \in \mathbb{N}^+$
$L^2(\mathbb{R}^d)$	Space of functions where $\int \lvert \cdot \rvert^2 \mathrm{d}\cdot < \infty$, $\forall \cdot \in \mathbb{R}^d$ for $d \in \mathbb{N}^+$
$\mathcal{F}\{\cdot\}$	Fourier transform operator
$\log_b(\cdot)$	Log operator with base b
$\ln(\cdot)$	Log operator with base e, natural log
$sgn(\cdot)$	Sign operator: $sgn(\cdot) = -1, \cdot < 0$; $sgn(\cdot) = 1, \cdot > 0$; 0 otherwise
$det(\cdot)$	Determinate operator
$(\cdot)'$	First Derivative
$(\cdot)''$	Second Derivative
$(\cdot)^{(n)}$	n^{th} derivative
$\nabla(\cdot)$	Gradient operator, $[\partial\cdot/\partial x_1, \dots, \partial\cdot/\partial x_d]$, $\forall \cdot \in \mathbb{R}^d$

(This page intentionally left blank)

Chapter 1: Introduction

Disorders of the brain and nervous system, of which there are more than 1,000, result in more hospitalizations and lost productivity than any other disease group [24], [144]. The World Health Organization estimated that neurological diseases make up 11% of the world's disease burden, not including mental health and addiction disorders, and affect up to one billion people worldwide [122], [144]. Electroencephalography (EEG), which measures brain electrical signals from the scalp, is a common neuro-monitoring technique used in both clinical and research settings for diagnostic, rehabilitative, and therapeutic purposes. EEG is typically non-invasive, mobile, and relatively inexpensive which makes it a promising neuro-monitoring method for widespread use on large populations.

The prevalence and availability of physiological data from portable non-invasive devices is growing at a rapid rate. How we use and quantify these data could effect the safety and effectiveness of medical devices that depend of them. This work aims to leverage new big data sources that have recently become available to better understand population-level quantitative characteristics of neural activity.

To achieve this aim we bring together topics discussed in this chapter. We begin with some basic background on EEG followed by common analytical methods

used in EEG research including spectral analysis, quantitative features, and artifact removal. Next we discuss a statistical test used throughout the dissertation, the Kruskal-Wallis test. As a segue into machine learning methods we present density estimation since these methods are often used both in statistical and machine learning applications. Finally, we discuss both feature-based (support vector machines) and deep learning-based (deep convolutions neural networks) methods in machine learning that are used throughout this dissertation. Altogether, applying quantitative methods such as machine learning and statistical approaches to large data sets, we can better characterize EEG data and advance our understanding of quantitative characteristics of neural activity from EEG.

Overview

The purpose of this chapter is to give a brief overview and background on methods used in this dissertation. Of course it would be impossible to give a proper and thorough treatment of all fields we draw upon in this work. Rather the goal is to give some background on specific methods used. Theory and proofs presented in this chapter are mainly drawn from textbooks and foundational papers. Certain sections topics, and their respective references should more detail be desired by the reader, are: electroencephalography [11], [134], [125]; power spectral density [109], [108], [101], [53], [137], [10]; quantitative EEG features [125]; artifact removal algorithms [143], [45], [1], [9], [63]; Kruskal-Wallis statistical tests [114], [113], [126], [79], [77]; density estimation [103], [116], [34]; machine learning [47], [136], [13], [32], [19].

1.1 Electroencephalography (EEG)

This section gives a brief introduction on EEG adapted from the 'Handbook of EEG Interpretation' and 'Niedermeyer's Electroencephalography: Basic Principles, Clinical Applications, and Related Fields' [134], [125].

Non-invasive surface EEG monitors the electrical activity of the brain from electrodes placed on the scalp. This is characterized by the voltage changes associated with the post-synaptic potentials from large synchronized activations of neocortical pyramidal cells. This was first measured in humans by Hans Berger in 1924 [11]. The mechanism within neurons that create this action potential, voltage, that is measured through surface electrodes is the exchange of sodium and potasium ions in and out of the cells. Adenosine Triphosphate (ATP) provides the energy for proteins to push out the sodium ions while pulling in the potassium ions through the cell. This makes the outside of the cells more positively charged and the neurons more negatively charged. This rapid exchange is the source of the action potentials that are measured. In general, only through the summation of a large collective activity of many neurons with similar spatial orientation can an EEG signal be detected from the scalp.

In terms of EEG electrodes and capturing the electrophysiolgical signals, these electrodes should maintain an impedance between 100 to 5000 Ohms. Electrodes placed usually follow a standard naming and corresponding placement convention. In general, the following table refers to the names and locations of electrodes:

Label	Location/Lobe
A1	Left Ear
A2	Right Ear
Fp	Frontal Pole
F	Frontal
P	Parietal
C	Central
T	Temporal
O	Occipital

Table 1.1 EEG lead labels and locations

A second letter of 'z' designates an electrode placed on the midline of the scalp. Even numbers refer to electrodes placed on the right hemisphere, and odd numbered electrodes refer to electrodes placed on the left hemisphere.

The placement of these electrodes usually follow was is called the 10-20 system which uses anatomical landmarks on the scalp. The numbers '10-20' refer to the intervals of 10% or 20% used for electrode placement. For the center-line, we consider the top 180° of the scalp. Starting at eye level at the front of the head we go up 10% (18°) to find the first electrode locations. Then we move up 20% (36°) to find the next electrode site. Moving 20% (36°) again, we arrive at the top of the head. The remaining sites between the top of the head and the back of the head are placed in a similar fashion in reverse, moving down from the top by 20% (36°), then 20% (36°) again, and finally 10% (18°). Electrodes are also placed based on these same percentages away from the midline of the head around the sides of scalp. This was initially introduced by H. H. Jasper in 1958 [64]. This has also since been extended to the 10-10 and 10-20 systems for the inclusion of more electrodes and higher spatial resolution [100]. The main electrode layout with 19 channels, used in three of our

research chapters and appendix, is shown in Figure 1.1. The final research chapter used recordings with 32 and 64 channel EEG 10-20 extended layouts, shown in Figure 1.2a and Figure 1.2b, respectively. We note that some electrodes overlap since these figures show three-dimensional electrode locations projected in two dimensions.

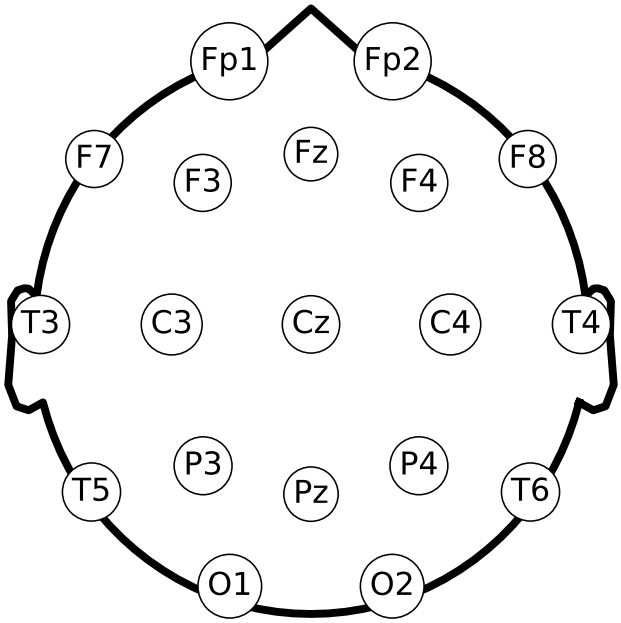
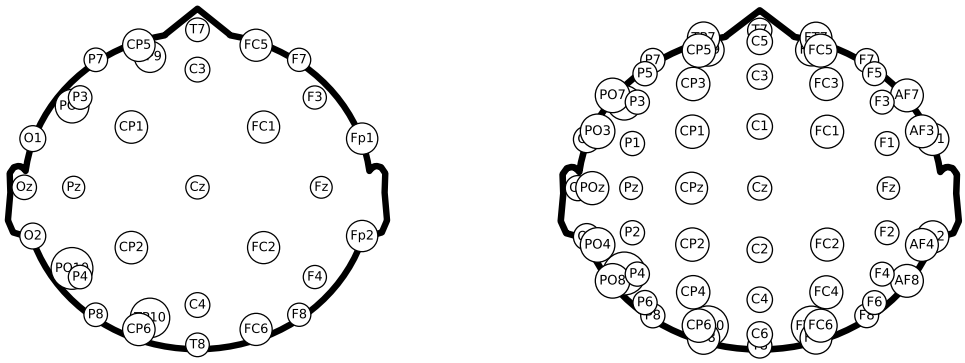


Figure 1.1 10-20 EEG electrode layout with 19 channels.

Each of the lobes referred to in the EEG electrode naming scheme have been studied extensively and have each been associated with specific cognitive functions. The frontal lobe is the region of the brain where conscious thoughts and decisions are made. Also, the frontal lobe contains motor areas where voluntary motor movements are controlled. The parietal lobe integrates information from external sources as well as internal sensory information such as muscles, and eyes. The parietal lobe is also responsible for combining all of these senses and stimuli into a coherent representation of our body. For example, hand-eye coordination would be impossible



(a) 32 channel extended 10-20 layout. (b) 64 channel extended 10-20 layout.

Figure 1.2 Extended 10-20 EEG electrode layout with (a) 32 (b) 64 channels.

without parietal lobe. Further, the parietal lobe is often studied for feelings of agency. The temporal lobe is responsible for long-term memory. It also includes the hippocampus, which is a central structure for memory. The left temporal cortex is responsible for language comprehension, and production, including Broca's area. Finally, the occipital lobe is where visual processing occurs. The occipital lobe is located in the rearmost part of the brain. There is no actual Central lobe, so C electrodes in EEG are generally used for identification purposes only.

EEG signals themselves usually range from $1-200\mu$V. Figure 1.3 shows a sample of EEG recording for one minutes of time across 19 channels. As we see, it can be difficult to interpret the underlying neurological activity directly from these signals. For these reasons developing quantitative representations and interpretations of EEG has been an active area of research.

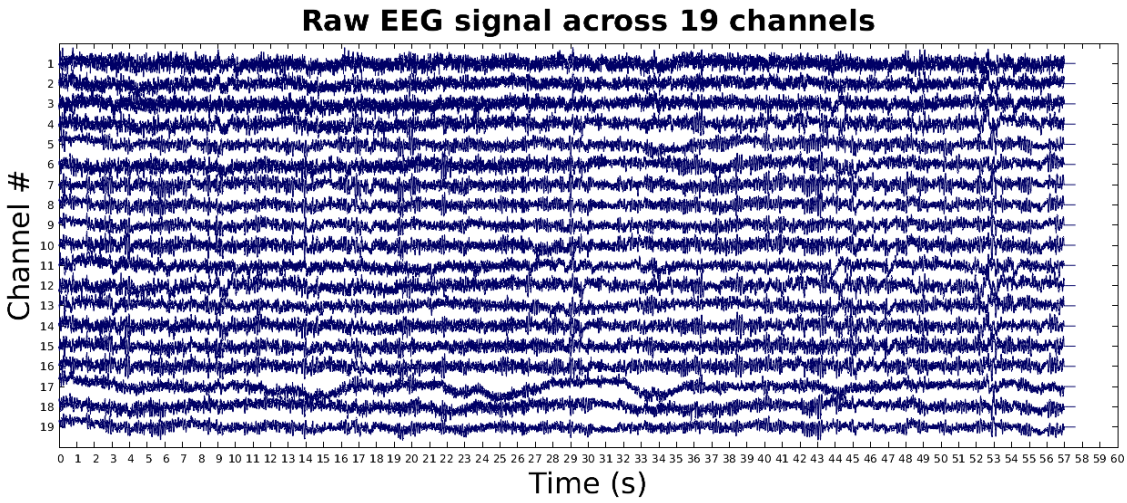


Figure 1.3 Sample EEG trace across 19 channels.

Until recently, the most common quantitative method has generally been spectral analysis of EEG. Different ranges of frequency vary in amplitude. In general, EEG spectrum follows a $1/f$ trend, decreasing in power as the frequency increases. For practical purposes, the range of frequencies that are usually considered of interest is below 50Hz. In addition, in terms of frequency information Figure 1.4 shows a sample power spectrum density (PSD) of a single EEG channel, with certain frequency bands marked. In general ranges of frequencies have been grouped together into established bands commonly analyzed and referred to in EEG research [99]. The exact cut-off frequencies of each band vary slightly depending on EEG applications or studies. In this research, we generally refer to the following frequency ranges for each bands: δ(delta) : $1-4$Hz, θ(theta) : $4-8$Hz, α(alpha) : $8-12$Hz, μ(mu) : $12-16$Hz, β(beta) : $16-25$Hz, γ(gamma) : $25-40$Hz. These frequency ranges are based on the band definitions in Niedermeyer's Electroencephalography except with the introduction of a μ band labeling the lower frequencies of the β

band separately [125]. Each of these bands have established normal properties as well as many studies linking changes in spectral properties to physiological or clinical differences that could not all be detailed here [134].

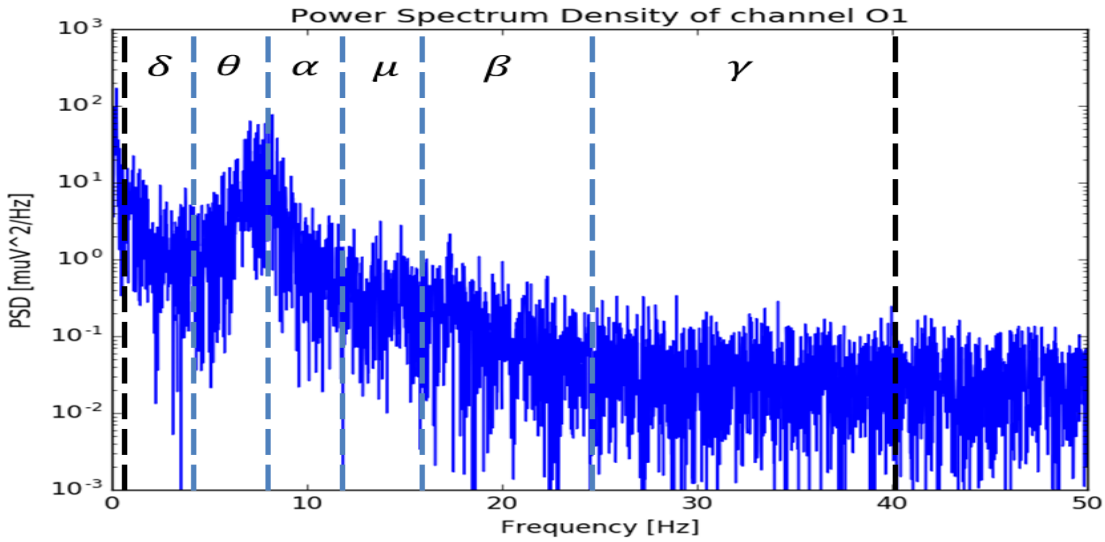


Figure 1.4 Sample power spectrum density of channel O1.

1.2 Power spectral density

With spectral analysis of EEG common, we briefly outline power spectral density (PSD) estimation. For all spectral estimation methods, we assume that the signal has finite energy, that is $x[t] \in L^2(\mathbb{R})$. Formally, the energy spectral density can be defined for a discrete signal as follows.

Definition 1.2.1. Using the definition of the discrete time Fourier transform (DTFT), $DTFT(x[t]) = \mathcal{F}\{x\}[e^{i\omega}]$ we have energy spectral density to be

$$S[\omega] = |\mathcal{F}\{x\}[e^{i\omega}]|^2 \tag{1.1}$$

writing $S[\omega]$ in lieu of $S[e^{i\omega}]$.

The essence of the spectral estimation problem is that we wish to estimate $S[\omega]$, how the total power of a signal is distributed over frequency given finite stationary data. In particular, we focus on non-parametric methods of spectral analysis. Specifically, a spectral estimation method that has commonly been used is the Welch method [142]. For a sampled signal $x[n], n = 1, \ldots, N$, we let

$$x_j[n] = x[(j-1)K + n], \quad n = 1, \ldots, M; j = 1, \ldots, S \tag{1.2}$$

be the j^{th} part of the signal. $(j-1)K$ denotes the starting point for the j^{th} sequence. If $K = M$ then the sequences do not overlap, thus $S = {}^{N}/_{M}$. Usually, $K = {}^{M}/_{2}$ and so there are $S \approx {}^{2M}/_{N}$ parts, with 50% overlap between successive parts. The windowed periodogram is then computed by

$$\phi_{W_j}(\omega) = \frac{1}{MP} \left| \sum_{n=1}^{M} v[n] x_j[n] e^{i\omega n} \right|^2, \tag{1.3}$$

where $v[n]$ is dictated by the window design such that,

$$w[k] = \frac{1}{N} \sum_{n=1}^{N} v[n] v^*[n-k] \tag{1.4}$$

and with the DTFT we see

$$E[\phi(\omega)] = \mathcal{F}\{w\}(\omega) * \phi_p(\omega) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \mathcal{F}\{w\}(\omega - \psi) \phi_p(\psi) \mathrm{d}\psi, \tag{1.5}$$

where $P = \frac{1}{M}\sum_{n=1}^{M}|v[n]|^2$ is the power of the temporal window. So, the Welch estimate of the power spectrum is found by computing the average windowed periodograms from Equation 1.3,

$$\phi_W(\omega) = \frac{1}{S}\sum_{n=1}^{S}\phi_{W_j}(\omega) \tag{1.6}$$

The Welch method, which allows for overlap, unlike the Bartlett method, enables more periodograms to be averaged [7], [8]. This is done to decrease the variance in the estimated PSD. Further, applying the window in the periodogram computation is again intended to allow for more control over the bias and resolution of the estimated PSD.

1.3 Quantitative EEG features

In addition to frequency band analysis, quantitative EEG research also uses quantitative features to represent EEG and analyze differences between populations. Below we define EEG features used throughout our work. Though many qEEG features exist, we chose to focus on these qEEG features based on their use in seizure research and other EEG consistency studies [29], [50]. For each of these features, they are computed on each EEG channel separately.

1.3.1 Spectral features

We first begin with features computed from the PSD of an EEG trace.

For frequency bands: ℓ(lower) $:<$ 1Hz, δ(delta) : $1-4$Hz, θ(theta) : $4-8$Hz,

α(alpha) : $8-12$Hz, μ(mu) : $12-16$Hz, β(beta) : $16-25$Hz, γ(gamma) : $25-40$Hz.

Relative band powers are defined as,

$$PSDR_b = \frac{\sum_{\omega \in b} P[\omega]}{\sum_{\forall f} P[f]}, \forall b \in \{\ell, \delta, \theta, \alpha, \mu, \beta, \gamma\} \tag{1.7}$$

while absolute band powers,

$$PSDA_b = \sum_{\omega \in b} P[\omega], \forall b \in \{\ell, \delta, \theta, \alpha, \mu, \beta, \gamma\} \tag{1.8}$$

where $P[\omega] = \frac{1}{N \cdot SR} |(\mathcal{F}\{X\})[\omega]|^2$ and $(\mathcal{F}\{X\})[\omega] = \sum_{i=0}^{N-1} x[i] exp\left(\frac{-2\pi j \omega i}{N}\right)$. With $N = \#X$, the cardinality of X, and SR the sampling rate.

Spectral Entropy is computed through the entropy of the normalized periodogram,

$$SE = -\sum_{k=0}^{N} \frac{P[k]}{\sum_{\forall \omega} P[\omega]} \log_2 \left(\frac{P[k]}{\sum_{\forall \omega} P[\omega]} \right). \tag{1.9}$$

1.3.2 Time domain features

Next we describe features that were computed using the EEG recording directly. For these features we use S the starting index and N the ending index.

We start with information theoretic features of EEG.

Entropy of shifted signal X, normalized,

$$ENT = -\sum_{i=S}^{N} \frac{x_s[i]}{\sum_{j=S}^{N} x_s[j]} \log_2 \left(\frac{x_s[i]}{\sum_{j=S}^{N} x_s[j]} \right) \tag{1.10}$$

where $x_s[i] = x[i] + \min\{X\}$.

Lempel-Ziv Complexity (LZC), obtained by converting X to a binary sequence through mean thresholding,

$$bs[i] = \begin{cases} 0, & \text{if } x[i] < T_{mean(X)} \\ 1, & \text{otherwise} \end{cases} \tag{1.11}$$

and LZC is computed through [92]. Based on methods from [5], [59].

We next define standard statistical measures also used to quantify EEG.

Minimum value,

$$MIN = \min\{x_S, \ldots, x_N\}. \tag{1.12}$$

Maximum value,

$$MAX = \max\{x_S, \ldots, x_N\}. \tag{1.13}$$

Median,

$$MED = \begin{cases} X_{sort}[\#X \div 2], & \#X \text{ even} \\ \frac{X_{sort}[\#X \div 2] + X_{sort}[\#X \div 2 + 1/2]}{2}, & \#X \text{ odd} \end{cases} \tag{1.14}$$

where X_{sort} is a sorted X and $\#X$ is the cardinality of X.

Variance,

$$VAR = E\left[(X - \mu)^2\right] = \frac{1}{N - S - 1} \sum_{i=S}^{N} (x[i] - \mu)^2. \tag{1.15}$$

Standard deviation,

$$SD = \sigma = \sqrt{VAR}. \tag{1.16}$$

Skew,

$$SKEW = E\left[\left(\frac{X-\mu}{\sigma}\right)^3\right] = \frac{\frac{1}{N-S}\sum_{i=S}^{N}(x[i]-\mu)^3}{\left[\frac{1}{N-S-1}\sum_{i=S}^{N}(x[i]-\mu)^2\right]^{3/2}}. \tag{1.17}$$

where $\mu = \frac{1}{N-S}\sum_{i=S}^{N} x[i]$.

Kurtosis,

$$KURT = E\left[\left(\frac{X-\mu}{\sigma}\right)^4\right] = \frac{\frac{1}{N-S}\sum_{i=S}^{N}(x[i]-\mu)^4}{\left[\frac{1}{N-S}\sum_{i=S}^{N}(x[i]-\mu)^2\right]^2} - 3. \tag{1.18}$$

Finally, we define features which represent the signal shape properties of the EEG trace.

Curve length,

$$CL = \sum_{i=S}^{N-1} |x[i] - x[i+1]|. \tag{1.19}$$

Energy,

$$E = \frac{1}{N-S}\sum_{i=S}^{N}(x[i])^2. \tag{1.20}$$

Non-linear energy,

$$NE = \frac{1}{N-S}\sum_{i=S+1}^{N-1}(x[i])^2 - x[i-1]x[i+1]. \tag{1.21}$$

Sixth power,

$$SP = \frac{1}{N-S}\sum_{i=S}^{N}(x[i])^6. \tag{1.22}$$

Sum,

$$SUM = \sum_{i=S}^{N} x[i]. \tag{1.23}$$

Mobility,

$$MOBI = \sqrt{\frac{a_1}{a_0}} \tag{1.24}$$

where $a_0 = VAR(X)$, $a_1 = VAR(X^{(1)})$, and where $X^{(n)}$ is the n^{th} discrete difference, derivative, of X.

Complexity,

$$COMP = \sqrt{\frac{a_2}{a_1} - \frac{a_1}{a_0}} \tag{1.25}$$

where $a_2 = VAR(X^{(2)})$.

1.4 Artifact removal in EEG

Artifact removal in EEG is an area of active research. It is a common and usually the most burdensome step in pre-processing EEG before analysis. Even when noise sources are known, artifacts cannot be directly subtracted from EEG signals and more sophisticated methods are needed to remove them. It is generally difficult to do in an automated fashion and usually requires manual review for consistent and accurate artifact removal. Because most of the data we used were well segmented and recorded in controlled clinical settings, we generally did not apply artifact removal algorithms. However in our last research chapter, as an application to one of our developed methods, we evaluate the effectiveness of automated artifact removal methods. Below we give some background and formulations that are the bases of the artifact removal algorithms used.

1.4.1 Independent component analysis

When EEG electrodes record electrical activity, both neural activity and artifacts from other physiological sources are generally mixed together. The presence of artifactual noise corrupts the neural signal of interest limiting its interpretability and utility. To motivate independent component analysis (ICA) in the context of EEG, recall that neural activity is recorded from a collection of electrodes from the scalp. However, as electrical activity permeates through the brain and skull, each electrode records a combination of neural and muscle activity from multiple sources in the brain and body. The goal of ICA, and more broadly blind source separation (BSS), is to identify and separate the sources of electrical activity, neural or artifactual, for each EEG electrode. If we assume that the signals are being linearly mixed, then for the set of electrode recordings, X, and separate neural activity sources, S, we can define

$$X = AS, \tag{1.26}$$

where A is called the mixing matrix. If we had either A or S, then we could solve for the other by inverting the system of equations. However, we know neither the mixing coefficients nor the neural activity and only have the recorded EEGs, X. We first assume that each source provides statistically independent data at each time instant. In practice this assumption does not need to be strictly true to obtain accurate solutions. Further, it is necessary that the independent components have non-Gaussian distributions. Otherwise, an orthogonal mixing matrix cannot

be estimated since uncorrelated jointly Gaussian variables are necessarily independent, and therefore A cannot be identified and only estimated up to an orthogonal transform.

In general, most ICA algorithms are designed to solve the un-mixing matrix, $W = A^{-1}$, the inverse of A. That is, algorithms look to find the solution to

$$WX \approx S \tag{1.27}$$

Once the ICA components are solved for, in the case of artifact removal, specific components that represent the noise sources (e.g., eye movement, muscles, and motion) are removed. That is, once all components $\{w_1, \ldots, w_n\}$ are solved, for $i = 1, \ldots n$, algorithms set $w_i = 0$ if i is identified a noise source.

However, even when W is solved for, a difficulty is identifying which components represent noise and which are relevant neural activity. Though most artifact removal algorithms do not allow automated application, there are several methods that have been developed to identify artifactual components automatically. In particular we use multiple artifact removal algorithm (MARA) [143], [62] and automated artifact removal (AAR) [45], [69], [70]. These algorithms and component identification methods are described when applied in our research in Chapter 5.

1.4.1.1 Pre-processing

To solve ICA several pre-processing steps simplify theory and algorithms, beginning with centering X by imposing that the recorded data have zero mean. This

is usually followed by whitening the data so that elements of X are uncorrelated and have unit variance, that is $E[\tilde{X}\tilde{X}^T] = I$, where $\tilde{X}$ is the covariance matrix of X.

We define the whitening transform, sometimes referred to as zero component analysis (ZCA), as follows [32].

Definition 1.4.1. If we consider data $X[n, i]$, where $n \in \{1, \ldots, N\}$ are each data point and $i \in \{1, \ldots, M\}$ the sample index, then the whitening transform is defined as,

$$X_{whitened} = Z \cdot X, \tag{1.28}$$

where

$$Z_{N\times N} = U \cdot diag(1/\sqrt{S+\epsilon}) \cdot U^T \tag{1.29}$$

with $\epsilon = 1 \times 10^{-5}$, included for numerical reasons, and where

$$\Sigma_{N\times N} = U_{N\times N} \cdot diag(S_{N\times 1}) \cdot U^T_{N\times N} \tag{1.30}$$

is the singular value decomposition of $\Sigma_{N\times N}$, the covariance matrix of $X_{N\times M}$. Note for vector $S_{N\times 1}$, $diag(S)$ is a diagonal $N \times N$ matrix with elements of S on the diagonal and zero everywhere else.

This transform can be thought of as scaling the variance across each dimension in the data to be equivalent. In other-words, the data space may initially be different sizes in each dimension but is transformed to a hyper-sphere. Once pre-processed there are several approaches to solving the ICA, we briefly outline two that are used

in our work here.

1.4.1.2 FastICA

Adapted from 'Independent Component Analysis', the first is a classic approach called FastICA and uses extrema of functions which measure non-Gaussianity of data [1], [57]. Kurtosis, the fourth-order moment of a random variable, is zero for Gaussian random variables and generally non-zero for all other non-Gaussian random variables, is an example of a measure of non-Gaussianity. Here $\text{kurt}(x) = E[x^4] - 3(E[x^2])^2 = E[x^4] - 3$ because of x's unit variance, $E[x^2] = 1$. However, kurtosis when estimated from measured samples is sensitive to outliers and thus not a robust measure of non-Gaussianity.

A more robust measure of non-Gaussianity of data is Negentroy. Entropy of a discrete random variable X is defined as

$$H(X) = -\sum_i P(X = x_i) \log\left(P(X = x_i)\right). \tag{1.31}$$

A fundamental result of information theory is that a Gaussian variable has the largest entropy among all random variables of equal variance and therefore can be used as a measure of non-Gaussianity [26]. We define Negentropy, a measure of non-Gaussianity that is zero for Gaussian variables and non-negative, as

$$N(x) = H(x_{gauss}) - H(x), \tag{1.32}$$

where differential entropy $H(x) = \int f(x) \log(f(x)) \mathrm{d}x$ and x_{gauss} is a Gaussian random variable with the same convariance matrix as x. Though this presents a somewhat optimal measure of non-Gaussianity, estimating negentropy would require estimating the PDF which can be difficult and computationally complex. Therefore, approximations of negentropy are used.

An early approximation, which has been proven to converge globally, used kurtosis such that $N(x) \approx (1/12)\, E[x^3]^2 + (1/48)\, \mathrm{kurt}(x)^2$, where $x \sim \mathcal{N}(0,1)$ is assumed. However, the same non-robustness issues arise for this approximation. Therefore, using the maximum-entropy principle this approximation form was generalized to

$$N(x) \approx \sum_{i=1}^{p} \kappa_i \left(E[G_i(x)] - E[G_i(\nu)]\right)^2, \tag{1.33}$$

where G_i are generally any non-quadratic function, κ_i are positive constants, and $\nu \sim \mathcal{N}(0,1)$, a Gaussian variable with zero mean and unit variance. We see setting $p = 2$ with $G_1(x) = x^3$, $G_2(x) = x^4$, $\kappa_1 = 1/12$, and $\kappa = 1/48$ yields the kurtosis-based approximation, noting that $E[\nu^3] = 0$. For $p = 1$, we have

$$N(x) \approx \kappa_1 (E[G_1(x)])^2. \tag{1.34}$$

More generally for $p = 2$

$$N(x) \approx \kappa_1 (E[G_1(x)])^2 + \kappa_2 (E[G_2(x)] - E[G_2(\nu)])^2, \tag{1.35}$$

where G_1 is an odd function and G_2 even.

For the implementations of FastICA, generally,

$$G_1(x) = \quad \tanh(a_1 x) \tag{1.36}$$

$$G_2(x) = \quad x \exp\left(-\frac{x^2}{2}\right) \tag{1.37}$$

where $1 \leq a_1 \leq 2$, often simply $a_1 = 1$.

So to optimize the approximation of the negentropy to determine the unmixing components w, if we consider for simplicity $p = 1$, we seek to optimize $E[G(w^T x)]$. By the Kuhn-Tucker conditions, the optima of $E[G(w^T x)]$ with $\|w\| = 1$ constraint are at points where

$$E[xG'(w^T x)] - \beta w = 0, \tag{1.38}$$

where $\beta = E[w_{opt}^T x G'(w_{opt}^T x)]$. If we apply Newton's method for iterations,

$$w^+ = w - (J(w))^{-1} f(w), \tag{1.39}$$

where $J(w)$ is the Jacobian. We now derive the Jacobian,

$$
\begin{aligned}
J(w) = & \quad \tfrac{\partial}{\partial w}\left(E[xG'(w^Tx)] - \beta w\right) && (1.40)\\
= & \quad E[xG''(w^Tx)x^T] - \beta I && (1.41)\\
= & \quad E[xx^TG''(w^Tx)] - \beta I && (1.42)\\
\approx & \quad E[xx^T]E[G''(w^Tx)] - \beta I, \text{ by approximation} && (1.43)\\
= & \quad E[G''(w^Tx)]I - \beta I, \text{ since whitened data, } E[xx^T] = I. && (1.44)
\end{aligned}
$$

Inverting $J(w)$ and plugging it into Equation 1.39 we have,

$$
\begin{aligned}
w^+ = & \quad w - \tfrac{E[xG'(w^Tx)] - \beta w}{E[G''(w^Tx)] - \beta} && (1.45)\\
w^+(\beta - E[G''(w^Tx)]) = & \quad w(\beta - E[G''(w^Tx)]) + E[xG'(w^Tx)] - \beta w, && (1.46)\\
& \quad \text{multiplying both sides by } \beta - E[G''(w^Tx)] &&\\
= & \quad E[xG'(w^Tx) - E[G''(w^Tx)] && (1.47)
\end{aligned}
$$

This update rule along with normalizing $w = {}^{w^+}/_{\|w^+\|}$ presents a basic form of FastICA for a single component. To estimate all components $w_1, \ldots, w_n$, we must decorrelate the outputs $w_i^T x$ after each iteration to prevent different components from converging to the same maxima. This can be achieved through Gram-Schmidt decorrelation. Thus, we can formulate the FastICA algorithm below where w_i are the column vectors of the unmixing matrix W [58].

Algorithm FastICA

1: $w_i \leftarrow$ initialized randomly
2: $w_i^+ \leftarrow E[XG'(w_i^T X)] - E[G''(w_i^T X)]w_i$
3: $w_i \leftarrow \frac{w_i^+}{\|w+_i\|}$
4: **if** $i \neq 1$ **then**
5: $\quad w_i^+ \leftarrow w_i - \sum_{j=1}^{i-1} w_i^T w_j w_j$
6: $\quad w_i \leftarrow \frac{w_i^+}{\|w_i^+\|}$
7: **else**
8: $\quad$ **if** Not converged **then** Return to line 2
9: $\quad$ **else**
10: $\quad\quad i \leftarrow i + 1$
11: $\quad\quad$ Return to line 1

FastICA has several properties that make it an effective algorithm for solving ICA. The algorithm's convergence is cubic, making it fast and, unlike gradient-based algorithms, there is no learning rate or other tuneable parameters, making it easy to use and produce consistent results.

1.4.1.3 SOBI

Another method used to solve the ICA problem is second order blind identification (SOBI). Adapted from 'A blind source separation technique using second-order statistics' [9], this method utilizes the time coherence of the source signals. Unlike other methods like FastICA, the SOBI method relies solely on stationary second-order statistics based on a joint diagonalization of a set of covariance matrices. In this case, whitening the data is considered essential and part of the algorithm implementation. This formulation assumes that the source signals $s(t)$ is either a deterministic ergodic sequence or a stationary multivariate process, with $E[s(t+\tau)s(t)^*] = \text{diag}[\rho_1(\tau), \ldots, \rho_n(\tau)]$, where $(\cdot)^*$ denotes the conjugate transpose.

So we can define the autocovariance of $s_i(t)$ as

$$\rho_i(\tau) = E[s_i(t+\tau)s_i^*(t)], 1 \leq i \leq n. \tag{1.48}$$

With this we have the covariance matrices of X as

$$R(\tau) = E[x(t+\tau)x^*(t)] = AR_s(\tau)A^H \tag{1.49}$$

with $(\cdot)^H$ the complex conjugate transpose of a matrix.

We now consider the whitened covariance matrices, $\overline{R}(\tau) = ZR(\tau)Z^H$. We see $\overline{R}(\tau)$ are simply the covariance matrices of the whitened data X. We also note the result that for any whitening matrix Z there exists a unitary matrix U, that is $UU^H = U^HU = I$, such that $ZA = U$. Thus, we can also see A can be found by,

$$A = Z^{\#}U \tag{1.50}$$

where $(\cdot)^{\#}$ is the Moore-Penrose pseudoinverse. Therefore, by combining these results and Equation 1.49 we see

$$\overline{R}(\tau) = UR_s(\tau)U^H. \tag{1.51}$$

Since U is unitary and $R_s(\tau)$ is diagonal, a whitened covariance matrix is diagonalized by the unitary transform, U, which can be found as a unitary diagonalizing matrix of $\overline{R}(\tau)$, for some lag τ.

Next we briefly introduce the notion of joint diagonalization. We define the Frobenius norm for off diagonal elements of matrix M,

$$\text{off}(M) = \sum_{1 \leq i \neq j \leq n} |M_{ij}|^2 , \tag{1.52}$$

where the unitary diagonalization of M is equivalent to zeroing $\text{off}(V^H M V)$ for some unitary matrix V. So, if $M = UDU^H$, with D a diagonal matrix with distinct elements, setting $\text{off}(V^H M V) = 0$, we have $V = U$. We can now consider the set $\mathcal{M} = \{M_1, \ldots, M_K\}$, a unitary matrix is considered to be a joint diagonalizer of $\mathcal{M}$ if it minimizes the non-negative function

$$\mathfrak{C}(\mathcal{M}, V) = \sum_{k=1}^{K} \text{off}(V^H M_k V). \tag{1.53}$$

So, if each $M \in \mathcal{M}$ can be unitarily diagonalized by U, $M_k = UD_kU^H$, then U is a joint diagonalizer of $\mathcal{M}$ and can be proven to be unique. Computationally, a joint diagonalizer can be found through a generalization of the Jacobi method for diagonalization of a single Hermitian matrix [44].

With these notions in place we can now outline the SOBI algorithm [9].

Algorithm SOBI

1: $\widehat{R}(0) \leftarrow$ Estimate of $R(0)$ from T samples
2: $\lambda_1, \ldots, \lambda_n \leftarrow n$ largest eigenvalues of $\widehat{R}(0)$
3: $h_1, \ldots, h_n \leftarrow n$ corresponding eigenvectors of $\widehat{R}(0)$
4: $\widehat{\sigma}^2 \leftarrow$ Average of the $m - n$ smallest eigenvalues of $\widehat{R}(0)$
5: $\widehat{Z} \leftarrow [(\lambda_1 - \widehat{\sigma}^2)^{-(1/2)} h_1, \ldots, (\lambda_n - \widehat{\sigma}^2)^{-(1/2)} h_n]^H$
6: $X_{whitened} \leftarrow \widehat{Z}X$
7: **for** $\tau \in \{\tau_j | j = 1, \ldots, K\}$ **do** $\widehat{\overline{R}}(\tau) \leftarrow E[X_{whitened}(t + \tau) X_{whitened}(t)]$
8: $\widehat{U} \leftarrow$ Joint diagonalizer of set $\{\widehat{\overline{R}}(\tau_j) | j = 1, \ldots, K\}$
9: $\widehat{A} \leftarrow \widehat{Z}^{\#}\widehat{U}$
10: $\widehat{s}(t) \leftarrow \widehat{U}^H \widehat{W} x(t)$

where for line 4 comes from the white noise assumption and $R(0)$ is an $m \times n$ matrix.

Overall, the SOBI algorithm begins by estimating the sample covariance $R(0)$ along with its eigenvalues and corresponding eigenvectors. Using these and estimated noise variance we estimate the whitening matrix Z. The unitary matrix U is determined as the joint diagonalizer of a set of estimates of whitened covariance matrices for fixed time lags. Finally, with estimates of U and Z we can compute the mixing matrix, A, and subsequently the source signals, $s(t)$, by Equation 1.50 and Equation 1.27, respectively. SOBI's implementation has been found outperform other ICA solution methods in terms of speed and accuracy [118].

1.4.2 H-infinity filtering

We investigate two types of noise that commonly affect EEG recordings. The first is the most commonly researched type, ocular artifacts, which generally come from eye movements and blinking. The second is motion artifacts that are generally more present when walking or performing other physical tasks. H^∞ filtering is a

linear adaptive filtering method that guarantees robustness where small modeling errors and external noise, such as ocular artifacts, do not cause large estimate errors. However, when linear models do not handle certain types of artifacts well, such as motion, a common non-linear representation of dynamic systems known as Volterra series can be used. So, we define the formulation of H^∞ filtering [52] applied to ocular artifact removal [73] as well as an extension of this formulation with Volterra series [89] to remove motion artifacts [72], adapted from these prior publications.

We begin with the following definition,

Definition 1.4.2. We let h_2 denote the vector space of square-summable complex-valued causal sequences with inner product $< \{f_k\}, \{g_k\} >= \sum_{k=0}^{\infty} f_k^* g_k$, where $(\cdot)^*$ denotes the complex conjugation. Let T be a transfer operator that maps an input sequence $\{u_i\}$ to an output sequence $\{y_i\}$. Then the H^∞ norm of T is defined as

$$\|T\|_\infty = \sup_{u \neq 0, u \in h_2} \frac{\|y\|_2}{\|u\|_2}, \tag{1.54}$$

where the notation $\|u\|_2$ denotes the h_2-norm of the causal sequence $\{u_k\}$, that is, $\|u\|_2^2 = \sum_{k=0}^{\infty} u_k^* u_k$.

For adaptive time-varying filtering, we assume that an output sequence $\{d_i\}$ follows a linear filter model

$$d_i = h_i^T w_i + v_i \tag{1.55}$$

where $h_i^T = [h_{i1}, \ldots, h_{in}]$ is the known input vector, w_i unknown filter time-varying weights we wish to estimate, and $\{v_i\}$ is an unknown disturbance sequence that

might include modelling errors. Further, we let $\widehat{w}_i = \mathcal{F}(d_0, \ldots, d_{i-1})$ denote the estimate of w_i given observations $\{d_j\}$ and $\{h_j\}$ from time zero up to and including time $i-1$, and define the output prediction error as

$$e_i = h_i^T w_i - h_i^T \widehat{x}_i. \tag{1.56}$$

We note that since the time-variation in x_i, $\delta x_i = x_{i+1} - x_i$, is unknown, it is also considered as a disturbance. So, for every estimator $\mathcal{F}$, there will be a transfer operator from disturbances $\{\mu^{-\frac{1}{2}}(x_0 - \widehat{x}_0), \{v_j\}_{j=0}^i, q^{-\frac{1}{2}}\{\delta x_j\}_{j=0}^i\}$, to the prediction errors $\{e_j\}_{j=0}^i$, where q is a positive constant that reflects a-priori knowledge of how quickly the weight vector w_i is time-varying ($q \approx 10^{-8}$ for slow signals). We denote this transfer operator as $T_{g,i}(\mathcal{F})$.

We can then formulate the problem of finding estimates of time-varying weights, $\widehat{w} = \mathcal{F}(d_0, \ldots, d_{i-1})$, such that the optimal H^∞ estimation method minimizes $\|T_{g,i}(\mathcal{F})\|_\infty$, and obtain the resulting

$$\gamma_g^2 = \inf_{\mathcal{F}} \|T_{g,i}(\mathcal{F})\|_\infty \tag{1.57}$$

$$= \inf_{\mathcal{F}} \sup_{u \neq 0, u \in h_2} \frac{\|e\|_2^2}{\mu^{-1}|x_0 - \widehat{x}_0|^2 + \|v\|_2^2 + q^{-1}\|\delta x\|_2^2}. \tag{1.58}$$

So, in context of EEG formulation for ocular noise (measured by electrooculography (EOG)) we let $d_i \equiv s_i$ be the measured primary signal and $h_i \equiv r_i$ be the

reference measurements from EOG electrodes which yield,

$$\widehat{w}_{i+1} = \widehat{w}_i + \frac{P_i r_i}{1+r_i^T P_i r_i} y_i \tag{1.59}$$

for which

$$y_i = s_i - d_i \tag{1.60}$$

with

$$d_i = r_i^T \widehat{w}_i, \tag{1.61}$$

and

$$P_i^{-1} = \tilde{P}_i^{-1} - \gamma^{-2} r_i r_i^T, \tag{1.62}$$

where

$$P_{i+1} = \left(\tilde{P}_i^{-1} + (1-\gamma^{-2}) r_i r_i^T\right)^{-1} + qI. \tag{1.63}$$

Here $\widehat{w}_{i+1}$ is the estimated weight per channel for sample $i+1$, r_i is the vector of reference measurements, s_i is the measured signal of interest, and $\tilde{P}_i$ is the noise convariance matrix initialized by $\tilde{P}_0 = \mu I$, with constant μ. Further, γ is the bound on the energy-to-energy gain from the disturbances to the output estimation error and is important for filter performance, since it defines the levels of disturbance that will be tolerated by the H^∞ filter. As γ gets closer to one, the filter behavior gets closer to the optimal filter behavior. In an optimal H^∞ filter $\gamma < 1$ provides a guaranteed robust performance under all levels of disturbances

that effects the system. In the above formulation the weights are assumed to be time-varying with unknown variation dynamics, and the change $\delta\widehat{w}$ is formulated as an unknown disturbance on the system. Hence the above formulation is valid for the inequality $\gamma^2 \leq 1 + q\overline{r}$ where $\overline{r}$ is the supremum of the measured disturbances [73].

Next, we define a second-order Volterra series of degree N extension of H^∞ filtering to model non-linear systems. We can redefine d_i from Equation 1.61

$$d_i = r_i^T w_i = \sum_{l_1=0}^{N} w_{l_1,i} r_{i-l_i} + \sum_{l_1=0}^{N} \sum_{l_2=l_1} w_{l_1 l_2,i} r_{i-l_1} r_{i-l_2} \tag{1.64}$$

using the linear H^∞ filtering formulation above. $w_{l_1,i}$ and $w_{l_2,i}$ represent the Volterra kernel identified through the time-varying H^∞ adaption rule.

In the implementation used on data in our work, filter order of $N = 3$ was used. Further, when applying the Volterra series extension for motion artifacts, the selection of the reference signal that is used to identify the motion artifacts in EEG signals is critically important [72]. 3-axis acceleration values, after gravity compensation using the quaternion of the IMU were used as the reference signal. As such, the 3-axis acceleration values for reference were

$$d_i = w_{1,i}\mathrm{acc}_{x,i} + w_{2,i}\mathrm{acc}_{y,i} + w_{3,i}\mathrm{acc}_{z,i}, \tag{1.65}$$

1.5 Kruskal-Wallis statistical test

In general, we use non-parametric statistical tests throughout this work. Parametric tests involve an underlying assumption that the data can be parameterized since it follows known distributions. In this case, estimates for these parameter can be computed for the data and be used in statistical tests. An example of this is data, z, from a Normal distribution which can parameterized by a mean, μ, and a variance, σ^2 can be written as $z \sim \mathcal{N}(\mu, \sigma^2)$. However, when data does not follow known distributions, we must apply non-parametric statistical methods. The majority of the these methods rely on the ranks of values of the data so to use the data directly rather than parameterize it. In order to determine whether a parametric or non-parametric test should be used there are several tests for determining whether data is distributed normally. Research has shown that through Monte Carlo simulation, that the Shapiro-Wilk test has the best power for a given significance when compared to the Anderson-Darling, Kolmogrov-Smirnov, and Lilliefors tests [113]. Thus, we use the Shapiro-Wilk test when evaluating the normality assumptions of data.

In particular, throughout our work we use the Kruskal-Wallis test as a non-parametric statistical test. The Kruskal-Wallis test is meant to compare the distributions of scores on quantitative variables obtained from two or more groups [77]. It can therefore be thought of as an analysis of variance (ANOVA) for independent samples. However, unlike an ANOVA, it is used when the data are not clearly normally distributed, the dependent variables are ordinally measured, or are collected

from too small of a sample size.

The null hypothesis, H_0, for this test is that the populations represented by the conditions, groups or samples, have the same median of scores on the quantitative response variable and thus distributed similarly. If we were to reject the null hypothesis and accept the alternative hypothesis, H_1, then that would imply that the populations' medians differ statistically.

Given some data, $D = \{x, y\}$, of length N, where each data point x is assigned a group, y, with G total groups. Each x is given a rank starting at 1 corresponding to its value in ascending order. Tied values receive a rank of the average across the ranks they occupy. For each group we compute the square of the sum of all ranks in that group, r_g^2, and the number of data points in each group, n_g. It should be the case that D has $\sum_{g=1}^{G} n_g = N$ data samples. To use this method, the test statistic, H, is computed by,

$$H = \left(\frac{12}{N \cdot (N+1)}\right)\left(\sum_{g=1}^{G} \frac{r_g^2}{n_g}\right) - 3(N+1). \tag{1.66}$$

χ^2 (chi-square) tests, such as the Kruskal-Wallis test, are often used to determine whether there is a significant difference between the expected frequencies and the observed frequencies in one or more categories. The test statistics used in these hypothesis tests generally follow a χ^2 distribution so long as the number of samples is greater than five for each group [79]. Therefore, the sampling distribution of H approximates that of a χ^2 distribution, particularly when there are five or more data points per group, $n_g \geq 5, \forall g \in \{1, \ldots, G\}$. So, H can be compared to the critical

χ^2 distribution value to test H_0. With degrees of freedom, $df = \min_{g \in G}\{n_g\} - 1$, and a $p-$value, we can determine $H_{\text{critical}} = \chi^2(df, p)$.

Finally, if $H \leq H_{\text{critical}}$ then we fail to reject H_0. Otherwise, if $H > H_{\text{critical}}$ then we reject H_0 and accept H_1.

If $G > 2$, and H_0 is rejected, pairwise comparisons may be needed to determine if any pairs of groups fail to reject H_0.

1.6 Density estimation

When we have data but wish to estimate a model that represents the distribution of the data, kernel density estimation (KDE) presents a non-parametric method to do so.

The following formulation of KDE was first presented both by E. Rosenblatt and E. Parzen [116], [103]. For $\{x_1, x_2, \ldots, x_N\}$ which is assumed are independent and identically distributed data sampled from a distribution, we begin by formulating an estimator of the cumulative density function (CDF), $F(x) = P[X \leq x]$. So we can estimate $F(x)$ by $\widehat{F}_n(x) = \frac{1}{n} \cdot \{\text{number of } x_i \leq x\}$. This can also be considered the frequency of occurrence estimator of $F(x)$.

From this we can formulate the estimation of the probability density distribution (PDF), $f(x)$. By definition we have $f(x) = \frac{\mathrm{d}}{\mathrm{d}x}F(x)$. So, we see from the fundamental theorem of calculus we can estimate $f_n(x)$ by

$$\widehat{f}_n(x) = \frac{F_n(x+h) - F_n(x-h)}{2h} \tag{1.67}$$

with h a small positive width. By plugging in our $F_n(x)$ we see

$$\widehat{f}_n(x) = \frac{1}{2nh} \cdot \{\text{number of } (x_1, \ldots, x_n) \in [x-h, x+h]\} \tag{1.68}$$

We can define

$$K(u) = \begin{cases} 1/2, & \text{if } |u| \leq 1 \\ 0, & \text{otherwise.} \end{cases} \tag{1.69}$$

Then, we have from Equation 1.68,

$$\widehat{f}(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \tag{1.70}$$

This result is known as a uniform KDE since $K(u)$ is the PDF of a uniform distribution. However, other kernels, K, can be used to estimate the distributions. In particular, we use a Gaussian kernel,

$$K(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right) \tag{1.71}$$

which satisfies the conditions that $\int_{-\infty}^{\infty} K(x)\mathrm{d}x = 1$ and $K(x) = K(-x)$. We further chose the bandwidth, h, the size of the kernel at each point, through Scott's Rule, $h = n^{-1/(d+4)}$, with n the number of data points and d the dimensionality of the data [34]. The bandwidth is a smoothing parameter that controls the trade-off between bias and variance in the resulting distribution. A larger bandwidth leads to a smoother density distribution with higher bias, while a smaller bandwidth leads

to less smooth density distribution with higher variance. Finally, we note that as $h \to 0$, the kernel approaches the Dirac delta function and $\widehat{f}(x)$ approaches the exact density of the data which would be a set of impulses located at each point.

1.7 Machine learning

Several of our works use machine learning methods. This section introduces some notions in machine learning and detail specific methods used in our research. Under the assumption that similar data is represented spatially close to each other, the simplest case in machine learning is that the data be clustered into appropriate groups based on its labels. Simpler methods that aim to solve this type of problem include the K-means clustering algorithm or Gaussian mixture models (GMM), which are equivalent when all variances in all dimensions are equal [47]. A large draw back of these methods however is that as the dimensionality of the data increases they do not tend to perform as well. Thus, other methods are explored in more depth. The main feature based method we use is kernel support vector machines (kSVM), which has been adapted from 'A Tutorial on Support Vector Machines for Pattern Recognition' [19]. Finally, a brief introduction, motivated by example, and some of the methods used in deep convolutional neural networks (DCNN) are adapted from the relevant parts of 'Deep Learning', which expands on these methods in much more detail [47].

In the case of machine learning application in these works, we mostly consider labeled, supervised machine learning methods. That is, consider datasets such that

$\{X_i, y_i\}$ where X is the data and y the data's label. Data was generally be separated into training and testing, sometimes called validation, sets. Machine learning algorithms and parameters are computed solely on the training data subset. The method's efficacy is determined by the classification accuracies of estimating y_i, the labels, of the testing subset and comparing these estimates to the true labels.

1.7.1 Support vector machines

The theory and basis of SVMs has been introduced some time ago [139]. However, it is still prevalent and used as a feature-based classification method because of its efficacy. This topic requires a great deal of general and convex optimization methods. Exact solutions methods and derivations of these optimizations are not presented here [17], [38].

Training data is labeled as $\{X_i, y_i\}, i = 1, \ldots, \ell, y_i \in \{-1, 1\}, X_i \in \mathbb{R}^d$. Now suppose there exists a hyperplane which separates the positive and negative labeled data, either $+1$ for one class, or -1 for the other class. The points X which lie on the hyperplane satisfy $\mathbf{w} \cdot X + b = 0$, where $\mathbf{w}$ is the normal to the hyperplane, $|b|/\|\mathbf{w}\|$ is the perpendicular distance from the hyperplane to the origin, and $\|\mathbf{w}\|$ the Euclidean norm of $\mathbf{w}$. We now let d^+ and d^- be the shortest distance from the separating hyperplane to the closest positive and negative training samples, respectively. The margin for the separating hyperplane can be defined as $d^+ + d^-$. So in the separable case, all the training data will satisfy the following equations,

$$X_i \cdot \mathbf{w} + b \geq +1, y_i = +1 \tag{1.72}$$

$$X_i \cdot \mathbf{w} + b \leq -1, y_i = -1 \tag{1.73}$$

and combining these inequalities,

$$y_i(X_i \cdot \mathbf{w} + b) - 1 \geq 0 \ \forall \ i \tag{1.74}$$

We can now consider points that satisfy equality in Equation 1.72. These points would lie on hyperplane $H_1 : X_i \cdot \mathbf{w} + b = 1$ with normal $\mathbf{w}$ and distance from the origin $|1-b|/\|\mathbf{w}\|$. Similarly, we can consider points that satisfy equality in Equation 1.73. These points would lie on hyperplane $H_2 : X_i \cdot \mathbf{w} + b = -1$ with normal $\mathbf{w}$ and distance from the origin $|-1-b|/\|\mathbf{w}\|$. So, $d^+ = d^- = 1/\|\mathbf{w}\|$ and the distance between these two hyperplanes, called the margin, is $2/\|\mathbf{w}\|$. Since H_1 and H_2 are parallel, with the same normal, and no training points between them, the pair of hyperplanes can be found to give the maximum margin by minimizing $\|\mathbf{w}\|^2$, subject to Equation 1.74.

Since this is a constrained optimization problem, the SVM optimization problem can be stated in a Lagragian formulation. So for each inequality constraint from Equation 1.74, we let $\alpha_i, i = 1, \ldots, \ell$ be Lagrange multipliers. For equality constraints, the Lagrange multipliers are unconstrained. However, with the inequality constraint, $c_i \geq 0$, constraint equations are multiplied by positive Lagrange multipliers and subsequently subtracted by the objective function. So,

$$L_P \equiv \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{\ell} \alpha_i y_i (X_i \cdot \mathbf{w} + b) + \sum_{i=1}^{\ell} \alpha_i. \tag{1.75}$$

The primal optimization formulation, L_P, can now be solved by minimizing L_P with respect $\mathbf{w}$, b. The solution will also require the gradients with respect to all α_i vanish and $\alpha_i \geq 0$.

Since this is a convex quadratic programming problem, we can equivalently formulate a dual optimization problem, L_D. Maximizing L_D subject to where the gradient of L_D vanishes with respect to $\mathbf{w}$, b, and $\alpha_i \geq 0$. The specific dual formulated here is known as the Wolfe dual. So, with the following conditions,

$$\mathbf{w} = \sum_i \alpha_i y_i X_i \tag{1.76}$$

$$\sum_i \alpha_i y_i = 0 \tag{1.77}$$

we can substitute these equality conditions into Equation 1.75, and yield,

$$L_D = \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j (X_i \cdot X_j) \tag{1.78}$$

If we let $b = 0$, thus requiring the hyperplanes to contain the origin and omitting condition from Equation 1.77 in higher dimensions, this reduces the degrees of freedom by one. Further, it is worth noting that these formulations of the SVM optimization problem satisfy the Karush-Kuhn-Tucker (KKT) conditions [38].

The data analyzed however does not always follow a linearly separable case. Thus, the constraints from Equations 1.72 and 1.73 need to be relaxed when necessary otherwise no solution will be found. To do this we introduce slack variables,

$\zeta_i = 1, \ldots, \ell$. So, the constraints become

$$X_i \cdot \mathbf{w} + b \geq +1 - \zeta_i, y_i = +1 \tag{1.79}$$

$$X_i \cdot \mathbf{w} + b \leq -1 + \zeta_i, y_i = -1 \tag{1.80}$$

$$\zeta_i \geq 0 \forall i \tag{1.81}$$

When an error occurs, the associated $\zeta_i > 1$. Thus, $\sum_i \zeta_i$ is the upper bound on the number of training errors. So, the objective function is changed to assign an extra cost to these errors such that,

$$\frac{1}{2} \|\mathbf{w}\|^2 \rightarrow \frac{1}{2} \|\mathbf{w}\|^2 + C \left(\sum_i \zeta_i \right)^k \tag{1.82}$$

with C is a now a parameter.

A larger C corresponds to a higher penalty for errors [25]. We note that the optimization problem has several nice desired properties depending on the value of k. The problem remains a convex optimization problem with any positive k, a quadratic optimization problem for $k = 1, 2$, and the slack variable ζ_i and their Lagrange multipliers do not appear in the dual formulation for $k = 1$, where we

have

$$\max L_D \equiv \quad \sum_i \alpha_i - \tfrac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j (X_i \cdot X_j) \tag{1.83}$$

$$\text{Subject to:} \qquad 0 \leq \alpha_i \leq C \tag{1.84}$$

$$\text{and} \qquad \sum_i \alpha_i y_i = 0. \tag{1.85}$$

So the solution is given by,

$$\mathbf{w} = \sum_{i=1}^{N_s} \alpha_i y_i X_i \tag{1.86}$$

where N_s is the number of support vectors. The main difference between the separable case here is that the α_i have an upper bound of C.

The primal formulation still satisfies the KKT conditions and can be written as

$$\min L_p = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_i \zeta_i - \sum_i \alpha_i \{y_i (X_i \cdot \mathbf{w} + b) - 1 + \zeta_i\} - \sum_i \mu_i \zeta_i \tag{1.87}$$

where μ_i are the Lagrangian multipliers applied to ensure $\zeta_i \geq 0$.

Finally, to extend this method even further to non-linear separation and hyperplanes we note that the data only appears as a dot product in Equations 1.83, 1.84, and 1.85. If the data is mapped to some Euclidean space, $\mathcal{H}$, $\Phi : \mathbb{R}^d \to \mathcal{H}$, then the training would only depend on $\Phi(X_i) \cdot \Phi(X_j)$ in $\mathcal{H}$. Further, allowing for a kernel function $K(X_i, X_j) = \Phi(X_i) \cdot \Phi(X_j)$, would allow the training procedure to only depend on K. Thus, the optimization will find a linear separation in a different

space [15]. Thus, the final SVM problem to solve is,

$$\max_{\alpha_i \geq 0} \quad \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j K(X_i, X_j) \tag{1.88}$$

$$\text{Subject to:} \qquad 0 \leq \alpha_i \leq C \tag{1.89}$$

$$\text{and} \qquad \sum_i \alpha_i y_i = 0 \tag{1.90}$$

where K must satisfy Mercer's Condition such that there exists Φ [140], with expansion

$$K(x, y) = \sum_i \Phi(x)_i \Phi(y)_i \tag{1.91}$$

if and only if for any $g(x) \in L^2$,

$$\int_{\mathbb{R}^2} K(x, y) g(x) g(y) \mathrm{d}x \mathrm{d}y \geq 0. \tag{1.92}$$

Kernel functions investigated here include

Polynomial Kernel:

$$K(x, y) = (x \cdot y + 1)^p \tag{1.93}$$

Radial Basis Function (RBF) Kernel:

$$K(x, y) = e^{-\|x-y\|^2/2\sigma^2} = e^{-\gamma \|x-y\|^2} \tag{1.94}$$

Kernel parameters have a significant effect on the decision boundaries learned during training. The degree of the polynomial kernel and the width parameter of

the Gaussian kernel control the flexibility of the resulting classifier. The lowest degree polynomial, $p = 1$, turns out to be the linear kernel, which may not perform well when non-linear relationships between features exist. For the Gaussian kernel, this expression is essentially zero if the distance between x and y is much larger than $1/\sqrt{\gamma}$. That is, for a fixed x it is restricted to a specific region around y. The support vector expansion, Equation 1.95, is a sum of Gaussian bumps centered around each support vector. When γ is small a given data point x has a non-zero kernel value relative to other support vectors. Therefore, the whole set of support vectors affects the value of the discriminant function at x. This results in a smooth decision boundary. As γ is increased the support vector's neighborhood increases, which leads to more curvature in the decision boundary. When γ is large the RBF γ parameter is more likely to lead the classifier to overfitting the data. The degree of the polynomial kernel, p, and the Gaussian kernel parameter, γ, determine the flexibility of the resulting SVM in fitting the data. If this complexity parameter is too large, over-fitting will likely occur [124]. In general one can think of these kernels as a form of measuring similarity.

Further, there exists a choice of hyperparameter of the soft-margin constant, C. For a large value of C, a large penalty is assigned to errors and margin errors. That is, points closest to the hyperplane affect its orientation, resulting in a hyperplane that comes close to several other data points. When C is decreased, those points close to the hyperplane may become margin errors and the hyperplane's orientation would be changed, providing a much larger margin for the rest of the data.

Once the optimization problem is solved based on the training subset of the

data, it can be tested by assigning test data, X_t, to $sgn(\mathbf{w} \cdot X_t + b)$ in the linearly separable case and in the nonlinear case $sgn(f(X_t))$, with

$$f(X_t) = \sum_{i=1}^{N_s} \alpha_i y_i \Phi(s_i) \cdot \Phi(X_t) + b = \sum_{i=1}^{N_s} \alpha_i y_i K(s_i, X_t) + b \tag{1.95}$$

where s_i are the support vectors.

Generally, when applying kSVM to a set of features several transforms on the data can be applied to produce better results and normalize data if they are on different scales. In general a transform is also determined based solely on the training data and applied to both the training data and later the testing data.

Specifically, we apply max normalization to features used with kSVM in our classifications which we define here.

Definition 1.7.1. If we consider training data $X[n, i]$ and testing set $X_t[n, i]$, where $n \in \{1, \ldots, N\}$ are each data point and $i \in \{1, \ldots, M\}$ the sample index, then the max normalization transform is defined for the training set as,

$$X_{max}[n, i] = \frac{X[n, i] - \overline{X}_{\mu}[n]}{\overline{X}_{max}[n]}, \quad \forall n \in \{1, \ldots, N\}, i \in \{1, \ldots, M\} \tag{1.96}$$

and for the testing set as,

$$X_{t-max}[n, i] = \frac{X_t[n, i] - \overline{X}_{\mu}[n]}{\overline{X}_{max}[n]}, \quad \forall n \in \{1, \ldots, N\}, i \in \{1, \ldots, M\} \tag{1.97}$$

where

$$\overline{X}_{\mu}[n] = \frac{1}{M}\sum_{i=1}^{M} X[n,i], \quad \forall n \in \{1,\ldots,N\} \tag{1.98}$$

and

$$\overline{X}_{max}[n] = \max_{i\in\{1,\ldots,M\}} (X[n,i]), \quad \forall n \in \{1,\ldots,N\} \tag{1.99}$$

In this case the transform values for each data point, $\overline{X}_{\mu}[n]$ and $\overline{X}_{max}[n]$, are determined by the samples from the training set only. This transform rescales all features to be $[0,1]$, allowing for them all to be weighted more equally during training since different scales and magnitudes across features can weight importance higher values during training kSVM models.

1.7.2 Deep convolutional neural networks

The field of neural networks, particularly in the context of machine learning and signal classification has a long history. Full derivations of each element of modern day DCNN are not presented here. Rather, by example, we consider a DCNN used in our works and expand upon its elements and layers shown in Figure 1.5, and learning algorithm used.

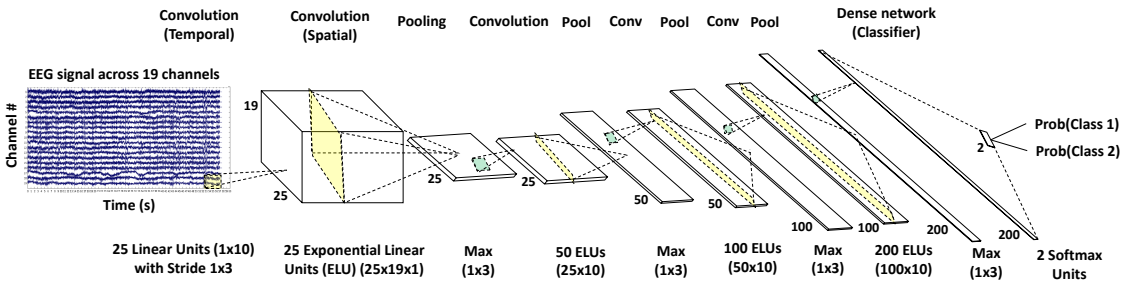


Figure 1.5 Deep convolutional neural network architecture. Convolutions are shown in yellow and all have stride 1x3, across the temporal dimension. Max pool layers are shown in green. The classification is determined by the class with the highest output probability.

Our work focused on feed-forward neural networks which do not include feedback within the network where the outputs of the model are used again as inputs. in essence are a collection of nodes that have the following properties.

The basic building block of neural networks consist of nodes with weighted connections. They are biologically inspired by neurons with their connections and activations by simulating neuronal connections and firings. Traditional neural networks would simply consist of an input layer with nodes equal to the size of the input space and an output layer equal to the number of classes of interest. The goal of a neural network is for weights of each node to be assigned in a manner that makes the network behave as a function that separates the classes. However, it was quickly understood that more complicated functions could not be estimated with these networks. The classical example of these simple networks failure can be seen in non-linearly separable cases, such as the exclusive or (XOR). Thus a hidden layer was added between the input and output layers. Further, simply having multiple layers is not quite enough since it is clear that he composition of two linearly op-

eration can be reduced to a single linear operation. So, each layer also includes a non-linearity. In general, most neural networks at each layer apply an affine transformation controlled by learned parameters followed by a non-linear function applied to the transformed input, also known as an activation function. Combining these components, the universal approximation theorem can be stated [55], [28].

Definition 1.7.2. The Universal Approximation Theorem states that a Feedforward network with linear output units and at least one hidden layer with a squashing non-linear activation function can approximate any continuous function on a closed and bounded subset of $\mathbb{R}^n$, also known as Borel measurable, from one finite dimensional space to another with any desired non-zero error given that the network has enough hidden units [47].

Though in theory only these three layers are needed, in practice it has been found that adding more layers and increasing the depth of the network rather than increasing the width of each layer has performed well.

1.7.2.1 Convolutional layers

Rather than using only fully connected layers, convolutional layers have recently become widely used since they allow for weights in filters to be learned. These embedded filters act as feature extractors that can then fed into traditional architectures for classification. Similar to the one dimensional convolution, the two dimensional discrete convolution used here can be stated as,

Definition 1.7.3. For signal I and convolution kernel K, the two dimensional

convolution is defined as,

$$(I * K)[i, j] = \sum_{m \in \mathbb{N}} \sum_{n \in \mathbb{N}} I[m, n] K[i - m, j - n] \tag{1.100}$$

The main motivation for using this convolutions stem from having sparse connectivity, share parameters, and is translation invariant, that is the output of the operation preserves translations applied to the input. Unlike dense neural networks that connect each node in layers to each other, using kernels, which are usually much smaller than the input dimension, allows for sparse connectivity. The kernel is still able to connect to the input by translating, or striding through the input, producing an output at each step. This notion of the same kernel striding through the input data is also what gives rise to the shared parameters of convolution kernels across the input. Finally, translation invariance comes from the fact that if a translation were to be applied to I before being convolved with K that it would in fact be equivalent to applying that same translation to $(I * K)$.

1.7.2.2 Activation functions and non-linearities

The rectified linear Unit (ReLU) function, $ReLU(x) = \max(0, x)$, is commonly used since it yields a non-linear transform needed but it is close to linear since it is based on piecewise linear functions with two linear pieces. Since it is nearly linear it preserves many of the properties that make optimizing linear functions easier with gradient based methods. An extension to the ReLU non-linearity is the exponential linear unit (ELU) [23].

Definition 1.7.4. The Exponential Linear Unit (ELU) non-linearity function is defined as,

$$ELU(x) = \begin{cases} \alpha(e^x - 1) & , x < 0 \\ x & , x \geq 0 \end{cases} \tag{1.101}$$

Ideally, as a non-linearity function is fed into the back-propagation algorithm, the derivative should have low computation cost. As such,

$$\frac{\partial}{\partial x} ELU(x) = \begin{cases} \alpha e^x & , x < 0 \\ 1 & , x \geq 0 \end{cases} \tag{1.102}$$

or alternatively, adding and subtracting α, $\alpha e^x - \alpha + \alpha = ELU(x) + \alpha$, so

$$\frac{\partial}{\partial x} ELU(x) = \begin{cases} ELU(x) + \alpha & , x < 0 \\ 1 & , x \geq 0 \end{cases} \tag{1.103}$$

ELU is a strong alternative to the commonly used ReLU function since it can generate negative outputs and smooth slowly until its output equals $-\alpha$.

1.7.2.3 Pooling

A pooling function replaces the input of a location with a summative operation of that location and other nearby outputs. For example, the max pooling operation finds the maximum value within a rectangular neighborhood [147]. In all cases, pooling helps to make the representation become approximately invariant to

small translations of the network input. Invariance to translation means that if we translate the network input by a small amount, the values of most of the pooled outputs will not change. Invariance to small, local, translation is important if the presence of a feature is more important than its location. Alternatively, pooling can be viewed as adding an infinitely strong prior that the mapping function, convolution, or dense layer learns must be invariant to small translations. When this is true, it can greatly improve the network's performance. In general, pooling over spatial regions produces a translation invariant functions. However, if we pool over the inputs of differently oriented convolutions, the features can learn which small transformations to become invariant to. Similar to the convolution kernels, pooling operators have a specified size, usually much smaller than the input size, which they stride across the input giving a result at each step.

1.7.2.4 Dropout

To consider dropout, we first note that it finds theoretical motivation in bootstrap aggregating [131]. Bootstrap aggregating looks to reduce generalization error in neural networks by combining several separately trained models. The idea is that this ensemble methods will use several models that are unlikely to generate the same errors on the test set. Dropout training is not quite the same as ensemble training. In the case of bootstrap aggregating, the models are all independently trained. On the other hand, in the case of dropout, the models share parameters with each other, inheriting a different subset of parameters from the different neural

networks. However, this parameter sharing makes it possible to represent an exponential number of models with dropout. The principle is to train on sub-models that do not include the outputs. Dropout trains the ensemble consisting of all sub-networks that can be formed by removing non-output units from the sub-networks. Training with dropout usually makes use of a minibatch-based learning algorithm that makes small steps, such as stochastic gradient descent. For each minibatch, a random sample of a different mask is applied to all of the input and hidden units of the network. The mask for each unit is sampled independently from all other units. The probability of sampling a mask value of one, where a unit is included, is a hyperparameter of dropout. Once the sub-network has been generated after applying the binary mask, forward propagation, back-propagation, and the learning update are performed as usual.

1.7.2.5 Batch normalization

With the increased complexity of deep learning models and the amount of data used to train them, training a full deep network can take a significant amount of time. Batch normalization is a method of normalizing internal convariates of layers so that higher learning rates can be used to potentially reduce the number of training steps and training time required [61]. It applies adaptive re-parametrization, again motivated by the difficulty of training very deep models that involve the composition of several functions or layers. The gradient and learning algorithms inform the network on how to update each parameter given that the other layers do not change.

In practice, all of the layers are updated simultaneously. We let H, structured as a design matrix, be a minibatch of activations of the layer to normalize, with the activations for each example in a row of H. H is normalized by replacing it with $H' = \frac{H-\mu}{\sigma}$, where μ is a vector containing the mean of each unit and σ is a vector containing the standard deviation of each unit. Further, as with all other functions applied to the input, the back-propagation learning algorithm is also applied to this transformation so that the gradient will never increase the mean or standard deviation. During training, μ and σ may be replaced by running averages that allow for the model to be evaluated on a single example, without needing definitions of μ and σ that depend on an entire minibatch. In convolutional networks it is important to apply the same normalizing μ and σ within a feature map so that the statistics of the feature map remain the same regardless of spatial location.

1.7.2.6 Objective functions

If we simply wished to minimize the distance between the network output and target output values, a least squares cost function may be appropriate. However, since in most cases the network is learning the distribution for $p(y|x, \theta)$ for output y given data x and learned parameters θ, we can use the cross-entropy cost function to evaluate errors between the training data and the model's predictions. This in essence is solving for the maximum likelihood. The cost function can be understood as the negative log-likelihood, or equivalently the cross-entropy, between training data and model's output.

Definition 1.7.5. The cross-entropy cost function is defined as,

$$J(\theta) = -E_{x,y\sim p_{data}} \log p_{model}(y|x) \tag{1.104}$$

For two classes, it would be possible to use a classic activation function on the last layer,

Definition 1.7.6. The Sigmoid function is defined as,

$$\sigma(x) = \frac{1}{1+e^{-x}} \tag{1.105}$$

However, to construct a probability distribution over several binary variables we can use the Softmax function,

Definition 1.7.7. The Softmax function is defined as,

$$\text{softmax}(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{C} e^{z_j}} \tag{1.106}$$

where $z = x - \max_i(x_i)$ is usually computed for numerical reasons to avoid overflow issues in computation. Further, C denotes the number of classes of interest.

This is a useful function when dealing with probabilities since $\text{softmax}(x) \in (0, 1)$. Further, the Sigmoid function saturates with large positive or negative numbers allowing for a flat function when its input is introduced to small changes. Thus the Softmax cross-entropy loss function for two class problems investigated in this

work can be formulated as,

$$\min_{y} L(y) = \frac{1}{N} \sum_{i=1}^{N} c_i \log(y_i) + (1 - c_i) \log(1 - y_i) \tag{1.107}$$

where $y = \frac{e^{out_c}}{\sum_{d=1}^{C} e^{out_d}}$, and c_i the class label for each input.

1.7.2.7 Back-propagation

The back-propagation algorithm is in essence a recursive application of the chain rule of derivatives in calculus. In general can be written as,

$$\frac{\partial z}{\partial x_i} = \sum_{j} \frac{\partial z}{\partial y_j} \cdot \frac{\partial y_j}{\partial x_i} \tag{1.108}$$

We can recursively apply the chain rule to obtain back-propagation. In the case of the neural network used (1.5), we have the function

$$Out = \text{softmax}(ELU(C_4(ELU(C_3(ELU(C_2(ELU(C_1(x)))))))))) \tag{1.109}$$

So, to learn the appropriate weights, w we wish to compute $\frac{\partial Output}{\partial w_i}$. We let $y = ELU(C_1(x))$, $z = ELU(C_2(y))$, $v = ELU(C_3(z))$, $k = ELU(C_4(v))$, $Out = softmax(k)$). When expanded with these terms is,

$$\frac{\partial Out}{\partial w_i} = \frac{\partial Out}{\partial k} \frac{\partial k}{\partial v} \frac{\partial v}{\partial z} \frac{\partial z}{\partial y} \frac{\partial y}{\partial w} \tag{1.110}$$

For each weight in the network, the particular path can be traced and used in

the computation for the chain rule and back-propagation. [47]

1.7.2.8 Learning algorithms

In order to find the optimal value to solve the objective function we look to solve, for a general objective function $f(x)$, $x_{obj} = argmin f(x)$. Through gradient based optimization, we see from the basic derivative that it tells us how to change x in order to make small changes in $f(x)$. We know that $f(x - \epsilon \cdot \text{sign}(f'(x))) < f(x)$ for a small enough ϵ, that is, we can reduce $f(x)$ by moving x in small steps in the opposite sign of the derivative. This leads to gradient decent in the simplest case. This method finds local critical points, such as a minimum, maximum, or saddle point. In general $x_n = x_{n-1} - \epsilon \nabla_x f(x)$ where we call ϵ the learning rate. An extension of this that is widely used is stochastic gradient descent (SDG). The insight that led to this algorithm is to consider the gradient as an expectation. A set of samples, a minibatch, of size m, is drawn from the training set. The gradient is then computed by $g = \frac{1}{m} \nabla_\theta \sum_{i=1}^{m} L(x^{(i)}, y^{(i)}, \theta)$ and parameters are updated by $\theta \leftarrow \theta - \epsilon g$. Unlike convex function optimization, these non-convex functions can be optimized but without convergence guarantees. We can also add momentum to this learning method to accelerate learning. The momentum algorithm accumulates an exponentially decaying moving average of past gradients and continues to move in their current direction. It introduces v, the velocity, and its effect is determined by the hyper-parameter $\alpha \in [0, 1]$, where the larger α is relative to ϵ, the more previous gradients affect the current gradient direction. The gradient update is then adjusted

to be

$$v \leftarrow \quad \alpha v - \epsilon \nabla_\theta \left(\frac{1}{m} \sum_{i=1}^{m} L(f(x^{(i)}; \theta), y^{(i)} \right) \tag{1.111}$$

$$\theta \leftarrow \qquad \theta + v \tag{1.112}$$

where velocity, v, accumulates the gradient elements $\nabla_\theta \left(\frac{1}{m} \sum_{i=1}^{m} L(f(x^{(i)}; \theta), y^{(i)} \right)$.

Specifically, the Adam learning algorithm, whose name comes from adaptive momentum estimation, uses these methods and is a widely used method, including in our DCCN model [74]. Adam combines benefits from two other extensions to SGD, namely adaptive gradient algorithm (AdaGrad) and root mean square propagation (RMSProp). Momentum is used as an estimate of the mean with exponential weighting of the gradient. Specifically, the algorithm calculates an exponential moving average of the gradient and the squared gradient, and the parameters control the decay rates of these moving averages. Further, Adam includes bias corrections to both the means, uncentered variance, and the momentum, to account for the bias of these moment estimates towards zero, which are calculated after calculating the biased estimates.

1.7.2.9 Architectures and layers

The two dimensional convolution can be further extended from the one defined previously. This is considered an affine transformation. The output of which is led to a non-linearity. For the pooling layer is important when we care more about whether some feature is present rather than where exactly it is.

A typical layer of a convolution network consists of three stages. In the first stage, the layer performs several convolutions in parallel to produce a set of linear feature maps. In the second stage, each feature map is run through a nonlinear activation function, such as the ELU function. This stage is sometimes called the detector stage. In the third stage, we use a pooling function to modify the output of the layer further, for instance to enforce local translation invariance. In addition to these classical three stages, modern deep networks have also began to apply methods at each of these stages to improve learning further, including dropout and batch normalization.

Finally, after several of these three stage layers, depending of the depth of the network, the output can be fed into a dense layer that act as a classifier on the features that have been identified and computed by the three stage convolution layers. In general, the loss function acts as as non-linearity at this stage as well. In the case of multinomial outputs, it is desired that the outputs be probability-like values. To show each of the methods applied in the deep network, Figure 1.6 shows the *Sequential* model used in the implementation of the DCNN model in Figure 1.5. Specific layer parameters not discussed in this section can be found in the PyTorch documentation [104].

```
Model:
Sequential(
(dimshuffle): Expression(expression=_transpose_time_to_spat)
(conv_time): Conv2d (1, 25, kernel_size=(10, 1), stride=(1, 1))
(conv_spat): Conv2d (25, 25, kernel_size=(1, 19), stride=(1, 1), bias=False)
(bnorm): BatchNorm2d(25, eps=1e-05, momentum=0.1, affine=True)
(conv_nonlin): Expression(expression=elu)
(pool): MaxPool2d(kernel_size=(3, 1), stride=(1, 1), dilation=(3, 1))
(pool_nonlin): Expression(expression=identity)
(drop_2): Dropout(p=0.5)
(conv_2): Conv2d (25, 50, kernel_size=(10, 1), stride=(1, 1), dilation=(3, 1), bias=False)
(bnorm_2): BatchNorm2d(50, eps=1e-05, momentum=0.1, affine=True)
(nonlin_2): Expression(expression=elu)
(pool_2): MaxPool2d(kernel_size=(3, 1), stride=(1, 1), dilation=(9, 1))
(pool_nonlin_2): Expression(expression=identity)
(drop_3): Dropout(p=0.5)
(conv_3): Conv2d (50, 100, kernel_size=(10, 1), stride=(1, 1), dilation=(9, 1), bias=False)
(bnorm_3): BatchNorm2d(100, eps=1e-05, momentum=0.1, affine=True)
(nonlin_3): Expression(expression=elu)
(pool_3): MaxPool2d(kernel_size=(3, 1), stride=(1, 1), dilation=(27, 1))
(pool_nonlin_3): Expression(expression=identity)
(drop_4): Dropout(p=0.5)
(conv_4): Conv2d (100, 200, kernel_size=(10, 1), stride=(1, 1), dilation=(27, 1), bias=False)
(bnorm_4): BatchNorm2d(200, eps=1e-05, momentum=0.1, affine=True)
(nonlin_4): Expression(expression=elu)
(pool_4): MaxPool2d(kernel_size=(3, 1), stride=(1, 1), dilation=(81, 1))
(pool_nonlin_4): Expression(expression=identity)
(conv_classifier): Conv2d (200, 2, kernel_size=(1, 1), stride=(1, 1), dilation=(81, 1))
(softmax): LogSoftmax()
(squeeze): Expression(expression=_squeeze_final_output)
)
```

Figure 1.6 DCNN model implementation in PyTorch software, adapted from [123].

Finally, we describe here the feature maps and layers in the deep convolutional network shown in Figure 1.5 and detailed in Figure 1.6.

- The first layer has 10×1 convolution kernels with 3×1 strides across time. We see it does not stride across channels. This process is done 25 times to create 25 feature maps.

- The second layer has $25 \times$ #-of-channels convolution kernels with stride 1 in time. Each convolution is across the channels, collapsing this dimension, and fed to 25 linear units. Again, this is done 25 times to create 25 of 1×198 feature maps that are run through an ELU non-linearity.

- The third layer is simply a max-pooling layer.
- The fourth layer has 25×10 convolution kernels with stride of 3 in time. The convolution is done across channels and time with each feature map, and run through an ELU non-linearity. This is done 50 times.
- The fifth layer is a max-pooling layer.
- The sixth layer has 50×10 convolution kernels with stride of 3 in time. The convolution is done across channels and time with each feature map, and run through an ELU non-linearity. This is done 100 times.
- The seventh layer is a max-pooling layer.
- The eight layer has 100×10 convolution kernels with stride of 3 in time. The convolution is done across channels and time with each feature map, and run through an ELU non-linearity. This is done 200 times.
- The ninth layer is a max-pooling layer.
- The tenth and final layer is a 200×2 dense network layer, with input of 200×1 and output 1×2 for each class, with softmax non-linearity at each node and optimized on a cross-entropy loss function to estimate predicted probabilities.
- The output with the highest probability determines the predicted class.

So, for #-of-channels = 19, there are $[(10 \times 1) \times 19] \times 25 + [25 \times 19] \times 25 + [25 \times 10] \times 50 + [50 \times 10] \times 100 + [100 \times 10] \times 200 + [200 \times 2] = 279,525$ weights in the DCNN.

This deep convolutional neural network model is used throughout our work, in Chapter 3 and Chapter 4 with 19 EEG channels, and Chapter 5 with 20 EEG channels.

1.8 Conclusion

Methods and background presented in this introduction are used throughout. Relevant sections are referenced when a method is discussed or used in each of the following chapters. These four chapters present original contributions to the area of quantitative characterizations of electrophysiological neural activity.

Chapter 2: Consistency of Quantitative Electroencephalography Features in a Large Clinical Data Set

Overview

This chapter presents work that was published in the Journal of Neural Engineering in 2019 [95].

We developed data driven methods to evaluate consistency metrics of quantitative EEG features and validated them using EEG data of clinically normal recordings from the Temple University Health EEG Corpus.

Motivation: For diagnostic purposes, EEG recordings are usually interpreted by trained professionals through visual inspection; however, variation in visual interpretations of EEG has been shown to account for 37% of the variance in EEG diagnoses. An alternative is to use quantitative EEG (qEEG) features. In order to understand the salience of observed qEEG changes across individuals or within the same individual, it is necessary to establish the consistency metrics for qEEG features.

Approach: For our stationarity metric we applied Kruskal-Wallis tests to all subjects and evaluated all ten combinations of five epoch lengths. Averages of all

percent of subjects with significant statistical comparisons were used to determine stationarity levels. For our inter-subject consistency metric we applied Kruskal-Wallis tests to subjects with two sessions and compared different subjects to each other over 100 iterations. Finally for our intra-subject consistency metric we applied Kruskal-Wallis tests to subjects with two sessions and comparing subjects' sessions to each other.

Validation: We validated our stationarity metric by comparing results to coefficients of variation, which can also represent stationarity for certain features, and found both methods to produce consistent results. To validate our inter- and intra-subject consistency metrics we obtained correlation coefficients to evaluate the relationship between inter- and intra-subject consistency of features and confirm results with our novel methods.

Application: We applied this framework to evaluate the consistency of 30 commonly used quantitative EEG features. We found in general that features with normalizing constants were more stationary ($n = 4,313$). We also found entropy, median, skew, and kurtosis of EEG to behave as baseline EEG metrics with high inter-subject consistency ($n = 419$). Finally, other spectral and signal shape features had stronger intra-subject consistency and thus are better for distinguishing individuals ($n = 419$).

Research impact: These results provide data-driven non-parametric methods of identifying EEG features and their spatial characteristics ideal for various EEG applications, and determining future EEG feature consistencies using existing EEG data.

2.1 Introduction

Electroencephalography (EEG), which measures brain electrical signals from the scalp, is a common neuro-monitoring technique used in both clinical and research settings. EEG is typically non-invasive and relatively inexpensive. For diagnostic purposes, EEG recordings are usually interpreted by trained professionals through visual inspection; however, variation in visual interpretations of EEG has been shown to account for 37% of the variance in EEG diagnoses [43]. An alternative is to use numerical quantities computed mathematically and reproducibly from the raw EEG. Such quantities are referred to as quantitative EEG features, or simply qEEG.

In general, neurophysiological measurements such as qEEG have great potential as biomarkers [110]. They directly track neural population activity with high temporal resolution, allowing for objective measurements of cognitive function and communication between brain regions [65]. Numerous qEEG features are under study as biomarkers for neurological and mental health disorders, including ADHD, epilepsy, and brain injury, and for behavioral states including specific movement intent [81], [78], [106]. Despite this increasing variety of applications with public health importance, little is known about the consistency and variability of baseline qEEG features in healthy individuals [129]. Because the population variance of qEEG features could impact the efficacy of medical devices that use this information for treatment or diagnosis, a better understanding of baseline qEEG in a healthy population is essential.

Many factors other than the underlying biology may influence the values of qEEG features. For example, features may be computed over the entire duration of a recording, or from sliding windows, or epochs, of consecutive samples. The choice of epoch length may influence the feature's value; this effect is referred to as the feature's stationarity or time-dependence. Previous studies have investigated the stationarity of spectral features of EEG and the influence of epoch length with up to 109 subjects through analysis of coefficients of variation (COV) [87], [130].

In order to understand the salience of observed qEEG changes across individuals or within the same individual, it is necessary to establish the consistency of features for a single subject across multiple EEG recording sessions (i.e. intra-subject consistency) and across subjects (i.e. inter-subject consistency). The intra-subject and inter-subject consistency of EEG features have also been studied on a limited number of subjects varying from 5 to 148 subjects [49], [50], [97], [96], with many of these studies focusing only on spectral features [42], [119], [40], [39].

With the availability of big data resources, such as the Temple University Health EEG Corpus, it is now possible to expand these types of analyses to a larger group of subjects and examine feature consistency under noisy, less controlled, conditions [51]. This study goes beyond previously reported consistency analyses of qEEG features by evaluating several time- and frequency-domain features using data from several thousand subjects to examine the stationarity and consistency of qEEG features within and between subjects. Unlike most previous studies in which subjects' EEGs are recorded under known conditions, this study is retrospective and based on a population of clinical recordings. It cannot be not known which

activities or tasks the subjects were performing during the EEG recordings used in this study. The features examined are therefore task-agnostic.

With data collection from wearable and even ingestible biosensors increasing at a rapid rate, and with regulatory interest in real-world data also increasing, both the volume of available observational biometric data and its potential utility if properly understood are growing rapidly. If these data are to be useful, it is critically important to understand the distributions of quantitative parameters that may be extracted from them. Human EEG data is a particularly important data type in this regard because of its relevance to a myriad of public health challenges. In this study, by performing analysis removed from any specific condition and with a large population, we hoped to identify characteristics that might be more universally representative than those observed under more controlled conditions, and thus establish a baseline useful for interpreting future data of this type. The aim of this study is to present novel methods of evaluating the consistency of qEEG features in a healthy population in order to inform the efficacy of future medical products utilizing the qEEG to treat, diagnose, or rehabilitate a patient population.

2.2 Methods

Figure 2.1 shows the data processing and analysis sequence used in this study.

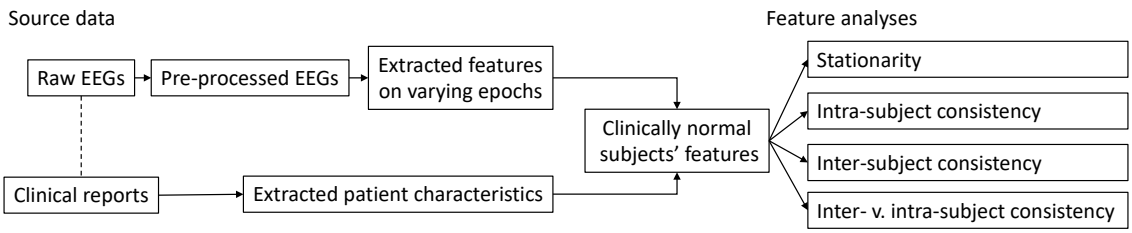


Figure 2.1 Data processing and analysis flowchart.

2.2.1 Data set: Defining 'normal'

Data from the Neural Engineering Data Consortium (NEDC) EEG Corpus from Temple University Health, version 1.0.0 were used in this study. This data set contains 23,257 EEG recording sessions (1.1TB) from 13,551 subjects. Each session is accompanied by a fully-anonymized text file containing non-identifying patient characteristics and clinical impressions of the EEG recordings [51]. We used custom text-mining scripts to extract patient demographics and clinical information from the clinical reports. Since the recordings in the NEDC database were not all acquired with the same EEG system (Section 1.1), the channel layouts were not all consistent. A subset of channels common across most recording sessions was identified and used in subsequent analyses. From a standard 10-20 electrode montage, the subset of channels used were: Fp1, Fp2, F3, F4, C3, C4, P3, P4, O1, O2, F7, F8, T3, T4, T5, T6, Fz, Cz, Pz.

Any session with at least one recording 17 minutes or longer in length was included in our analysis. The first minute of each recording was excluded to reduce artifacts often present at the start of an EEG recording [123]. Our analyses were

applied to the next 16 minutes of EEG recordings, based on recording lengths used in subsets of the NEDC and for ease of epoching, across the 19 common channels named above [83].

Consistent with previous EEG feature consistency studies cited, to avoid confounds from known EEG pathologies in the analysis of feature consistency, only EEG recordings deemed clinically normal by a trained professional were included in the analysis reported here. While some of the individuals presenting with a normal EEG recording may have an underlying pathology, their recordings were still included in the analysis since the recording itself was deemed clinically normal. Subgroup analyses of patient-specific pathologies may be addressed in future work. We developed an automated labeling method based on textual analysis of the clinical reports to classify the recordings as normal or abnormal. We tested our labeling method against the results of a classification of a subset of patient sessions recently published by the NEDC [83]. Lopez et al. reported 99% inter-rater agreement in the classification of a subset of 3,017 sessions as clinically normal ($n = 1,529$) or abnormal ($n = 1,488$). Our automated labeling method matched the NEDC classifications with 98.8% accuracy, labeling 1,510 of 1,529 normal and 1,471 of 1,488 abnormal samples correctly [94]. Applying this method to the entire NEDC data set resulted in a total of 4,313 normal EEG recordings across 3,733 unique subjects, which comprised the normal EEG data set used in this study (Table 2.1).

	All Recordings	Subjects with two visits
Total n	4,313	419
Male	1,772	154
Female	2,226	239
Sex not determined	315	26
Mean age (years)	41.57±18.60, IQR=27 ($n = 4,225$)	42.01±16.60, IQR=26 ($n = 416$)
Median time between visits (days)	N/A	324, IQR=785

Note: IQR - Inter-quartile range

Table 2.1 Normal population demographics.

2.2.2 qEEG features

2.2.2.1 Pre-processing

If necessary, recordings were down-sampled to 100Hz. The data were then filtered through a 0.5Hz to 50Hz band pass filter, and re-referenced using a common average. From this pre-processed data, a set of time and frequency domain features (see column one of Table 2.2) were computed for each of the 19 channels on each recording. All pre-processing and analyses were done in Python 2.7 using a combination of available packages and custom developed code (available at `github.com/dbp-osel/qEEG-consistency`).

2.2.2.2 Time domain features

A set of of time domain features (Section 1.3.2) were selected based on their use in seizure research and other EEG consistency studies [29], [50]. The following time domain features were computed for all epoch lengths: Information theoretic

(I.T.) features: Shannon entropy of the normalized minimum shifted signal, mean thresholded Lempel-Ziv complexity (LZC); Statistical features: minimum value, maximum value, median, variance, standard deviation (SD), skew, kurtosis; Signal shape features: curve length, energy, non-linear energy, sixth power, complexity, and mobility. Time domain features were computed on non-overlapping epochs of 7.5, 15, 30, 60, and 120 seconds across each 16 minute recording, resulting in sets of 128, 64, 32, 16, and 8 measurements per channel per recording, respectively.

2.2.2.3 Frequency domain features

In addition to the time-domain features, spectral features (Section 1.3.1) were computed from Fourier transforms taken over the same epoch lengths. The power spectral density (PSD) (Section 1.2) of frequency bands commonly analyzed in the EEG were estimated using the periodogram as follows: ℓ(lower) $:<$ 1Hz, δ(delta) : $1-4$Hz, θ(theta) : $4-8$Hz, α(alpha) : $8-12$Hz, μ(mu) : $12-16$Hz, β(beta) : $16-25$Hz, γ(gamma) : $25-40$Hz [125]. Both absolute powers and relative powers were computed, with relative power equal to the power in a frequency band divided by the total power. The entropy of the normalized periodogram, spectral entropy, was also computed using the Shanon entropy definition [14].

2.2.3 Feature epochs and sampling

As stated previously, each of the 7.5, 15, 30, 60, and 120 second epoch lengths resulted in sets of 128, 64, 32, 16, and 8 measurements per channel per recording, re-

spectively. To control for spurious variability effects based on number of samples, a subset of eight measurements were used to determine stationarity, intra-subject consistency, and inter-subject consistency of features using non-parametric statistical methods. The subset of eight measurements was selected by dividing the sequence of measurements into even eighths and choosing the first measurement of each segment, that is, choosing the n^{th} measurement from the total of p measurements when $\text{mod}(n, {}^{p}/_{8}) = 0$, where the measurement index n is taken to begin at zero.

2.2.4 Stationarity analyses

2.2.4.1 Coefficients of variation

Before computing COVs, we applied a logarithm transform to the feature values to bring the data closer to a normal distribution [87]. The COV of the natural log of absolute band powers were then compared across different epoch lengths for all subjects to determine the feature's stationarity and stability [87], [130]. COV was computed as $c_v = {}^{\sigma_s}/_{\mu_s}$, where σ_s is the sample standard deviation and μ_s the sample mean (e.g. *subject*: all 4,313 subjects, *feature*: $\log_e$(absolute θ power), *channel*: Fp1, *epoch*: 30 seconds, *samples:* all 32 of 32).

2.2.4.2 Tests of similarity

To expand and test for stationarity on each feature across varying epoch lengths, ten separate Kruskal-Wallis tests (Section 1.5) were applied to each pairwise combination across the five epochs (e.g. 120 seconds **v.** 60 seconds, 120 seconds

v. 30 seconds, etc.). For each of the 4,313 recordings, the Kruskal-Wallis test was applied between the eight sampled measurements of the same recording for each pair of the epoch lengths (e.g. *subject*: 4,313, *feature*: non-linear energy, *channel*: Fp1, *epoch*: 30 seconds, *samples*: {0, 4, 8, 12, 16, 20, 24, 28} of 32 **v.** *subject*: 4,313, *feature*: non-linear energy, *channel*: Fp1, *epoch*: 60 seconds, *samples*: {0, 2, 4, 6, 8, 10, 12, 14} of 16). The null hypothesis of the Kruskal-Wallis test is that the medians of group of samples are the same implying that the distribution of the measurements are consistent. The H statistic was computed on ranks and a p-value obtained using a Chi-Square distribution by $Pr[\chi^2(df - 1) \geq H]$, where $df = 8$ was the degrees of freedom [77]. For each pairwise comparison of epoch length, we report these results as the percentage of 4,313 recordings that were found to have non-significant Kruskal-Wallis results, with $p > 0.05$, indicating that the qEEG feature was consistent across the time series for two given epoch lengths.

2.2.5 Intra-subject consistency analysis

Intra-subject consistency of features was assessed using data from subjects with at least two sessions ($n = 419$). A Kruskal-Wallis test was applied to the eight sampled measurements obtained from two sessions for a given subject and epoch length (e.g. *subject*: 419, *feature*: skew, *channel*: Fp1, *epoch*: 15 seconds, *samples*: {0, 8, 16, 24, 32, 40, 48, 56} of 64, *session*: 1 **v.** *subject*: 419, *feature*: skew, *channel*: Fp1, *epoch*: 15 seconds, *samples*: {0, 8, 16, 24, 32, 40, 48, 56} of 64, *session*: 2). For 15, 30, 60, and 120 second epoch lengths, we report these results as the percentage

of 419 subjects that were found to have non-significant Kruskal-Wallis results, with $Pr[\chi^2(df-1) \geq H] = p > 0.05$, where $df = 8$, indicating the feature tested was consistent across sessions for a given subject.

2.2.6 Inter-subject consistency analysis

Inter-subject consistency was also assessed using the subset of measurements from subjects with at least two sessions for 15, 30, 60, and 120 second epoch lengths ($n = 419$). For each of these subjects, the data for each subjects' second session's recording was replaced with data from a random subject. The Kruskal-Wallis test was then applied as in Section 2.2.5 (e.g. *subject*: 419, *feature*: skew, *channel*: Fp1, *epoch*: 120 seconds, *samples*: {0, 1, 2, 3, 4, 5, 6, 7} of 8, *session*: 1 **v.** *subject*: 92, *feature*: skew, *channel*: Fp1, *epoch*: 120 seconds, *samples*: {0, 1, 2, 3, 4, 5, 6, 7} of 8, *session*: 1). The maximum percentage across channels that were found to have non-significant Kruskal-Wallis results, with $p > 0.05$, was then taken as the level of inter-subject consistency. This process was repeated 100 times to generate a distribution of inter-subject consistency [88].

2.2.7 Intra-subject versus inter-subject consistency analysis: Correlation coefficients

A non-parametric correlation coefficient was used to determine the relationship between intra-subject and inter-subject consistency of features. The absolute Spearman correlation coefficient, r, was computed to determine the correlations of

features based on the difference of within subject ranks and between subject ranks [50]. Spearman correlation values range from $0 \leq r \leq 1$, where zero implies there is no correlation and a value of one implies a direct correlation. A single random measurement for each feature from each channel for each of 15, 30, 60, and 120 second epoch lengths was taken to represent the feature for that subject. The correlation coefficient was then computed on the 419 subjects $\times$ 2 measurements for each feature, channel, and epoch length combination (e.g. *subject*: all 419 subjects, *feature*: skew, *channel*: Fp1, *epoch*: 60 seconds, *samples*: random 1 of 16, *session*: 1 **v.** *subject*: all 419 subjects, *feature*: skew, *channel*: Fp1, *epoch*: 60 seconds, *samples*: random 1 of 16, *session*: 2).

There is support for defining a general threshold of moderate or high correlations as above 0.5 and a negligible correlation as below 0.3 [91]. However, with the current data set, we define a more targeted threshold for negligible correlation coefficient by computing the correlation coefficients of session one features from a specific subject and session two features from a random subject (e.g. *subject*: all 419 subjects, *feature*: skew, *channel*: Fp1, *epoch*: 60 seconds, *samples*: random 1 of 16, *session*: 1 **v.** *subject*: all 419 subjects shuffled in order, *feature*: skew, *channel*: Fp1, *epoch*: 60 seconds, *samples*: random 1 of 16, *session*: 2). This random subject correlation can be calculated 100 times for each feature on each channel for all epoch lengths. By taking the maximum correlation coefficient value that result from these random correlation calculations, we can define a negligible correlation coefficient threshold that is more representative of the data [88]. We expected that features with higher intra-subject consistency from Section 2.2.5 than inter-subject

consistency from Section 2.2.6 to have higher correlation coefficients while features with similar intra-subject and inter-subject consistencies to have lower correlation coefficients.

2.3 Results

If we ultimately seek to understand the relative utility of various qEEG features for applications such as identifying a subject's change in brain state or distinguishing one individual from another, we must first understand the variability of these features in a large and uncontrolled data set. This requires study of the baseline consistency of these measurements, to inform power calculations and data collection for future studies. To assess this consistency we computed a set of features as described in sections 2.2.2.2 and 2.2.2.3, and then performed a series of variability and consistency assessments: stationarity, intra-subject consistency, inter-subject consistency, and intra- versus inter-subject consistency. In all cases, non-parametric statistical methods were used since Shapiro-Wilk's tests indicated that the qEEG features were not normally distributed [126].

Results for these four assessments are shown in Table 2.2. Column one lists the 30 features analyzed, grouped by type of feature. Column two shows stationarity results as described in Section 2.2.4.2. Column three shows inter-subject results as described in Section 2.2.5. Column four shows inter-subject results as described in Section 2.2.6. Finally, column five shows intra- versus inter-subject similarity through correlation coefficient results as described in Section 2.2.7. Further, Figures

2.2*b*, 2.3, and 2.4 show representative examples of results in Table 2.2 graphically, across channels.

2.3.1 Stationarity analyses

Our main strategy for examining stationarity was to make comparisons between feature distributions computed using different epoch lengths, following the reasoning that stationary features should be independent of epoch length. However, given the unknown conditions under which the EEG data were recorded in the NEDC database, we first sought to replicate simpler COV analyses done previously on absolute band powers to compare those results with the larger, task-agnostic data set.

2.3.1.1 Coefficients of variation

COVs were computed as described in Section 2.2.4.1. Figure 2.2a shows the average c_v of 81,947 EEG channels (4,313 subjects $\times$ 19 channels per subject) for each absolute band power using 7.5, 15, 30, 60, and 120 second epoch lengths. We found that as the epoch length increased, c_v values decreased. However, as the epoch length increased, the amount by which c_v decreased was reduced. Figure 2.2b shows the distribution c_v values across all 19 channels for each of the five epoch length for a single representative feature, absolute θ band power. The c_v of the absolute θ band power decreased quickly with increasing epoch lengths, with smaller decrements and less variability in the c_v across channels at longer epoch lengths. These results

	qEEG Features	Stationarity	Intra-subject	Inter-subject	Intra- v. inter-subject
		Percent of subjects with $p > 0.05$			Correlation coefficient
Spectral	Relative ℓ Power	98.66 ± 0.85	66.95 ± 7.04	59.96 ± 4.53	0.29 ± 0.07
	Relative δ Power	99.68 ± 0.22	66.23 ± 4.77	58.11 ± 4.53	0.35 ± 0.10
	Relative θ Power	99.76 ± 0.27	65.39 ± 4.65	57.58 ± 4.65	0.33 ± 0.08
	Relative α Power	99.85 ± 0.19	61.58 ± 5.61	45.47 ± 4.65	0.49 ± 0.05
	Relative μ Power	99.84 ± 0.28	63.37 ± 5.13	51.67 ± 5.07	0.39 ± 0.08
	Relative β Power	99.83 ± 0.24	59.31 ± 4.77	48.39 ± 4.24	0.43 ± 0.09
	Relative γ Power	99.72 ± 0.32	61.46 ± 4.42	49.76 ± 3.34	0.40 ± 0.10
	Absolute ℓ Power	0.00 ± 0.01	67.30 ± 4.83	60.02 ± 4.18	0.23 ± 0.06
	Absolute δ Power	0.00 ± 0.00	61.34 ± 6.38	56.68 ± 5.31	0.34 ± 0.08
	Absolute θ Power	0.00 ± 0.00	51.19 ± 6.74	42.00 ± 4.36	0.52 ± 0.13
	Absolute α Power	0.00 ± 0.01	58.23 ± 5.13	33.77 ± 3.40	0.67 ± 0.05
	Absolute μ Power	0.00 ± 0.01	57.04 ± 6.21	37.23 ± 3.40	0.57 ± 0.08
	Absolute β Power	0.00 ± 0.00	58.35 ± 6.86	41.95 ± 3.70	0.56 ± 0.10
	Absolute γ Power	0.00 ± 0.01	59.67 ± 6.44	51.01 ± 4.06	0.44 ± 0.12
	Spectral Entropy	0.08 ± 0.14	65.87 ± 4.77	56.44 ± 4.06	0.31 ± 0.08
I.T.	Entropy	0.00 ± 0.00	78.64 ± 5.13	73.57 ± 2.92	0.18 ± 0.08
	LZC	83.64 ± 8.85	67.54 ± 6.92	64.80 ± 4.47	0.34 ± 0.12
Statistical	Minimum	1.68 ± 0.37	73.03 ± 5.73	67.90 ± 4.36	0.32 ± 0.12
	Maximum	1.63 ± 0.74	72.91 ± 5.79	65.75 ± 4.42	0.31 ± 0.11
	Median	100.00 ± 0.00	78.16 ± 7.46	77.80 ± 4.06	0.17 ± 0.15
	Variance	99.55 ± 0.48	62.65 ± 7.16	57.16 ± 5.13	0.45 ± 0.14
	SD	99.55 ± 0.48	62.65 ± 7.16	57.46 ± 4.59	0.45 ± 0.14
	Skew	98.31 ± 1.65	80.91 ± 6.38	78.16 ± 4.00	0.15 ± 0.10
	Kurtosis	26.57 ± 11.36	80.55 ± 4.89	77.80 ± 3.82	0.16 ± 0.10
Signal Shape	Curve Length	0.00 ± 0.00	60.26 ± 3.94	45.70 ± 4.12	0.38 ± 0.09
	Energy	0.00 ± 0.00	59.07 ± 7.58	42.78 ± 4.00	0.56 ± 0.12
	Non-linear Energy	99.55 ± 0.48	62.65 ± 7.16	57.16 ± 5.01	0.45 ± 0.14
	Sixth Power	99.70 ± 0.37	59.79 ± 7.82	43.20 ± 3.64	0.55 ± 0.11
	Mobility	99.46 ± 0.27	61.10 ± 6.86	50.72 ± 3.64	0.35 ± 0.10
	Complexity	97.75 ± 1.80	64.44 ± 5.25	52.68 ± 4.30	0.44 ± 0.12
	Legend:	≤ 1% 1% < ≤ 99% 99% <	≤ 50% 50% < ≤ 75% 75% <	≤ 0.22 0.22 < ≤ 0.5 0.5 <

Note: I.T. - Information theoretic, LZC - Lempel-Ziv complexity, SD - Standard deviation

Table 2.2 Stationarity results show the $median \pm (range \div 2)$ across channels and all pairwise comparisons, with highly stationary (green), somewhat stationary (yellow) and non-stationary (red) features, as percent of subjects ($n = 4,313$). Intra-subject consistency shows the $median \pm (range \div 2)$ across channels and epochs, as percent of subjects. Inter-subject consistency shows the $median \pm (range \div 2)$ of the max across channels and epochs of the inter-subject distribution across 100 comparisons, as percent of subjects. Colors in these two columns reflect the level of consistency of the feature ($n = 419$). Intra-subject versus inter-subject consistency shows the $median \pm (range \div 2)$ across subjects, channels, and epochs of the correlation coefficients. Colors in this column reflect how much higher the intra-subject consistency is to consistencies from the random distribution, maximally 0.22, and the significance of the correlation ($n = 419$).

were consistent with those previously reported [87], [130]. Having established this baseline concordance, we proceeded to assess stationary, and intra- and inter-subject consistency for the complete set of features.

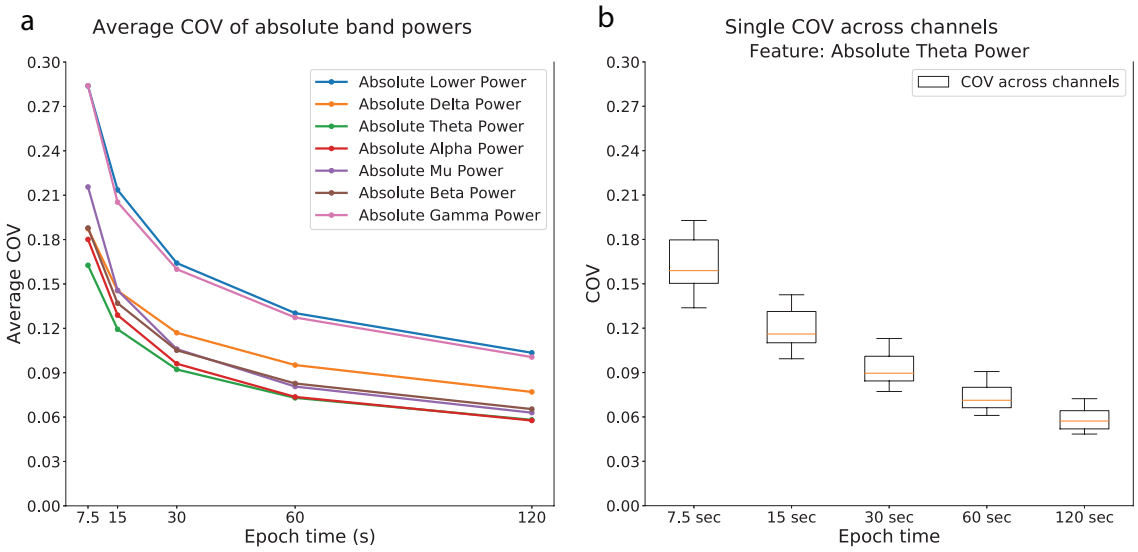


Figure 2.2 Coefficient of variation (COV) results: (**a**) averaged across channels for all absolute power bands, and (**b**) example band with channel distribution. ($n = 4,313$)

2.3.1.2 Tests of similarity

We next investigated the similarity of feature distributions when computed from different epoch lengths. To do this we performed a series of Kruskal-Wallis tests. Each individual test was a comparison of the distributions of a feature computed from two different epochs. A non-significant ($p > 0.05$) result therefore indicated that, for that subject, the feature in question was not statistically different between the two epoch lengths. The percentage of subjects with non-significant Kruskal-Wallis test results when comparing qEEG features from varying epoch lengths was calculated as described in Section 2.2.4.2. Representative exam-

ples of results are presented as box plots showing the distribution of the percentage of tests with $p > 0.05$ across channels for each epoch length comparison in Figure 2.3. Each panel of this figure summarizes the results for one feature (4313 subjects $\times$ 10 pairwise Kruskal-Wallis tests). There were three main trends revealed from the population results. For certain features, such as non-linear energy (Figure 2.3a), we found that nearly all subjects had non-significant Kruskal-Wallis test results, indicating that the computed feature did not statistically differ over different epoch lengths and therefore had high stationarity. For other features, such as Lempel-Ziv Complexity (shown in Figure 2.3b), Kruskal-Wallis results showed significant differences in distribution in some epoch pairs, but not others, indicating that choice of certain epochs could modulate the distribution of this feature. Lastly, some feature distributions, such as spectral entropy (Figure 2.3c), gave significant Kruskal-Wallis results for all epoch pairs, indicating that the computed feature differed statistically over all tested epoch length choices, and was therefore non-stationary. Table 2.2 provides a summary of stationarity for all analyzed features based on these three observed trends. As a summary statistic, the median value of the percentage of subjects for all epoch comparisons was used to categorize each feature. Features categorized as highly stationary with a median percent of consistent subjects across all tests greater than 99% are highlighted in column two of Table 2.2 in green. Features with a median percent of consistent subjects between 1%-99% were categorized as somewhat stationary and highlighted in column two of Table 2.2 in yellow. Features with a median percent of consistent subjects less than 1% were categorized as non-stationary and highlighted in column two of Table 2.2 in red. In general, the

greater the time difference between the pair of epoch lengths compared, the lower the percent of consistent subjects (Figure 2.3).

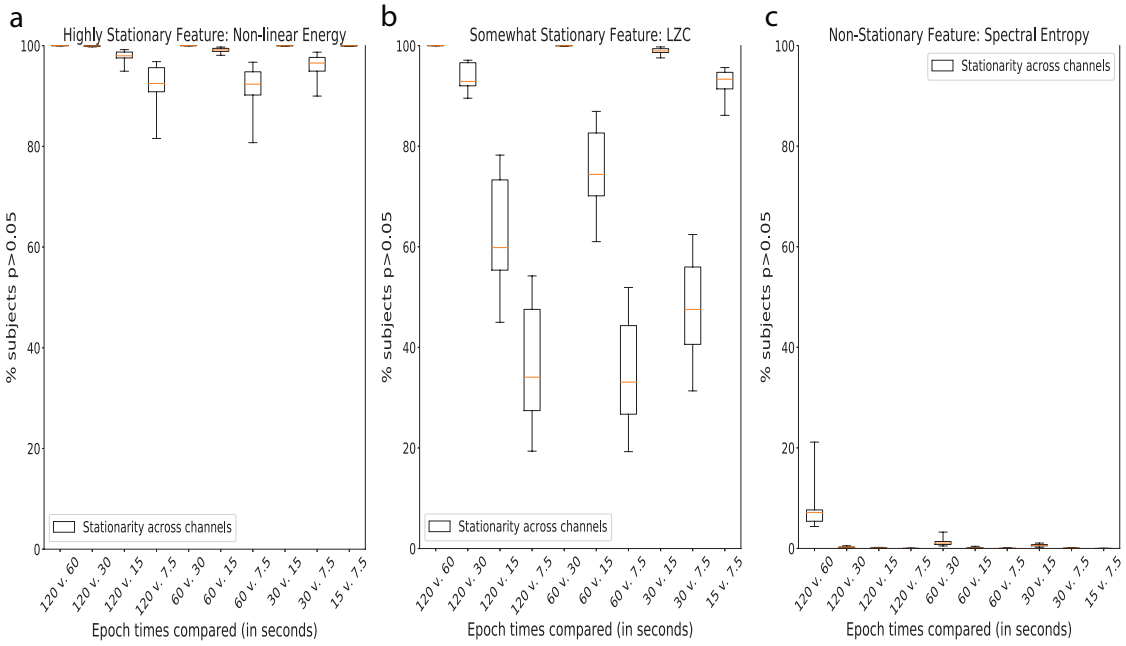


Figure 2.3 Example features for each level of stationarity: (**a**) highly stationary, (**b**) somewhat stationary, (**c**) non-stationary, with box-plots showing the percent of consistent comparisons across channels. ($n = 4,313$)

2.3.2 Intra-subject and inter-subject consistency analyses

We next examined the question of whether particular features could be used to distinguish between individuals. This would be of particular interest for features such as non-linear energy or relative α power, that had been shown in our previous analysis to be highly epoch-independent or stationary within an individual. Could this within-individual consistency be used as a "marker" of individuals, or would the same consistency be observed across individuals?

Intra-subject and inter-subject consistencies were determined by using a Kruskal-

Wallis test for qEEG features calculated across two sessions for either the same subject or different subjects, respectively, for various epoch lengths as described in Sections 2.2.5 and 2.2.6. In Figure 2.4a-b, intra-subject results are displayed as a box plot of percentage of subjects with non-significant Kruskal-Wallis tests, with each box plot representing the distribution across channels. The distribution of inter-subject results are shown as points on the same axes where each of the points represent the maximum percentage across all channels of one iteration of the comparisons of random subjects repeated 100 times. Representative examples of the intra-subject box plots and inter-subject distributions are shown in Figure 2.4a-b with skew and absolute α band power.

To distinguish between different levels of consistency we establish 75% as the high consistency threshold and 50% as a low consistency threshold based on our observed results. Features that had the highest intra-subject and inter-subject consistencies with at least 75% of the subjects with consistent features were entropy, median, skew (Figure 2.4b), and kurtosis, highlighted in green in columns three and four of Table 2.2. On the other hand, features with low inter-subject consistencies, defined as those consistent across fewer than 50% of subjects, included: relative α, β, and γ powers, absolute θ, α (Figure 2.4a), μ, and β powers, curve length, energy, and sixth power, highlighted in column four of Table 2.2 in red. These features were often found to have higher intra-subject consistencies, as indicated in column three of Table 2.2. Features with moderate levels of intra-subject and inter-subject consistencies showed between $50\% - 75\%$ of the subjects with consistent features, highlighted in yellow columns three and four of Table 2.2. In general, the differ-

ences between features and epochs were much larger than the differences between channels, although systematic spatial differences across channels are addressed in Section 2.3.2.1.

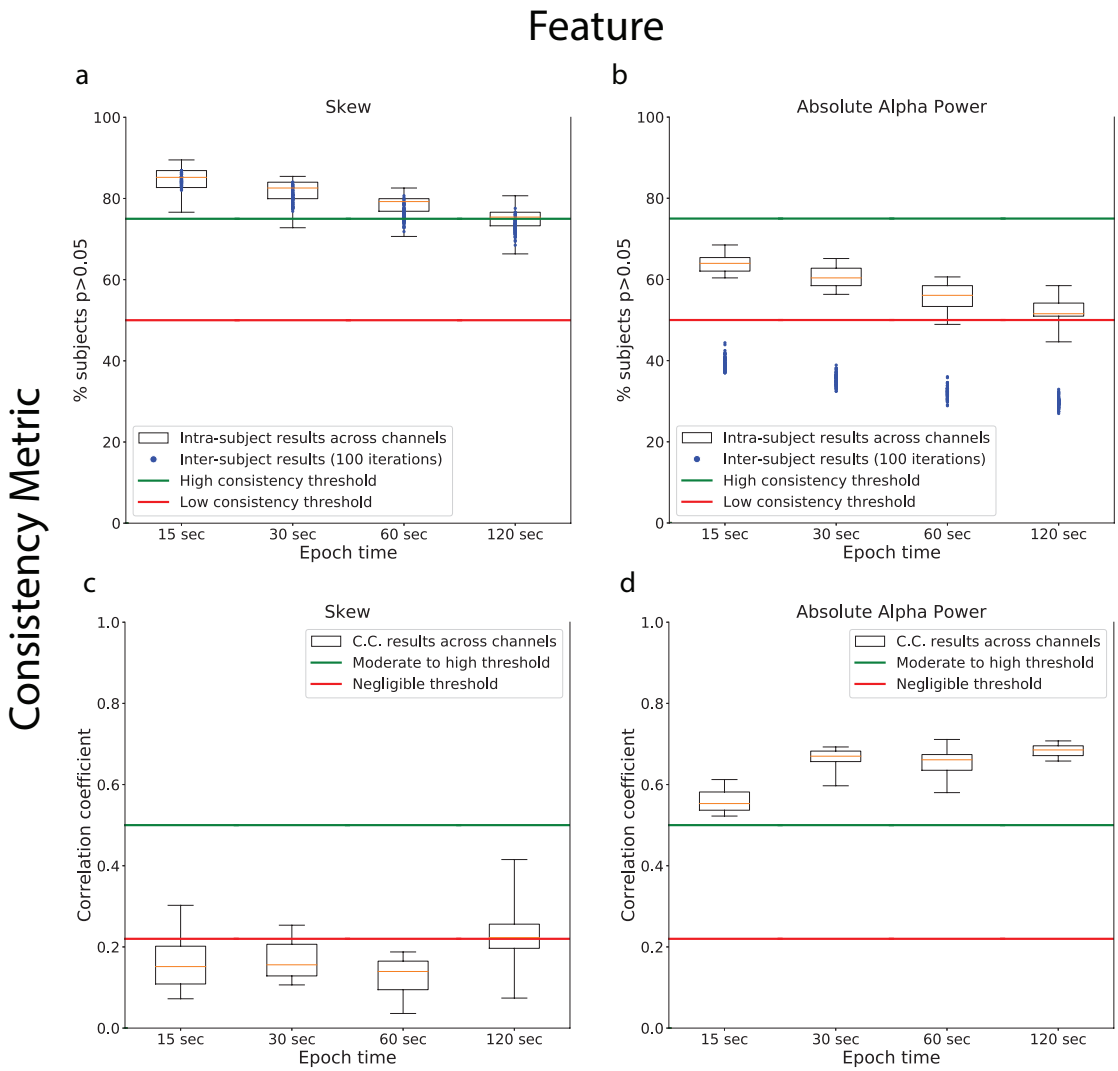


Figure 2.4 Intra-subject consistency (**a**)-(**b**): Box-pots show the percent of subjects with statistically consistent features between two of their sessions across channels. Inter-subject consistency: Points show the 100 iterations of the maximum across channels of the percent of subjects with statistically consistent features with another random subject. Green lines show high consistency threshold of 75% and red lines show low consistency threshold of 50%. Correlation coefficients (**c**)-(**d**): Box-plots show subjects' correlation coefficient between two of their sessions across channels. The negligible correlation threshold from the maximum correlation from 100 iterations of random comparisons, 0.22, is marked in red and the moderate to high correlation threshold, 0.5, is marked in green. ($n = 419$)

2.3.2.1 Spatial analysis of intra-subject consistency

To understand the spatial distribution of feature consistency within subjects, these intra-subject consistency results were visualized topographically. Topographic maps are constructed by piecewise linear barycentric interpolation from the 19 channel values [6]. Figure 2.5 shows the relative strength of channels' median percent of consistent subjects, across epoch lengths, for each feature. In general, the frontal lobe showed the highest levels of correlation followed by the temporal lobe. However, relative band powers, lower frequency absolute band powers, spectral entropy and mobility show the highest levels of consistency in the pre-frontal lobe.

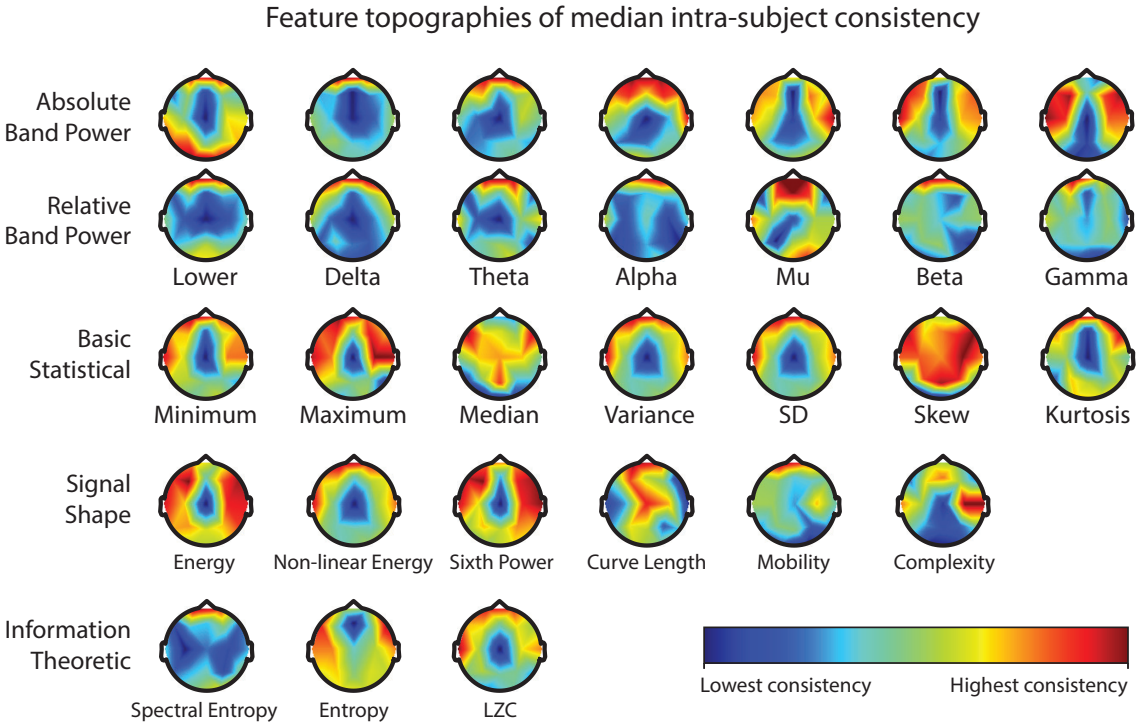


Figure 2.5 Topographies of subjects' median intra-subject consistency across epoch lengths. ($n = 419$)

2.3.3 Intra-subject versus inter-subject consistency analysis: Correlation coefficients

As Figure 2.4a-b shows, some features had high intra-subject but lower inter-subject consistency. To quantify this difference, per-feature per-channel, correlation coefficients between intra- and inter-subject consistencies were computed as described in Section 2.2.7 and are shown in the fifth column of Table 2.2. Representative examples of the correlation coefficient distributions across channels are shown in Figure 2.4c-d, again using skew and absolute α power as examples.

The distribution of correlation coefficients of features between two randomly selected subjects had $median \pm (range \div 2)$ of 0.09 ± 0.07, with a maximum of 0.22 across all features. This maximum was thus taken to be the negligible correlation threshold used for this data. Entropy, median, skew (Figure 2.4c), and kurtosis showed correlation coefficients between the same subject's sessions that were at or below the negligible correlation coefficient threshold. These features are shown in Table 2.2 (column five, in red). All other features had correlation coefficients between sessions of the same subject higher than the defined negligible correlation coefficient threshold, though generally a low to moderate level of correlation. Features where the median correlation across epochs was less than 0.5 are shown in column five of Table 2.2 in yellow. Absolute θ, α, μ, and β powers, energy, and sixth power are found to have correlations greater than 0.5, shown in column five of Table 2.2 in green. An example feature, absolute α power, with correlation coefficients greater than 0.5 is shown in Figure 2.4d.

2.3.3.1 Spatial analysis of correlation coefficients

To understand the spatial distribution of feature consistency within versus between subjects, these correlation coefficient results were visualized topographically as done in Section 2.3.2.1. Figure 2.6 shows the relative strength of channels' median correlation coefficients, across epoch lengths, for each feature. In general, the occipital lobe showed the highest levels of correlation followed by the central-parietal lobe. However, spectral entropy and curve length show the highest levels of correlation in the frontal lobe.

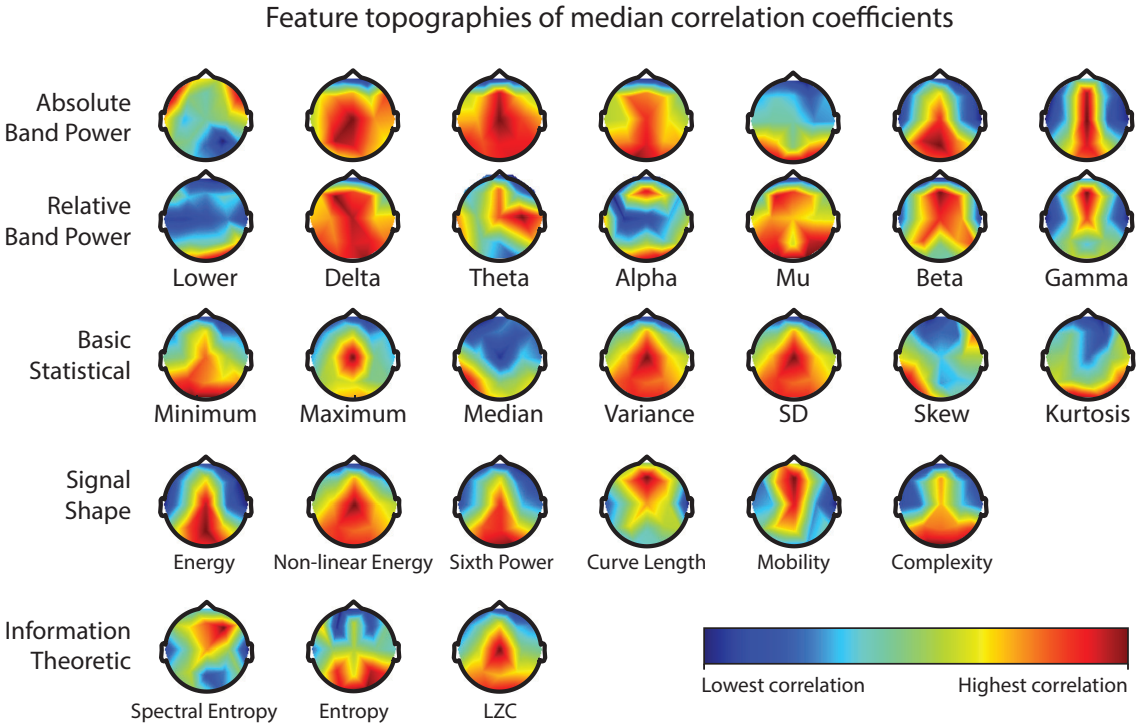


Figure 2.6 Topographies of subjects' median correlation coefficients between two of their sessions. ($n = 419$)

2.4 Discussion

The research presented here expands upon previous consistency and variability studies by applying analyses to a much larger set of subjects. A reproducible data-based, non-parametric, framework utilizing a publicly accessible data set was developed to assess consistency of qEEG features.

The COV results of power spectra bands shown in Figure 2.2a have the same trends as previously found [87], [130]. c_v decreased as the epoch lengths increased and stabilized at higher epoch lengths. The replication and expansion of these results validated that the data followed known EEG trends with regard to features measured on different epochs. Further expanding stationarity analysis through a test of similarity, features' dependence on epoch lengths were evaluated. As expected, we found that absolute band powers were not stationary as evident from the COV results which varied across epoch lengths. In general, we found that most features with a normalizing constant (e.g. relative powers) were consistent across varying epoch lengths and thus highly stationary (highlighted in column two of Table 2.2 in green). These normalizing constants include a division by overall information about the signal, such as total spectral power or number of samples. However, some features without a normalizing constant were also found to be stationary, such as median, while features with a normalizing constant were found to be non-stationary including energy. Other features were found to be less consistent as the differences between epoch lengths increased making them somewhat stationary. In particular, for minimum and maximum values it is thought to be more likely to find spurious

extreme values as the epoch lengths are increased. Stationarity and epoch length influence of features can further inform which features might be best to consider depending on the application the feature is used in, and control over timing of stimuli exposure and EEG recording time.

Table 2.2 shows that entropy, median, skew, and kurtosis had the highest percent of intra-subject and inter-subject consistency yet low correlation coefficients. This was expected and implies that these features were as consistent across subjects as they were within subjects. These types of features could be useful in establishing baseline features for a standard EEG signal and subsequently detect variations away from those baselines. If these particular features are to be used for a given application that requires high intra-subject consistency, for example, the results shown in Figure 2.5 would generally indicate that measurements from frontal and temporal areas would be most consistent. This information can be used to inform brain-computer interface (BCI) design or provide more targeted analysis for diagnostic purposes. Alternatively, features with significant differences between intra-subject and inter-subject consistencies meant that the similarity and consistency of the features were more influenced by the individual. Most spectral and signal shape features reflect this pattern. These features would therefore be better for applications that require unique identification of individuals as well as the many machine learning applications in EEG [27]. Our results showed that absolute band powers had stronger correlation coefficients than relative band powers. However, previous results showed relative band powers generally had a higher correlation than the corresponding absolute band powers [50]. These differences may come from the dif-

ferences between subjects' tasks during recordings or the number of subjects used in this study. The topographic information for channel consistency and importance for each feature shows which areas of the brain, primarily the occipital lobe, would best be targeted for the highest levels of correlation coefficients of each feature.

A new method of characterizing consistency of qEEG features using non-parametric statistical tests is introduced in the current study. These 'tests of similarity' applied to thousands of EEG recordings provided a sense of how similar qEEG features are as a function of epoch length, both within and across subjects, by looking at the percentage of subjects with non-significant statistical results. The agreement between the COV results and our novel tests of similarity as well as agreement between inter-subject and intra-subject analyses with the correlation coefficient results support this novel method in the context of previous works and methods.

Limitations of many EEG studies may include sensitivity to the recording environment and tasks being performed as well as the limited number of subjects used in analysis. The current study provides a framework to analyze task-agnostic recordings from thousands of individuals in order to assess the consistency of qEEG features. There were several conclusions drawn from this work about the best features to use for given applications. While the number of subjects analyzed in this work impresses confidence in the consistency results, we cannot neglect the possible influence of recording condition and task performed. Additional work should be done using large data sets to investigate the influence specific tasks have on consistency of qEEG features. Alternatively, similarities would distinguish between

features that represent baseline EEG and the individual rather than a biomarker of specific tasks being performed. We provide the data pipeline (Figure 2.1) as outlined in this study so that these analyses can be applied to other potential qEEG features on other data sets, or on larger versions of this data set. Future work will investigate wavelet, connectivity, and spectral graph theoretic based features. Finally, since differences between intra-subject and inter-subject consistencies can quantify a qEEG feature's discriminability between subjects, they may also be used to inform feature selection in future machine learning classification applications.

2.5 Conclusion

This study addresses the knowledge gap in our understanding of the consistency of qEEG measurements derived from 'big' data sets collected in noisy, less controlled environments in healthy individuals. A data-driven non-parametric framework is introduced to analyze task-agnostic EEG recordings from thousands of individuals to assess the recording time dependence and inter- and intra-subject consistency of thirty commonly used qEEG features. We identify qEEG features and their spatial characteristics ideal for different EEG recording time protocols, certain baseline qEEG comparisons, or machine learning applications. Results from this work provide novel methods of evaluating the consistency of qEEG features in a healthy population. These tools can be used to evaluate the potential efficacy of qEEG features in future diagnostic and BCI applications of EEG. These results also provide methodology for determining the consistency of novel qEEG features using

an existing EEG data set.

(This page intentionally left blank)

Chapter 3: Deep Learning and Feature Based Medication Classifications from EEG in a Large Clinical Data Set

Overview

This chapter presents work that has been submitted for journal publication.

Using EEG data from the Temple University Health EEG Corpus, we applied kernel support vector machines (kSVM) and deep convolutional neural networks (DCNN) to classify subjects' medication status solely through EEG and compared results from each method.

Motivation: Although research into applications of machine learning to EEG has been underway for a decade, more work is needed to understand its limits in deriving information that could inform clinical decisions. The ability to discern medications taken by an individual through neurophysiological features alone could have benefits in emergency medicine (e.g. determining appropriate life-saving interventions), obtaining more objective and accurate medication reporting, and predicting treatment outcome and effectiveness

Approach: We identified two medications that were prominently used by subjects in the EEG data set, Dilantin (generic phenytoin) and Keppra (generic

levetiracetam). We further separated subjects based on presence of EEG pathologies to reduce confounds. Using previously evaluated EEG features we found optimal kSVM feature-based models. We also adapted a deep learning model to 19 channel EEG inputs for medication identification.

Application: We applied these methods to identify differences between subject's taking anticonvulsants and no medications solely based on EEG signal ($350 \leq n \leq 640$). We obtained significant results with both kSVM and DCNN approaches. KSVM and DCNN model obtained similar levels of accuracy on test data sets. We found that results were more significant for subjects with abnormal EEG. Finally, each of the two anticonvulsants studied seem to generate similar levels of discriminability.

Research impact: This work provides a first step in determining the utility of machine learning in medication-use classification based on neurological activity. By investigating the abilities of different machine learning algorithms to differentiate individuals taking different medications though neurophysiological signals provided through EEG, this work is a step on the road to the broader utility of machine learning in biomarker development.

3.1 Introduction

Machine learning applied to medical research and treatment has the potential to revolutionize the patient-care continuum. From diagnosis to prognosis, treatment to rehabilitation, advanced algorithms are enabling more patient-specific care by leveraging existing data from patients [112]. With this emerging analytical approach nipping at the heels of medical practice, more portable, compact medical devices are being developed to capture physiological and/or movement data from individuals within controlled laboratory settings or in the real-world that can be used as input in machine learning algorithms [102].

Electroencephalography (EEG) is one example of an application space with a high potential for impact from machine learning, as this device technology is increasingly being used for diagnostic and rehabilitative purposes [78] [117], and as a surrogate biomarker for pharmacodynamic modelling of drugs [66] [4] [90]. EEG is typically non-invasive, relatively inexpensive, and allows for direct tracking of neural population activity with high temporal resolution for objective measurements of cognitive function and communication between brain regions [125]. Although research into applications of machine learning to EEG has been underway for a decade, more work is needed to understand its limits in deriving information that could inform clinical decisions. Specifically, adequate validation data sets with known subject or patient characteristics are necessary in order to assess the accuracy and adequacy of machine learning approaches to clinical problems. Up to the present, this necessity has limited the size of most research studies to a small number of patients.

Fortunately, the creation of large clinical databases with both patient information and corresponding physiological data will now enable deeper exploration of many classification problems with clinical relevance. One such database is the Neural Engineering Data Consortium (NEDC) EEG Corpus from Temple University Hospital (TUH).

The TUH corpus is an unprecedented data set composed of thousands of subject recordings collected under conditions less constrained than those typically employed in laboratory research settings [51]. It primarily consists of data from subjects with EEG abnormalities, with many subjects taking anticonvulsant medications, thus providing the means to explore medication state classification by evaluating the neurophysiological signals of thousands of subjects. The ability to discern medications taken by an individual through neurophysiological features alone could have benefits in emergency medicine (e.g. determining appropriate life-saving interventions), obtaining more objective and accurate medication reporting, and predicting treatment outcome and effectiveness [12]. Can EEG alone be used to differentiate someone taking a certain medication from someone not taking any medication? Can EEG alone be used to differentiate individuals taking two different medications from the same drug class (e.g. anticonvulsants, opioids)? More generally, what types of biomedical questions are amenable to conclusive retrospective analysis of EEG data? With the increasing availability of large sets of clinical EEG data, we can begin to explore these questions.

When applying machine learning, there are generally two different algorithmic approaches to consider: deep learning or feature-based algorithms, each with

advantages and disadvantages. Since EEG data can be difficult to interpret, even by experts, the automated feature extraction provided by deep convolution neural network (DCNN) learning models may be advantageous and require less knowledge about the data set and signal to perform well [27]. When expert knowledge about the data and signals are available, application-specific features can be crafted and used in a feature-based approach such as kernel support vector machines (kSVM). Feature-based methods have been shown to perform well in EEG applications and to produce consistent results [84] [19]. Thus, to determine potential differences in classification performance and impact on understanding the underlying clinical implications, both feature-based and deep learning models were investigated and compared as previously done in novel classification paradigms[76] [107] [133].

The goals of this work were to (1) determine if machine learning can predict medication states from neurophysiological activity captured through EEG and (2) compare the accuracy of feature-based and deep learning classification methods. By investigating the abilities of different machine learning algorithms to differentiate individuals taking different medications though neurophysiological signals provided through EEG, this work is a step on the road to the broader utility of machine learning in biomarker development.

3.2 Methods

To investigate the different medication-based classifications, we applied several processing steps to EEG data obtained and corresponding clinical reports from

the NEDC EEG Corpus from TUH, version 1.0.0 (Fig. 3.1). This data set contains 23,257 EEG recording sessions (1.1TB) from 13,551 subjects with a total of 61,802 recording files. Each session was accompanied by a report containing clinical impressions and patient characteristics, including age and sex, as well as the medications being taken by the patient at the time of recording [51]. The pre-processing methods, classification definitions, and machine learning approaches are described below.

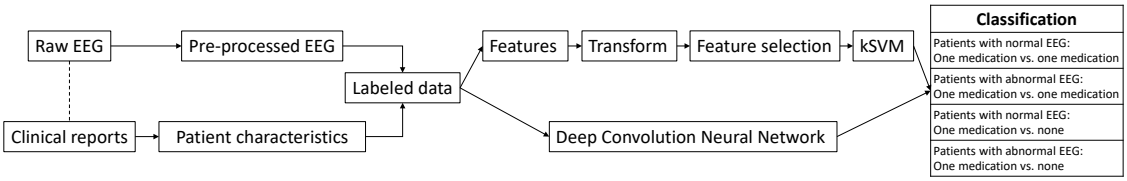


Figure 3.1 Data processing and classification flowchart.

3.2.1 EEG data

Since the recordings in the NEDC database were not all acquired with the same EEG system, the channel layouts were not all consistent. A subset of channels common across most recording sessions was identified and used in subsequent analyses. From a standard 10-20 electrode montage (Section 1.1), the subset of channels used were: Fp1, Fp2, F3, F4, C3, C4, P3, P4, O1, O2, F7, F8, T3, T4, T5, T6, Fz, Cz, Pz. Any recording that did not include all of these channels was omitted from analysis.

Recordings without at least one recording six minutes in length were omitted from analysis. The first minute of each recording was excluded to reduce artifacts present in the EEG [123]. Our analyses were applied to the next five minutes of

EEG recordings across those 19 channels identified above [83]. If another sampling rate was used, recordings were down-sampled to 100Hz. The data were then filtered through a 0.5Hz to 50Hz band pass filter, and re-referenced using a common average. All pre-processing and analyses were done in Python using a combination of available packages, published code from previous works, and custom developed code [105] [123].

3.2.2 Classifications: Defining subject populations

A subset of patient clinical reports within the NEDC data set had been previously reviewed and labeled by curators of this database to be clinically normal or abnormal [83]. The definitions of normal and abnormal used by the NEDC generally followed methods neurologists use to identify abnormalities [86]. These labels were applied broadly to recordings where the clinical reports identified an abnormality in the EEG and do not distinguish between different abnormalities. An automated labeling method previously developed and validated based on textual analysis of the clinical reports was applied to categorize all available recordings as normal or abnormal [95]. Applying this method to the entire NEDC data set resulted in a total of 6,001 EEG recordings with normal EEG across 4,058 unique subjects and 15,347 EEG recordings with abnormal EEG across 7,432 unique subjects.

Given the majority of clinical recordings from this data set were focused on diagnosing or monitoring individuals with epilepsy or seizure disorders, our medication classification focused on anticonvulsants. An initial textual analysis of the

clinical notes for each EEG recording revealed hundreds of individuals taking the drugs Dilantin (generic name phenytoin) or Keppra (generic name levetiracetam), which are categorized as anticonvulsants [146]. For the purposes of this study, the clinical notes were used to determine which subjects were taking either Dilantin or Keppra, and which subjects were taking no medications. Any subject with more than one medication listed in the clinical notes was excluded from the evaluation to reduce results related to drug interaction effects. Furthermore, given the clinical nature of the data set and to avoid confounds from normal and abnormal EEGs, these classifications were performed separately on subjects with normal EEG and abnormal EEG.

Overall, six (three medication states $\times$ two EEG pathology states) different classifications were defined and analyzed using these data with two different machine learning approaches. The population demographics of the entire data set used and each group of subjects are shown in Table 3.1.

	All recordings	Taking Dilantin	Taking Keppra	No medications
Total n	35,370	718	898	5,1815
Normal EEG	6,001	179	175	1,203
Abnormal EEG	15,347	320	264	1,285
EEG status not obtained	14,022	291	459	2,693
Male	15,106	344	378	2,017
Female	16,434	320	465	1,895
Sex not obtained	3,830	54	55	1,269
Age (Mean $\pm$ SD years) (# recordings w/ age)	50.23$\pm$19.15 ($n = 33,021$)	41.14$\pm$18.38 ($n = 692$)	48.04$\pm$18.98 ($n = 862$)	44.55$\pm$20.11 ($n = 4,142$)
Unique subjects	13,486	456	507	2,530

Table 3.1 Population demographics across all medications of interest and no medications.

Pre-processed data were linked to medication labels extracted from their corresponding clinical reports for two machine learning approaches, described below.

We note that other patient demographics (age and sex) were well balanced across all populations investigated.

3.2.3 qEEG feature-based classifiers

qEEG features

From the pre-processed data, a set of time and frequency domain features (Section 1.3) were computed for each of the 19 channels on each recording. qEEG features used in this study were found in EEG literature and previously analyzed for inter- and intra-subject consistency [29] [50] [95]. Time domain features were computed on each five-minute recording. For spectral features, the Fourier transforms were taken on the five-minute pre-processed recording after which various spectral features were computed.

The power spectral density (PSD) (Section 1.2) of frequency bands commonly analyzed in the EEG were estimated using the periodogram. The ranges of the frequency bands applied in this study were as follows: δ(delta) : $1-4$Hz, θ(theta) : $4-8$Hz, α(alpha) : $8-12$Hz, μ(mu) : $12-16$Hz, β(beta) : $16-25$Hz, γ(gamma) : $25-40$Hz [125]. Both absolute powers and relative powers were computed, with relative power equal to the power in a frequency band divided by the total power. The entropy of the periodogram, and entropy of the normalized periodogram, were found using the Shanon entropy definition [14]. In addition to the spectral features, the following time domain features, directly from the pre-processed EEG signal, were computed: entropy of the normalized signal, mean thresholded

Lempel-Ziv complexity (LZC), curve length, energy, non-linear energy, sixth power, minimum value, maximum value, median, variance, standard deviation, skew, kurtosis, integral, sum, mobility, complexity. The frequency and time domain features selected resulted in a total of 570 features per recording (30 features $\times$ 19 channels).

3.2.3.1 Kernel support vector machine classifiers

Features were transformed using max normalization [132] [32]. The max normalization transform was applied to the data by finding the mean and maximum value of each feature across all subjects in the training data. Each value, in both the training and testing data set, was then subtracted by the mean and divided by the maximum values obtained from the corresponding training set. This transformation scaled each feature to a range of values between zero and one in the training data and approximately between zero and one in the testing data set.

A grid search was used to determine optimal dimensionality reduction method, kernel, and hyperparameters for the kSVM classifier (Section 1.7.1). A grid search on 500 different combinations across: (1) dimentionality reduction methods ={PCA, K-best}, (2) number of features used = $\{30, 60, 90, 120, 150\}$, of the available 570 features, (3) kernel ={linear, radial basis function (RBF)}, (4) the penalty parameter $C = \{0.1, 1, 10, 100, 1000\}$, and (5) the RBF smoothness parameter $\gamma =$ {0.1, 0.01, 0.001, 0.0001, 0.00001}, was performed for each classification [105]. This training and grid search was done on another 10-fold cross validation to tune the feature-based model using a validation set extracted from the initial training data

set, independent from the testing set.

3.2.4 Deep convolution neural network EEG classifiers

We used a variant of a DCNN model (Section 1.7.2), introduced in previous research shown in Figure 3.2, which was applied to the abnormal TUH data set [123] [83]. The model used required 279,525 weights to be trained, which were randomly initialized. The main non-linearity used in many of the layers was the

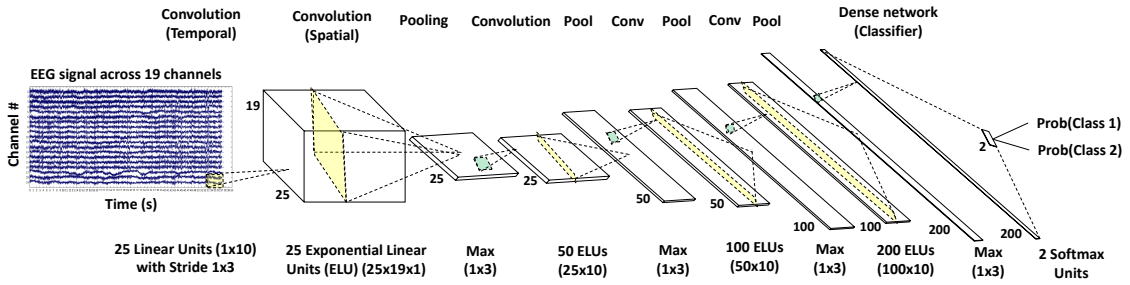


Figure 3.2 Deep convolutional neural network architecture. Convolutions are shown in yellow and all have stride 1x3, across the temporal dimension. Max pool layers are shown in green. The classification is determined by the class with the highest output probability.

exponential linear unit (ELU) function [23]. ELU is a strong alternative to the commonly used rectified linear unit (ReLU) function since it can generate negative outputs and decreases smoothly and slowly until its output equals the parameter value. The network also applies batch normalization and dropout of 0.5 with each convolution and pooling pair [61] [131]. This neural network was optimized though the stochastic gradient descent inspired Adam algorithm [74]. The architecture used was similar to ResNet DCNN, which has been shown to be an effective neural network architecture for automated feature extraction [54]. The first convolution and pooling pair was different from those used in most deep learning models because of

the structure of EEG data. The two-step convolutional sequence in the first two convolution and pooling pairs is inspired in part by the filter-bank common spatial pattern (FBCSP) algorithm [80] [3] [21]. Since EEG data is a collection of one-dimensional signals rather than a two-dimensional array where the dimensions are coupled, such as an an image, the first convolution was applied to each channel separately, across time. The subsequent convolution then uses this result to apply a convolution on the resulting feature maps and channel collection, over time. Afterwards, three additional convolution and pooling pairs can be applied to the resulting feature-maps in a more traditional fashion. The final layer takes the final feature-map and applies a dense neural network with a softmax non-linearity to obtain prediction probabilities for each class. The class with the highest probability is the class determined by the deep network.

The model was trained for 35 epochs for a first run and then restarted for a second run based on the epoch model with the highest training accuracy among the epochs from the first run. The second run's stopping condition was based on a specified training loss to be met based on the first run' results. A validation set was taken from 10% of the training set, independent of the testing data, for 10-fold cross-validation for model tuning. The application was run on an NVIDIA GTX-1080Ti for training and subsequent testing through Python 3.5 and Torch [123] [98].

3.2.5 Classification results and significance

For each binary classification, an equal number of subjects from each population was used to ensure equal prevalence of each condition. 90% and 10% of the data were assigned randomly to a training and testing set, respectively. This process was repeated ten times to obtain classification test results using different subsets of the data for a 10-fold cross-validated set of results. To measure the significance of the results, we randomly assigned medication-status labels to the training data and again obtained ten cross-validated test set classifications based on random group labels. We compared the test set accuracy results of the ten models using true labels and ten models using randomly labeled training data with a non-parametric Kruskal-Wallis test (Section 1.5) to determine the true labels' significance [77]. We used thresholds of $P < .01$ and $P < .001$ as two levels of significance. If the true label test results were similar to the random label results and non-significant P-values were found, the labels were considered to be non-informative in the classification task.

3.3 Results

We report our classification results in Table 3.2 from feature-based kSVM and deep learning DCNN classifiers on different medications for both normal EEG and abnormal EEG populations. We report the mean and standard deviation of the 10-fold cross validation test accuracy results as a way to represent how the method best generalized on unseen data. We further report the P-value from Kruskal-Wallis tests

between the test results obtained from the true data labels and randomly labeled training data.

Classification	n**-total** (n**-test**)	**kSVM %** (P-value)	**DCNN %** (P-value)
Dilantin vs. Keppra with Normal EEG	350 (34)	51.18 ± 4.96 ($P < .790$)	59.12 ± 7.49 ($P < .061$)
Dilantin vs. Keppra with Abnormal EEG	528 (52)	$59.23 \pm 6.25^{**}$ ($P < .001$)	60.39 ± 5.03 ($P < .021$)
Dilantin vs. No medications with Normal EEG	358 (34)	$67.50 \pm 7.86^{**}$ ($P < .001$)	$64.12 \pm 8.19^{*}$ ($P < .002$)
Dilantin vs. No medications with Abnormal EEG	640 (64)	$68.91 \pm 7.18^{**}$ ($P < .001$)	$68.75 \pm 6.29^{**}$ ($P < .001$)
Keppra vs. No medications with Normal EEG	350 (34)	$60.88 \pm 4.17^{**}$ ($P < .001$)	$64.71 \pm 6.31^{*}$ ($P < .002$)
Keppra vs. No medications with Abnormal EEG	528 (52)	$70.00 \pm 4.14^{**}$ ($P < .001$)	$68.85 \pm 6.87^{**}$ ($P < .001$)

Table 3.2 Mean ± Standard deviation percent test classification accuracies and significance (P-value) for kSVM and DCNN results for six different classifications. n-total represents all subjects used in training and testing while n-test were number of subjects used exclusively for testing. Kruskal-Wallis P-value significance thresholds set at: *: $P < .01$, **: $P < .001$.

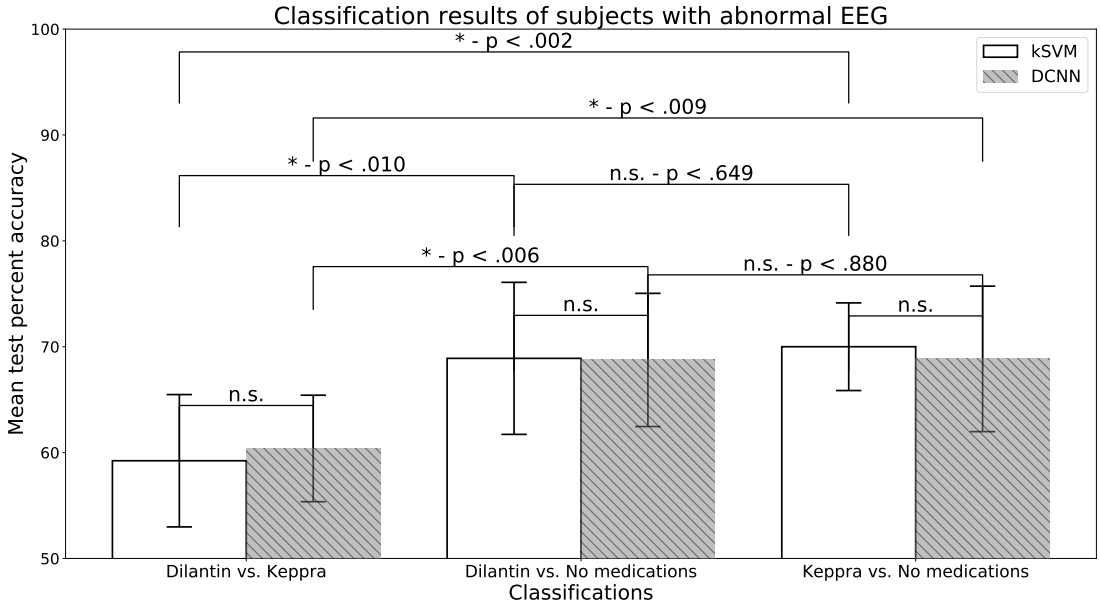


Figure 3.3 Statistical comparison of kSVM and DCNN results for subjects with abnormal EEG recordings. Kruskal-Wallis test P-value significance thresholds set at: *: $P < .01$, **: $P < .001$, or n.s.: not significant.

Results show that the kSVM and DCNN models models yielded similar levels of accuracies for all classifications. On average, the accuracy across all classifications with DCNN models were 1.37% higher than the kSVM models. Non-significant results were generally observed when comparing individuals taking Dilantin vs. individuals taking Keppra (rows 1 & 2 in Table 3.2). However, kSVM models classifying subjects with abnormal EEGs taking Dilantin vs. those taking Keppra did find significant results. For kSVM models comparing individuals taking Dilantin versus individuals taking no medications or individuals taking Keppra versus individuals taking no medications, the classification accuracy and significance of the result generally increased. With a threshold of $P < .01$, classifications from our DCNN models comparing only Dilantin versus Keppra were not found to be significant, $P < .061$

for subjects with normal EEG and $P < .021$ for those with abnormal EEG. For our kSVM model results, all results except for subjects taking Dilantin or Keppra with normal EEG, $P < 0.790$, were found to be significant with $P < .001$. We also note that the standard deviation of the kSVM results were, on average, 0.94% smaller than those from the corresponding DCNN models.

From these results we see that the classification accuracies from the kSVM and DCNN models of subjects with abnormal EEG were highest (rows 2, 4, and 6 in Table 3.2). To better assess discriminability of these classifications from the abnormal EEG data set, an isolated statistical test was performed using the vectors of classification accuracies from the 10-fold cross validation. Figure 3.3 shows the results of a Kruskal-Wallis test between these three classifications. The results of this additional statistical test showed the two anticonvulsants had the same levels of discriminability since no significant difference was found between subjects taking Dilantin vs. subjects taking no medications and subjects taking Keppra vs. subjects taking no medications ($P < .649$ for kSVM models and $P < .880$ for DCNN models). However, the accuracies from abnormal EEG recordings of subjects taking Dilantin vs. subjects taking Keppra were significantly different from both accuracies of subjects taking Dilantin vs. subjects taking no medications and subjects taking Keppra vs. subjects taking no medications ($P < .01$) (Fig 3). Finally, we see that for each classification with abnormal EEG, the kSVM and DCNN classification results were not significantly different, all $P > .62$. The implications of these results are discussed in the Discussion section.

3.4 Discussion

The goals of the current study were to (1) determine if machine learning could predict medication state solely from EEG and (2) compare the accuracies of feature-based and deep learning classification methods. A very limited number of previous studies have looked at neurological marker differences using EEG between different anticonvulsant medications in order to assess the impact of the drugs on cognitive performance and neurological patterns [90]. To our knowledge, at the time of this publication, this study presents the first aiming to distinguish different anticonvulsant medications taken by patients using solely neurological activity though machine learning methods.

We first discuss the ability of advanced machine learning algorithms to classify the medication taken using only neurophysiological activity data. To better understand the clinical implications, we compare results using abnormal EEG which were the most significant classification results. Though there were no significant differences in classification accuracies between subjects taking Dilantin vs. subjects taking no medications and subjects Keppra vs. subjects taking no medications, there were significant differences in accuracies comparing each of the anticonvulsants vs. subjects taking no medications and the classification results of subjects taking Dilantin vs. subjects taking Keppra (Figure 3.3). Our comparisons of the cross-validated results imply that the clinical effects of Keppra and Dilantin on neural activity on subjects with abnormal EEG produced similar levels of discriminability to those not taking any medications. This is not an unexpected result

given that both of these drugs are from the same class, anticonvulsant. Dilantin's (phenytoin) mechanism of action, like most anti-epileptic drugs, is believed to act by limiting bursts of neuronal activity via blockade of sodium channels that give rise to the action potential [115]. However, Keppra's (leviracetam) mechanism of action is less understood but believed to act through binding to the synaptic vesicle protein SV2A, resulting in reduced vesicle release [2]. Since anti-convulsants would be prescribed with the intent of reducing seizure activity, and since seizure activity has visible macroeffects in the EEG, it is not unexpected that they might show similar levels of discriminability. Though discriminability of each of the anticonvulsants versus subjects taking no medications were similar, further investigation would be needed to determine if the same differences were found for both medications.

Since the two medications used in this study are anticonvulsant drugs, without controlling for the clinical impression of the EEG, classifications found may have been based on the EEG pathology state rather than the differences in medications. By performing classifications on each group of subjects separately, we ensured that results found were not based on the presence of visible EEG abnormalities. Comparing the abnormal and normal EEG recordings, the abnormal EEG populations tests outperformed the normal EEG populations across all medication comparisons. This result is logical considering the medications investigated are intended to treat those with abnormal EEGs and the expectation is that these drugs would have a more prominent effect on those with abnormal EEGs recorded. While in the case of diagnosis, having the highest classification accuracy may be desirable, for medication classification, a better understanding of brain rhythms and neurological

characteristics that separate the two populations may lead to better targeted drug therapies. The classification accuracies obtained in this study, all at or below 70%, do not achieve levels usually found in diagnostic applications [127]. However, they do provide promising results of a potential avenue for inexpensive, non-invasive, and more efficient methods to determine different neurological effects from drugs. Recently several large pharmaceutical manufactures have obtained approval to market existing medications for new uses using real-world data by analyzing medical records of subjects' who had taken those medications [128] [82]. These type of retrospective analyses combined with methods used in this study have the potential to accelerate the evaluation of safety and effectiveness of medications drastically by avoiding sometimes lengthy drug trials. Furthermore, several factors suggest that the present results may represent a lower bound on future classification performance with data of this type. These factors include continual increases in EEG resolution, electrode technology, and denoising techniques since the time that many of the TUH recordings were made, as well an expected manifold increase in the data available.

We discuss next the different machine learning algorithms and the advantages and disadvantages of each for this application. In the current study we obtain novel classifications for patient populations in the NEDC data set based on medications taken. Recent works have begun to focus on comparing different machine learning methods in pharmaceutical research but there are still gaps in this area [76]. This work addresses one such gap, namely the relative performance of feature-based and deep learning model approaches. When compared to classification accuracies with randomly assigned medication status labels, we found significant results ($P < .01$)

for deep learning classifications comparing subjects taking either one of the anti-convulsants against subjects taking no medications. For our feature-based classifications we found significant classifications ($P < .01$) for all classifications except normal EEG recordings of subjects taking Dilantin vs. subjects taking Keppra, and comparing abnormal EEG recordings of subjects taking Keppra vs. subjects taking no medications. The kSVM classification accuracies were comparable to the DCNN classification accuracies, though on average slightly lower. Further, the standard deviations of the kSVM models were lower than the corresponding DCNN model results. This was expected given the stochastic nature of the optimization of DCNN models that produce larger variation in the cross-validated results in contrast to kSVM's more deterministic approach. We also found that certain kSVM models yielded lower P-values than corresponding DCNN models, sometimes with lower cross-validated classification accuracies. Therefore, we see that the DCNN models were able to obtain accuracy percentages using random labels closer to results obtained from correctly labeled data. Given the automated nature of DCNN feature extraction and recently proven efficacy in complex data sets, DCNN models are expected to be able to find patterns where labels are non-informative [27].

Both machine learning methods yielded significant and generally comparable results. To understand the clinical relevance of these classifications and medication effects on neural activity, investigation into what characteristics of EEG are important to these classifications would be important. There are several well established methods available to identify important characteristics in feature-based classifications [85]. Some of these methods were applied in this study along with

cross-validation to identify subsets of features that performed best in kSVM models. Future work may investigate which specific features were chosen in these classifications and their clinical implications. However, with respect DCNN classifications, the nature of these models limit our understanding of the important features in EEG when making classifications since the thousands, sometimes millions, of the features identified in the deep learning architecture remain a "black box". This is in stark contrast to the discrete number of known features used in feature-based algorithms. Methods to investigate deep learning models have mainly been developed in the image processing space but have rarely been used in EEG [22]. However, recently, neurological differences between sexes and EEG pathologies have been investigated by attempting to understand what deep learning models distinguished using subjects' EEGs [138] [123]. Expanding available methods of investigating deep learning models using EEG would benefit understanding the underlying neurophysiological differences. Further research in explainable artificial intelligence to classifications investigated in this work, two anti-epileptic drugs with similar target patient populations, may better inform pharmacodynamic models.

(This page intentionally left blank)

Chapter 4: Easy Perturbation EEG Algorithm for Spectral Importance (easyPEASI): A simple method to identify important spectral features of EEG in deep learning models

Overview

This chapter presents work that has been submitted for publication as a conference proceeding.

We developed a novel method of identifying frequency band importance in EEG-based deep learning classifications and applied this method to EEG data from the Temple University Health EEG Corpus.

Motivation: Recently deep learning, specifically deep convolutional neural networks (DCNN), have shown state-of-the-art results and superb effectiveness in EEG applications. Deep learning models have found such success in part because it introduced automated feature extraction as part of the end-to-end algorithm. However, in clinical applications, it may be vitally important to understand the underlying mechanisms of what was automatically learned by the deep learning models.

Approach: We obtained trained DCNN models on data to identify groups

between populations. We applied frequency domain perturbations to training data in a simple yet novel way. We next observed changes in accuracies of training models with new perturbed data. Greater drops in accuracy indicated frequency band importance.

Validation: We validated this method by identifying important neurological differences when classifying EEG pathologies. Results from this method concurred with previous findings using the same data set.

Application: We applied this method to a new application by identifying differences between two anticonvulsants, Dilantin (generic phenytoin) and Keppra (generic levetiracetam), and their individual effects on neural activity ($350 \leq n \leq 640$). We found that for subjects with normal EEG that these drugs effected the Alpha band most significantly but likely did so differently. For subjects with abnormal EEG, we found that both medications affected the Delta band when compared to non-medicated subjects. However, again these bands seemed affected differently when comparing subjects taking one of these medications. We also found the Beta band to be affected differently by the two medications.

Research impact: Results show promise for a method of obtaining explainable artificial intelligence to identify important frequency bands from EEG-driven deep learning models through a simpler more accessible method, manipulating only the input data.

4.1 Introduction

Deep learning has recently propelled the field of machine learning to new heights. More traditional machine learning models required domain experts to engineer quantitative measurements, or features, which represented the data in meaningful ways. However, deep learning models have introduced automated feature extraction as part of the end-to-end algorithm. By combining the automation of feature extraction and classification, more tailored features of the data that separate groups defined in a classification problem can be found. In applications whose primary objective it is to obtain the highest level of classification, these methods have led to state-of-the-art results without an understanding of what patterns were learned from the data [123]. However, in clinical applications, it may be vitally important to understand the underlying mechanisms of what was automatically learned by the deep learning models [37]. Take, for example, a case in which physiological signals acquired from a patient population and a healthy population are applied to a deep learning model and used for disease detection or diagnosis. By understanding the underlying physiological differences between populations classified by deep learning models, it is possible to discover potentially new biomarkers of disease or elucidate the progression and impact of a disease on those physiological functions.

Electroencephalography (EEG) is one modality that captures such physiological signals and is used both in research and clinical settings to detect or diagnose medical conditions [125]. EEG measures brain electrical signals from the scalp in a typically non-invasive and relatively inexpensive way and has been shown to be a

promising neuro-monitoring technique to identify a wide range of neuro-physiological biomarkers for many different diseases [78]. However, EEG presents complex data that has a low signal to noise ratio (SNR), thus making the data difficult to interpret. More difficult yet is understanding how to best quantify EEG using complex signal processing and mathematical methods. Because of these challenges, deep learning has recently shown promising results using EEG directly as input to distinguishing individuals based on neurological activity. However, only about one quarter of these studies investigate the underlying physiological implications on which these models are based [117].

Much like the early development of deep learning models, methods to understand and visualize these models have mainly developed and focused in the image processing domain [145]. With the adoption of these models to EEG, some of these investigation methods have also been adapted. The most common method of model inspection analyzes model weights learned. These analyses are usually limited to the first layer of the model because of the limited interpretability of subsequent filter layers [138]. Another common approach has been to investigate the model activations on representative inputs [120]. Additionally, the ideas of input perturbation and input occlusion have been explored in limited context [117]. Using established EEG-based deep learning models, an input-perturbation network-perturbation correlation map technique was developed. These methods combine both input perturbation and model weight analysis to determine the correlation between input and network characteristics [123]. When applying occlusion-based methods in image processing, parts of the input image are occluded and the effect on the network output

is analyzed to determine the most important area of the image to the model. Some research in analyzing EEG-based deep learning models have generated images that represent the EEG and subsequently applied occlusion methods on those images [135]. Although previous research is promising and allows for better understanding learned features in deep learning models, most of these methods require in depth analysis of model parameters or activations and can be difficult to interpret. Given the widespread application of deep learning in many different fields and the potential for interpretation by individuals who do not have expert knowledge in these models, the development of a simple, easily interpretable, automated method that manipulates EEG directly to determine important learned EEG features from deep learning models is needed.

This work seeks to address this need and expand upon previous methods by applying perturbation methods in image processing applications to EEG directly. There is a long history of analyzing frequency bands in EEG to understand underlying neurological mechanisms and most work in EEG has focused on understanding frequency bands of importance [99]. We introduce a novel method of applying frequency domain perturbations to EEG to understand how important specified frequency bands are on the identifying populations based solely on EEG. The ability to decipher important features in the EEG has many potential applications. Specifically, deep learning can been used to identify subjects' medication status, specifically anticonvulsant medications. To develop more informative EEG based pharmacodynamic models, it is important to determine which, if any, specific frequency bands are the underlying bases for the differences in anticonvulsant effects.

The aims of this work are to (1) introduce a novel, simpler input perturbation method for deep-learning classification problems that allows for the determination of important frequency band features in a clinical application, (2) validate this novel approach by comparing those features identified as important between our novel approach and other established approaches, and (3) apply our proposed method to patient medication status classification problems to determine which EEG frequency bands are important in distinguishing individuals taking different anticonvulsant medications or individuals taking no medications.

4.2 Methods

We first introduce the novel algorithm and applied statistical methods to determine important frequency bands for a given classification problem. We next describe the specific deep learning model used to validate the novel method and the EEG data set used for validation and application of the novel method.

4.2.1 easyPEASI: Easy Perturbation EEG Algorithm for Spectral Importance

At a high level, the proposed easy perturbation EEG algorithm for spectral importance (easyPEASI) is shown visually in Figure 4.1. easyPEASI requires that k-fold samplings of the EEG data and associated labels be obtained. These k sub-sets are used to train k binary classification deep learning models. With input data in the form of a collection of EEG channels recordings across time, the aim of this method

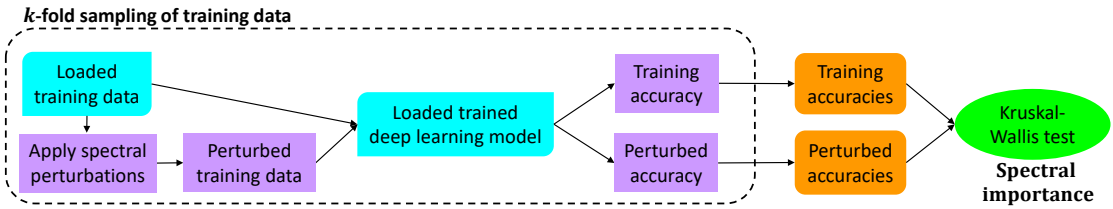


Figure 4.1 easyPEASI flowchart. Blue semi-rounded items represent pre-computed elements, purple rectangular items are intermediate elements, orange rounded items are intermediate results, and the green circle represents the last step and result. Elements in dotted area are repeated for each $1, 2, \ldots k$ samplings.

is to identify important frequency bands that distinguish the labeled data. This is accomplished by introducing perturbations (i.e. noise) to the recordings to specific frequency bands for all recorded channels for each subject. Entire frequency bands were analyzed rather than each individual frequency since these bands have known clinical meanings and power within each band may not be equally and consistently distributed across all frequencies for all recordings. Investigating single frequencies may also smear results when analyzing results across many recordings and obscure meaningful insights.

Using the Discrete Fourier Transform (DFT) for a real-valued signal, rFFT (real Fast Fourier Transform) of $EEG[channel]$,

$$\Omega = \text{rFFT}\left(EEG[channel]\right) = \begin{cases} [y[0], \text{Re}\left(y[1]\right), \text{Im}(y[1]), \ldots, \text{Re}\left(y[n/2]\right)], & n \text{ even} \\ [y[0], \text{Re}\left(y[1]\right), \text{Im}(y[1]), \ldots, \text{Re}\left(y[n/2]\right), \text{Im}\left(y[n/2]\right)], & n \text{ odd} \end{cases} \tag{4.1}$$

where

$$y[j] = \sum_{k=0}^{n-1} EEG[channel, k] \exp\left(-i \cdot \frac{j \cdot k \cdot 2\pi}{n}\right) \tag{4.2}$$

for $j = \{0, \ldots, n-1\}$, we obtain the frequency domain representation of $EEG[channel]$ [141].

The central manipulation in easyPEASI of the EEG data is the replacement of the real and complex DFT coefficients of the targeted frequency band (Section 1.2) in line 6 of the *PerturbEEGBands* function. These coefficients are replaced with random numbers generated from a Normal distribution, $\mathcal{N}(\mu, \sigma)$, where μ and σ are the mean and standard deviation (SD), respectively, of the real and complex DFT coefficients of Equation (4.1) in the targeted frequency band. Once replaced the inverse DFT, irFFT (inverse rFFT), is computed to obtain the final perturbed EEG channel. This specific spectral manipulation affects the spectral power dynamics within the frequency band, the phase coupling of the frequencies within the band and with other frequencies, and the non-stationarity of the frequencies in the band present in signal.

From this we describe the steps for our proposed method. Two parameters of choice that further need to be defined are frequency bands of interest and the number of k-fold data samplings. The frequency bands commonly analyzed in the EEG are as follows: δ (Delta) : $1-4$Hz, θ (Theta) : $4-8$Hz, α (Alpha) : $8-12$Hz, μ (Mu) : $12-16$Hz, β (Beta) : $16-25$Hz, γ (Gamma) : $25-40$Hz [125]. These frequency band bounds were used as the f_Low and f_High for each subject's EEG data as the inputs to the perturbation function, defined in Line 3 in *easyPEASI*. Further, we used a $k = 10-$fold cross-validation factor implementing this algorithm. In our outline of easyPEASI below, boldface is used for keywords, italics represent functions, underlined italics functions that are defined as part of easyPEASI, and

all others are variables.

From the k-fold cross-validated set of results we are then able to apply statistical methods. The training results are reported by the mean and standard deviation of the k training results, TrainAcc in line 6 of the *BandImportance* function. To measure the significance of each frequency band we obtained k cross-validated classifications from the trained models of the perturbations applied training data. Next, the training accuracy results of the k original and k perturbed results are compared by measuring the difference between the mean results, Δ in line 8 of the *BandImportance* function. Finally, a Kruskal-Wallis test (Section 1.5) is used to determine the statistical significance of the difference between sets of accuracy, P in line 9 of the *BandImportance* function [77]. Though $k = 10$ was used here, other values $k > 5$ can be used to meet Kruskal-Wallis test requirements. The null hypothesis of the Kruskal-Wallis test is that the medians of group of samples are the same implying that the distributions of the measurements are consistent. The H statistic was computed on ranks and a P-value obtained using a Chi-Square distribution by $Pr[\chi^2(df - 1) \geq H]$, where $df = k$ is the degrees of freedom. Therefore, if the perturbed data classification results were not similar to the original training classification results and significant P-values were found, then the effect of the perturbation on the frequency band was informative and important in the classification task. We used thresholds of $P < .01$ and $P < .001$ as varying levels of significance, though other P-value thresholds could be used.

Algorithm easyPEASI: Easy perturbation EEG algorithm for spectral importance

function PERTURBEEGBANDS(EEG,F_Low,F_High)
    for channel ∈ EEG **do**
        $\Omega \leftarrow rFFT$(EEG[channel])
        $\mu \leftarrow Mean(\Omega$[F_Low,F_High])
        $\sigma \leftarrow SD(\Omega$[F_Low,F_High])
        Ω[F_Low,F_High] $\leftarrow \mathcal{N}(\mu, \sigma)$
        EEG[channel] $\leftarrow irFFT(\Omega)$
    return EEG

function BANDIMPORTANCE(k,F_Low,F_High)
    for i $\leftarrow 0$ **to** $k-1$ **do**
        EEG, Labels $\leftarrow$ *LoadTrainData*(i)
        Model $\leftarrow$ *LoadTrainedDeepModel*(i)
        PerturbedEEG $\leftarrow$ *PerturbEEGBands*(EEG,F_Low,F_High)
        TrainAcc[i] $\leftarrow$ *EvalModel*(Model,EEG,Labels)
        PerturbAcc[i] $\leftarrow$ *EvalModel*(Model,PerturbedEEG,Labels)
    $\Delta \leftarrow \|Mean$(TrainAcc) $- Mean$(PerturbAcc)$\|$
    $P \leftarrow$ *KruskalWallisTest*(TrainAcc,PerturbAcc)
    return P, Δ

procedure EASYPEASI( )
    k $\leftarrow$ 10 ▷ k-fold samplings
    BandFreqLim $\leftarrow$ [1,4,8,16,25,40] ▷ Hz
    for b $\leftarrow 0$ **to** *length*(BandFreqLim)-2 **do**
        F_Low $\leftarrow$ BandFreqLim[b]
        F_High $\leftarrow$ BandFreqLim[b+1]
        P, $\Delta \leftarrow$ *BandImportance*(k,F_Low,F_High) ▷ Results

4.2.2 Trained deep convolutional neural network

As discussed, easyPEASI requires that trained EEG-based deep learning model and corresponding labeled training EEG data be available. We used a variant of a deep convolutional neural network (DCNN) model (Section 1.7.2) introduced in previous research and further modified for data used in this study (Section 4.5) [123]. For each binary classification, population groups were evenly balanced with recordings from each of the two labels. This was done so that there would be equal

	All recordings	Normal EEG	Abnormal EEG
Total n	35,370	6,001	15,347
Normal EEG	6,001	6,001	0
Abnormal EEG	15,347	0	15,347
EEG status not obtained	14,022	0	0
Male	15,106	2,371	7,088
Female	16,434	3,222	7,289
Sex not obtained	3,830	408	970
Age (Mean ± SD years)	50.23±19.15	41.14±18.38	52.90±19.04
(# recordings w/ age)	$(n = 33,021)$	$(5,865)$	$(n = 15,027)$
Unique subjects	13,486	4,058	7,432

Table 4.1 Population demographics of data set across recordings with EEG pathology status.

prevalence of each condition. 90% of the available data were randomly sampled and used to train the DCNN (line 2 in Train Deep Models). This process was repeated k times to obtain k trained models using different random sub-sets of data. The application was run on an NVIDIA GTX-1080Ti for training through Python 3.5 and Torch [123], [98].

Algorithm Train Deep Models

```
1: k ← 10                                              ▷ k-fold samplings
2: Ratio ← 0.9                                         ▷ Training set proportion
3: Data,Labels ← LoadData()
4: for i ← 0 to k − 1 do
5:     TrainData,TrainLabels ← SampleData(Data,Labels,Ratio)
6:     Model ← TrainDeepModel(TrainData,TrainLabels)
7:     SaveData(TrainData,TrainLabels)
8:     SaveModel(Model)
```

4.2.3 EEG data

EEG data from the Neural Engineering Data Consortium (NEDC) EEG Corpus from Temple University Hospital (TUH) version 1.0.0 were used as input to the

	Taking Dilantin	Taking Keppra	No medications
Total n	718	898	5,181
Normal EEG	179	175	1,203
Abnormal EEG	320	264	1,285
EEG status not obtained	291	459	2,693
Male	344	378	2,017
Female	320	465	1,895
Sex not obtained	54	55	1,269
Age (Mean ± SD years)	48.36±16.72	48.04±18.98	44.55±20.11
(# recordings w/ age)	($n = 692$)	($n = 862$)	($n = 4,142$)
Unique subjects	456	507	2,530

Table 4.2 Population demographics of data set across recordings with medication profiles of interest.

DCNN. This is an unprecedented real-world data set composed of thousands of subject EEG recordings collected under conditions less constrained than those typically employed in laboratory research settings [51]. This data set contains 35,370 EEG recordings sessions (1.1TB) from 13,486 subjects. Each session was accompanied by a report containing clinical impressions and patient characteristics, including age and sex, as well as the medications being taken by the patient at the time of recording. It primarily consists of data from recordings of subjects with EEG abnormalities, with many subjects taking anticonvulsant medications. Thus, the TUH data set provides the means to explore EEG pathologies and medication state classifications by evaluating the neurophysiological signals of thousands of subjects. To investigate the different frequency band importance of EEG-based classifications in these populations we applied several processing steps to the EEG data and corresponding clinical reports from the TUH data set.

Using this data set, the two most common anticonvulsant medications taken

alone based on the clinical reports were identified: Dilantin (generic phenytoin) and Keppra (generic levetiracetam) (Table 4.1 and Table 4.2). Any subject with more than one medication listed in the clinical notes was excluded from the evaluation to reduce results related to drug interaction effects. Furthermore, given the clinical nature of the data set and to avoid confounds from clinically-determined normal and abnormal EEGs, these classifications were performed separately on recordings deemed normal EEG and abnormal EEG. A subset of patient clinical reports within the TUH data set had been previously reviewed and labeled by curators of this database to be clinically normal or abnormal [83]. The definitions of normal and abnormal used by the TUH generally followed methods neurologists use to identify abnormalities [86]. These labels were applied broadly to recordings where the clinical reports identified an abnormality in the EEG and do not distinguish between different abnormalities. An automated labeling method previously developed and validated based on textual analysis of the clinical reports was applied to categorize all available recordings as normal or abnormal [95]. Applying this method to the entire TUH data set resulted in a total of 6,001 EEG recordings with normal EEG across 4,058 unique subjects and 15,347 EEG recordings with abnormal EEG across 7,432 unique subjects.

Because the recordings in the TUH data set were acquired over several years and with different EEG systems (Section 1.1), a subset of channels following a standard 10-20 electrode montage were used: Fp1, Fp2, F3, F4, C3, C4, P3, P4, O1, O2, F7, F8, T3, T4, T5, T6, Fz, Cz, Pz. Our analyses were applied to five minutes of EEG recordings across those 19 channels identified above [83]. If another

sampling rate was used, recordings were down-sampled to 100Hz. The data were then filtered through a 0.5Hz to 50Hz band pass filter, and re-referenced using a common average. All pre-processing and analyses were done in Python using a combination of available packages, published code from previous works, and custom developed code [105], [123] (available at `github.com/dbp-osel`).

4.3 Results

4.3.1 Validation: Spectral importance of EEG pathology detection

We validated easyPEASI though classifying EEG pathologies, normal EEG recordings against abnormal EEG recordings, since previous works investigating deep learning models that used the TUH data classified EEG pathology states [123]. This allows for us to compare results from the the same classification using a similar data set. We report the mean and standard deviation of the 10-fold cross validation training accuracy results as the baseline for how well the deep learning model learned classification results as a line and shaded region in Figure 4.2. With 6,001 available normal EEG recordings and 15,347 abnormal EEG recordings, balancing these two populations yields an n-total = 12,002 and using 90% of the data for training gives $n-$train=10,802. These training models generalized well through 10$-$fold cross-valued testing results with $n-$test=1,200. The training and testing results of this classification are shown in Table 4.3. We further report the differences, Δ, and P-values from Kruskal-Wallis tests between the results obtained from the training data and perturbed data where we denote *: $P < .01$, **: $P < .001$, or n.s.: not

significant in Figure 4.2.

Classification	n**-train** (n**-test**)	**Train accuracy (%)** **(Test accuracy (%))**
Normal EEG vs. Abnormal EEG	10,802 (1,200)	80.99 ± 2.22 (80.40 ± 2.41)

Table 4.3 Classification accuracies (Mean ± Standard deviation %) for subject EEG pathologies.

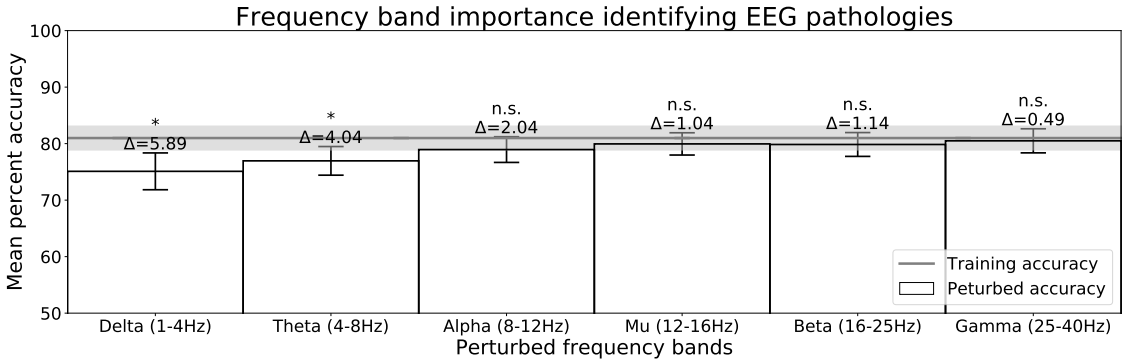


Figure 4.2 easyPEASI results for models detecting EEG pathologies. Each bar represents the classification accuracy from the DCNN model when a particular frequency band was perturbed using the easyPEASI. Line and shaded area at 80.99% ± 2.22 are mean and standard deviation of training accuracies with no perturbed data. Kruskal-Wallis P-value significance thresholds set at: n.s.: not significant, *: $P < .01$, **: $P < .001$.

As in previous work identifying band importance, we find that the Delta and Theta bands were most distinctive between populations when classifying EEG pathologies [123], instilling confidence that our novel method to identifying these bands is valid.

4.3.2 Application: Anticonvulsants effects on brain rhythms

Given the validation of this novel algorithm, we next report the results of applying easyPEASI to determining those frequency bands most important to med-

ication state classifications for anticonvulsants. These training data sets and trained models are made available for rapid reproducibility (Section 4.5). In this work we focus on training accuracies for comparison to perturbed data accuracy values found with easyPEASI. We began by comparing recordings of subjects taking either only Dilantin or only Keppra. We further separated subjects based on whether they had abnormalities present in their EEG recordings. Using 90% of the data for training, recordings of subjects with normal EEG taking Dilantin vs. Keppra we report training and testing accuracies in Table 4.4 and show easyPEASI results in Figure 4.3. These models obtained results above 50%, which is considered chance for a binary classification. Applying easyPEASI to the training data, we show the importance of each band in the two classifications identifying anticonvulsants.

Classification	n**-train** (n**-test**)	**Train accuracy (%)** (**Test accuracy (%)**)
Dilantin vs. Keppra with Normal EEG	316 (34)	80.19 ± 13.86 (59.12 ± 7.49)
Dilantin vs. Keppra with Abnormal EEG	476 (52)	83.91 ± 3.48 (60.39 ± 5.03)

Table 4.4 Classification accuracies (Mean ± Standard deviation %) for recordings of subjects taking one of two anticonvulsants, either Dilantin or Keppra, with normal or abnormal EEGs.

In the case of normal EEG recordings, no frequency bands were found to be significant in differentiating between the effects of the two anticonvulsants Dilantin and Keppra, though we note the Alpha band was found to have a $\Delta = 16.80\%$, which was 10% larger than any other frequency band. Further, for abnormal EEG recordings we found Theta, Alpha, Delta and Beta bands were most important for

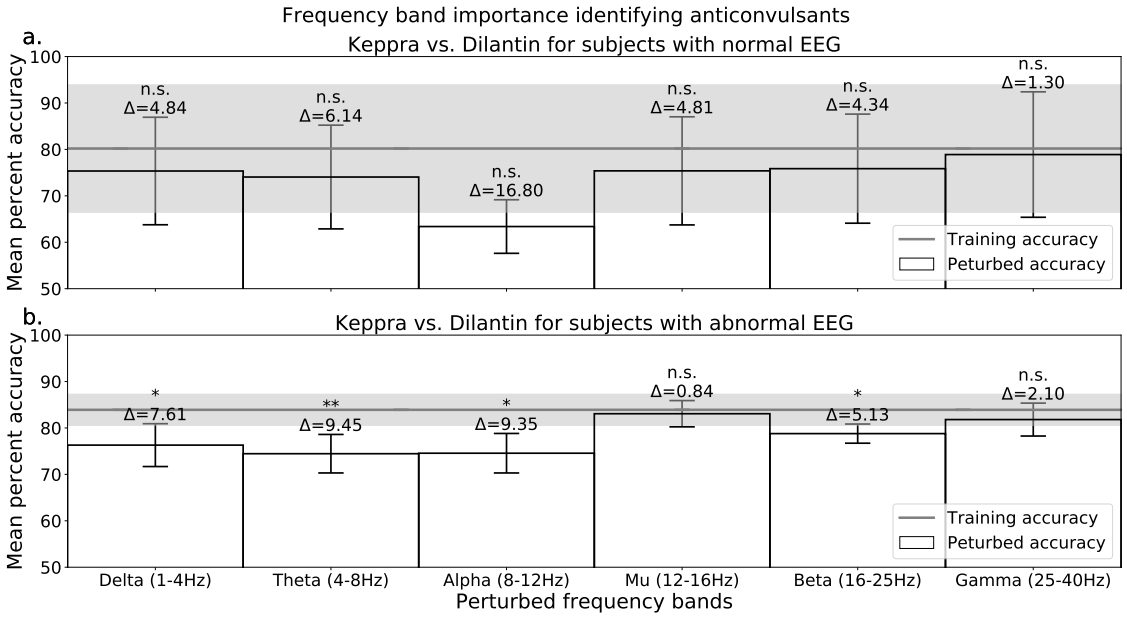


Figure 4.3 easyPEASI results for models comparing the differences between anticonvulsants for normal and abnormal EEG recordings. Each bar represents the classification accuracy from the DCNN model when a particular frequency band was perturbed using the easyPEASI. Lines and shaded areas at (a) $80.19\% \pm 13.86$ and (b) $83.91\% \pm 3.48$ are means and standard deviations of training accuracies with no perturbed data. Kruskal-Wallis P-value significance thresholds set at: n.s.: not significant, *: $P < .01$, **: $P < .001$.

this classification.

We then evaluated the bands' importance to distinguishing recordings of subjects taking either one of the two anticonvulsants against subjects taking no medications. Again, classifications and easyPEASI were applied separately for abnormal and normal EEG recordings. Table 4.5 shows the mean and standard deviation of the training and testing accuracies obtained, with easyPEASI results in Figure 4.4 and Figure 4.5. Table 4.5 shows that testing classification results were above chance. Applying easyPEASI, the important bands for these classifications are shown in Figure 4.4 for normal EEG recordings and Figure 4.5 for abnormal EEG recordings.

Classification	n**-train** (n**-test**)	**Train accuracy (%)** (**Test accuracy (%)**)
Dilantin vs. No medications with Normal EEG	324 (34)	82.22 ± 8.24 (64.12 ± 8.19)
Keppra vs. No medications with Normal EEG	316 (34)	85.82 ± 5.83 (64.71 ± 6.31)
Dilantin vs. No medications with Abnormal EEG	576 (64)	84.36 ± 7.21 (68.75 ± 6.29)
Keppra vs. No medications with Abnormal EEG	476 (52)	82.69 ± 5.86 (68.85 ± 6.87)

Table 4.5 Classification accuracies (Mean ± Standard deviation %) for recordings of subjects taking one of two anticonvulsants and subjects taking no medications.

The results for recordings of subjects taking a single medication and subjects taking no medications show that for normal EEG recordings, both anticonvulsants affected the Alpha band most, followed by the Theta band. The Beta band was also found to be significant for recordings of subjects with normal EEG taking Keppra against subjects taking no medications. In the cases of abnormal EEG recordings, both classifications comparing subjects taking anticonvulsants and subjects taking no medications, the Delta band was found to be most significantly affected. However, the Theta and Alpha bands were only significant when distinguishing recordings of subjects with abnormal EEG taking Dilantin against those taking no medications.

4.4 Discussion

In the current study, we introduce and validate a method (easyPEASI) to investigate deep learning models that use EEG as their input to identify important frequency bands in a given binary classification problem and provide an example of

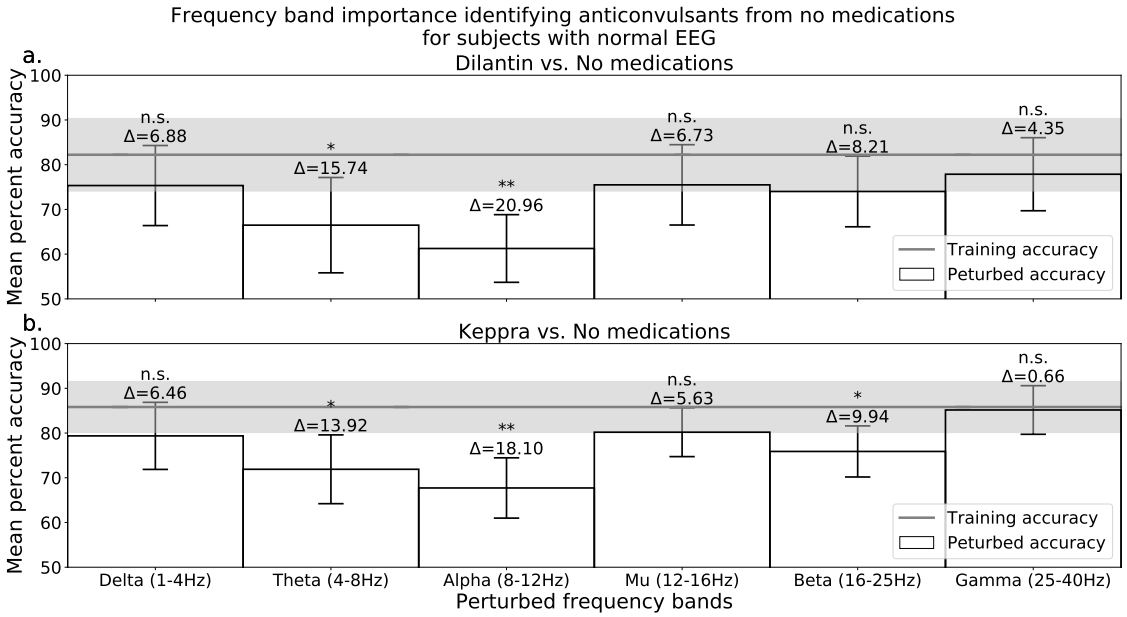


Figure 4.4 easyPEASI results for models comparing the differences for recordings of subjects with normal EEG taking one anticonvulsant and subjects with normal EEG taking no medications. Each bar represents the classification accuracy from the DCNN model when a particular frequency band was perturbed using the easyPEASI. Lines and shaded areas at (a) $82.22\% \pm 8.24$ and (b) $85.82\% \pm 5.83$ are means and standard deviations of training accuracies with no perturbed data. Kruskal-Wallis P-value significance thresholds set at: n.s.: not significant, *: $P < .01$, **: $P < .001$.

how this algorithm can be applied to real clinical classifications problems. Unlike previous methods that have applied some type of manipulation on the input data and associated those manipulations to changes in deep learning network weights and activations, our method only requires applications of perturbations to the input data to determine frequency band importance. Our work focuses on frequency bands because of its prominence in prior work to investigate underlying neuro-physiology through EEG [99].

We first validated our method by identifying frequency bands most important in classifying EEG pathologies, as done with other methods [123]. With these re-

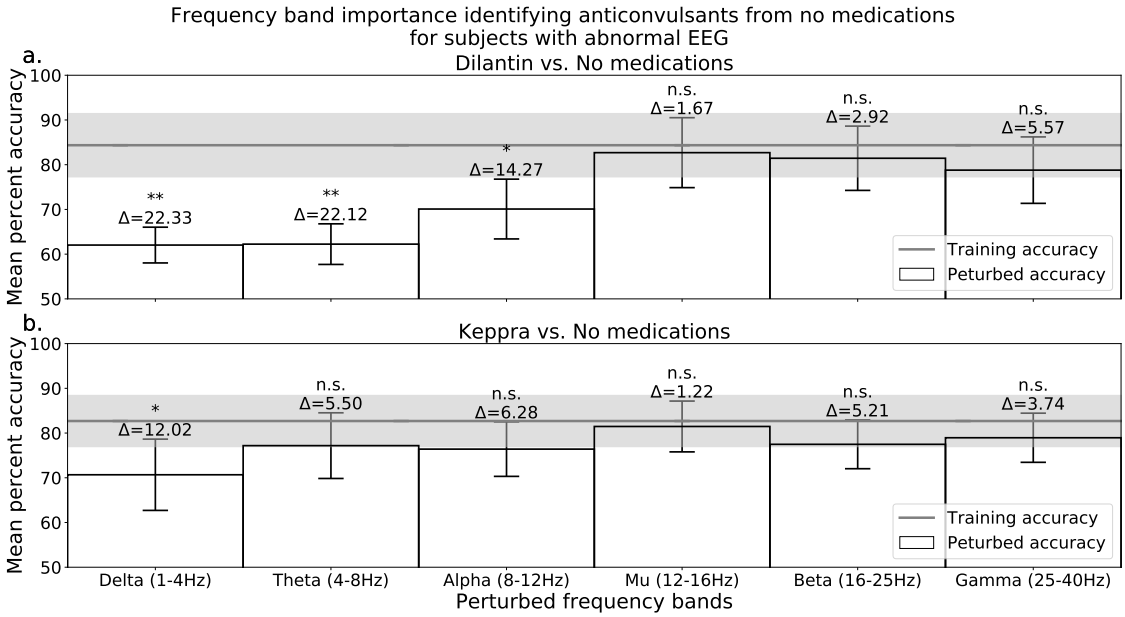


Figure 4.5 easyPEASI results for models comparing the differences for recordings of subjects with abnormal EEG taking one anticonvulsant and subjects with abnormal EEG taking no medications. Each bar represents the classification accuracy from the DCNN model when a particular frequency band was perturbed using the easyPEASI. Lines and shaded areas at (a) $84.36\% \pm 7.21$ and (b) $82.69\% \pm 5.86$ are means and standard deviations of training accuracies with no perturbed data. Kruskal-Wallis P-value significance thresholds set at: n.s.: not significant, *: $P < .01$, **: $P < .001$.

sults we see that using easyPEASI to investigate EEG-driven deep learning models shows promise in identifying important frequency bands in classification tasks. This method allows for the investigation of important spectral characteristics of EEG learned by deep learning models without the need to directly investigate the deep learning model itself. The method introduced and validated in this work is simpler than previous methods since it does not require investigating the deep learning model and rather only relies on manipulation of the input data. The interpretation of the results is also based upon a commonly used, well-known statistical method, the Kruskal-Wallis test. Therefore, easyPEASI may be more accessible to

researchers in the field of neuroscience that may not be experts in deep learning but often require explainable artificial intelligence and explainable models to understand clinical implications. The spectral perturbation that the *PerturbEEGBand* function in easyPEASI applies to the EEG signal is simple in nature and affects particular properties of the signal. Replacing the real and complex coefficients of the DFT with normally distributed random values affects the peaks and troughs of the spectral power profile. Further, when applying the perturbation to the data, the phase coupling and non-stationarity of the EEG signal is affected. Therefore, it is possible that any combination of these properties of each frequency band may have been most influential when significant bands were identified by easyPEASI. Along with varying the number of data samplings k used or frequency bands investigated, different perturbations that manipulate other specific properties of frequencies may be investigated in future works with easyPEASI.

Identification of EEG pathologies through deep learning models using solely EEG as inputs achieved accuracies above 80% using about 3,000 EEG recordings [123]. Our automated EEG pathology labeling method allowed for classification of EEG pathologies with deep learning using 12,002 EEG recordings, the largest EEG data training and testing sets to classify EEG pathologies. Using this data set also achieved testing accuracy greater than 80% and with easyPEASI identified the same bands important to the classification (Figure 4.2). These levels of testing accuracy allow for deep learning models for classification to be potentially used in diagnostic applications [127]. However, these levels of accuracy have yet to be achieved using deep learning models to identify subject's medications using EEG,

but have shown the ability to distinguish these populations above chance (50%). This means that these models generalized to unseen data to a certain degree and identified meaningful patterns in the data. Concurrent research outside the scope of this work is in progress that analyses and discusses the significance and impact of these classification results (Table 4.4 and Table 4.5).

As a novel application of easyPEASI, we investigated EEG recordings from individuals taking different anticonvulsant medications and no medications. To our knowledge, this kind of analysis into neurophysiological differences for this drug class is not well researched. Here, we discuss the results of the easyPEASI method and correlate these results with expectations based on clinical literature. For normal EEG recordings, though not statistically significant, the largest differences in effects between medications were found to be in the Alpha band (Figure 4.3a). Comparing recordings of subjects with normal EEG taking a single medication (i.e. Keppra or Dilantin) to subjects taking no medications, the Alpha band was found to be significantly different (Figure 4.4). With the Alpha band the most important band comparing normal EEG recordings, it seems that anticonvulsants affect this band in a similar manner when compared to recordings of subjects taking no medications since it is not significant when compared to each other (Figure 4.3a). Similarly, the Theta band seems to be equally important (Figure 4.4) and comparably affected by these medications (Figure 4.3a). Finally, we note that though the Beta band was found to be important in only classifying recordings of subjects with normal EEG taking Keppra against subjects taking no medications (Figure 4.4b), the Beta band was not found to be important classifying the two medications for normal EEG

recordings (Figure 4.3a).

For abnormal EEG recordings, results showed both anticonvulsant medications affected the Delta band (Figure 4.3a and Figure 4.5). Further, the Theta and Alpha bands were found to be important when classifying recordings of subjects with abnormal EEGs taking Dilantin against those taking no medications (Figure 4.5a). These results are consistent with known effects of EEG pathologies [123]. Delta and Theta bands have been important for identifying EEG pathologies, so it is expected that the medication aimed to treat epilepsy, one of the most common pathologies, would affect these bands for abnormal EEG recordings [86]. When comparing these medications to each other, Dilantin's mechanism of action, like most anticonvulsant medications, is primarily attributed to modulating voltage-gated sodium and calcium channels and enhancement of $GABA_A$ (gamma aminobutyric acid, type A) [115]. However, Keppra's mechanism of action is less understood but is thought to be through binding to the synaptic vesicle protein SV2A [2]. With these differences, our finding that the Beta band was uniquely differently affected by these two medications (Figure 4.3b) is consistent with known neurophysiology since the Beta band is known to be affected by GABA and is only known to be directly affected by one of these medications [67]. Further, since they have different mechanisms of action, though they may both be affecting the Delta bands, they may be doing so differently since we observe the Delta band was important when comparing anticonvulsants to each other (Figure 4.3b). However, we also found the Theta and Alpha bands to be important when distinguishing these medications from each other for abnormal EEG recordings (Figure 4.3b) and were only found to be important comparing recordings

of subjects taking Dilantin to those taking no medications (Figure 4.5a).

A limitation of this method, however, is that the perturbations were applied to all channels together. Future development of extensions to easyPEASI could be done to identify brain regions that are important in classification tasks. Regarding the applications investigated in this work, future research may investigate the difference between subjects with abnormal EEG taking anticonvulsants against subjects with normal EEG taking no medications to potentially determine anticonvulsants' efficacy.

4.5 Online resources for reproducibility

Software and scripts for reproducibility can be found at our GitHub Repository: `github.com/dbp-osel`.

Software implementation for easyPEASI and analysis of this work are at: `github.com/dbp-osel/easyPEASI`. This repository also includes saved trained deep learning models for all classifications reported here. For the application area of medication classifications, data and labels are also made available. These resources allow for fast reproducibility of results presented in this work.

For other classifications using the TUH EEG Corpus to reproduce results obtained here (e.g. larger EEG pathology detection data set) several pre-processing steps need to be applied the data before these deep learning models can be trained and subsequently applied to easyPEASI. These pre-processing tools and data are available as part of work previously published: `github.com/dbp-osel/qEEG-consistency`

[95].

The open source EEG data is available from Temple University Health at `www.isip.piconepress.com/projects/tuh_eeg` [51].

(This page intentionally left blank)

Chapter 5: Quantifying Signal Quality from Unimodal and Multimodal Sources: Application to EEG with ocular and motion artifacts

Overview

This chapter presents work has not been submitted for publication. It is intended to be submitted partially or in full for publication.

We developed novel methods of scoring physiological signal quality for data and noise from unimodal source and multimodal sources. We applied these methods to score EEG data from two experimental data sets from the University of Houston.

Motivation: Unlike in clinical or research settings, electrophysiological data collected in the real world is often contaminated with noise that does not represent the physiological signal of interest. For applications where we need these data to be reliable, consistent, and informative, the presence of noise that corrupt the signal of interest can degrade the effectiveness of diagnostic tools and brain-machine interfaces. If these data are to be useful, it is important to understand signal quality which would enable more meaningful measurements from patients.

Approach: We developed two methods to evaluate quality of electrophys-

iological signals with a continuous score that ranged from zero to one. Our first method targeted cases when the data and noise were unimodal, recorded from the same source (e.g. EEG and ocular artifacts from EOG). We used a Bayesian approach to estimate distributions of features of signals free from artifacts and noise signals. Estimated distributions that met our inclusion criteria were then used to score new signals. Our second method was designed for cases when the data and noise were multimodal, recorded from different sources (e.g., EEG and motion artifacts from IMUs). For this case, we used a deep learning-based approach to train models to identify differences between signals with noise from artifacts present and signals without noise. Trained models were then used to score new signals.

Validation: We validated our first approach by successfully developing the models and scoring unseen EEG data with ocular noise ($192 \leq n \leq 2,784$). We next validated our second approach by training models to identify motion artifacts in EEG and appropriately score validation data ($n = 548$).

Application: We further applied our first method to quantifying the effectiveness of EEG artifact removal algorithms targeting ocular noise ($n = 2,784$). We tested two artifact removal algorithms commonly used and available in EEGLAB. EEG recordings were scored before and after application of the algorithms and observed the differences in scores between the two algorithms.

Research impact: Results allow for rapid evaluation of signal quality of EEG data. Models are easily extendable to other physiological measurements where noise is often present. This work also provides a novel data-driven method to quantitatively evaluate and compare effectiveness of noise removal algorithms with real data.

5.1 Introduction

Advancements in and availability of wearable technologies that can readily collect physiological data from individuals in both controlled laboratory and real-world settings have been growing rapidly. As such, both the volume of available observational biometric data and its potential utility, if properly understood, are also increasing. If these data are to be effectively applied and correctly interpreted, it is important to understand the quality of data being recorded. We define quality by how much of the acquired signal is from the source of interest and not noise from external or internal (i.e. other physiological) sources. Unlike in clinical or research settings, electrophysiological data collected in the real world is often contaminated with noise that does not represent the physiological signal of interest.

With the ability to directly acquire signals from noise sources, there have been greater successes in applying artifact removal algorithms when the noise signal is known [73], [72]. In many situations, it may not be possible to directly measure the source of artifacts, making the process by which those artifacts are removed arduous and more error-prone. Depending on the application, it would be beneficial to evaluate the quality of data and know how clean electrophysiological recordings are before attempting to analyze it or use it as input to a model. To better identify data quality where noise sources are not available, we can leverage data from studies with the appropriate data to generate models that characterize and score electrophysiological recordings.

In the case of electrocardiography (ECG), an example of an electrophysiologi-

cal signal, recordings often include electromyography (EMG), and movement, among other noise sources. For ECG, there have been several efforts in developing methods to assess signal quality [121]. However, for another type of electrophysiological signal, electroencephalography (EEG), there has been little research on developing quality metrics. EEG, which measures brain electrical signals from the scalp, is a common neuro-monitoring technique used in both clinical and research settings. EEG is typically non-invasive, relatively inexpensive, and has been used for diagnostic, rehabilitative, and therapeutic purposes [78], [106]. For clinical applications where we need these data to be reliable, consistent, and informative, the presence of noise that corrupt the signal of interest can degrade the effectiveness of diagnostic tools and brain-machine interfaces. Despite their increasing use and public health importance, little is known about the quality of EEG with respect to neural activity content. If these data are to be useful, it is important to understand EEG signal quality which would enable more meaningful measurements from patients to identify targeted neurological activity in EEG. With EEG, ocular activity (measurable by electrooculography (EOG)), muscle activity (measurable by EMG), cardiac pulses (measurable by ECG), and movement (measurable by inertial measurement units (IMU)) can often corrupt the purity of neural activity that is targeted by EEG recordings [63]. Creating metrics to determine the quality of non-invasive electrophysiological recordings would inform those using the data how representative it is of the desired physiological source and not riddled with noise from sources not of interest. Human EEG data is a particularly important data type in this regard because of its relevance to a myriad of public health challenges [78].

There have been a few approaches to scoring EEG signal quality. One method calculated 11 different features of EEG that were used to identify clean EEG recording segments by thresholding these signal parameters [30]. Data across four data sets were used to determine these thresholds with $n = 58$. Based on these thresholds, features values from new EEG signals were categorized as either clean or not clean. In another study, three quantitative EEG features were used to assess signal quality to obtain three scores which were combined into one score [56]. The data used was from the OPTIMI data set with $n = 90$, but the method may need modification to be generalized to other headsets. Recently, machine learning was applied to this problem using 114 features from the EEG [48]. The EEGs were classified using several classification approaches, along with feature selection and a five-fold cross validation into three quality levels: low, medium, and high. This study used EEG across five data sets with $n = 43$.

The aims of this work are to (1) develop a scoring method for data from a unimodal source when the noise can be measured directly from the same modality and apply it to EEG with ocular artifacts, (2) develop a scoring method for data when the noise can only be measured from another modality, requiring multimodal sources, and apply it to EEG with motion artifacts, and (3) apply our developed scoring metric to evaluate artifact removal algorithms, specifically comparing two artifact removal algorithms that target ocular artifacts.

This work proposes new methods to create a metric to quantify quality of electrophysiological data. Our first proposed approach is targeted at applications when the noise source can be recorded directly using the same measurement tool, i.e. uni-

modal data such as EEG and EOG, which are both recorded from electrodes on the head. Our second approach would be needed when there are noise sources that cannot be recorded directly and can only be quantified by other means, i.e. multimodal data such as EEG and motion, which require both EEG electrodes and IMUs. We propose a feature-based Bayesian approach to score EEG with ocular artifacts since EEG and EOG can be directly measured through same set of electrodes. Recently deep learning, specifically deep convolutional neural networks (DCNN), have shown state-of-the-art results and superb effectiveness in EEG applications [117]. As such, we next present a deep learning-based approach to score EEG with motion artifacts since motion cannot be directly recorded with electrodes but rather is quantified by IMU or other motion tracking tools.

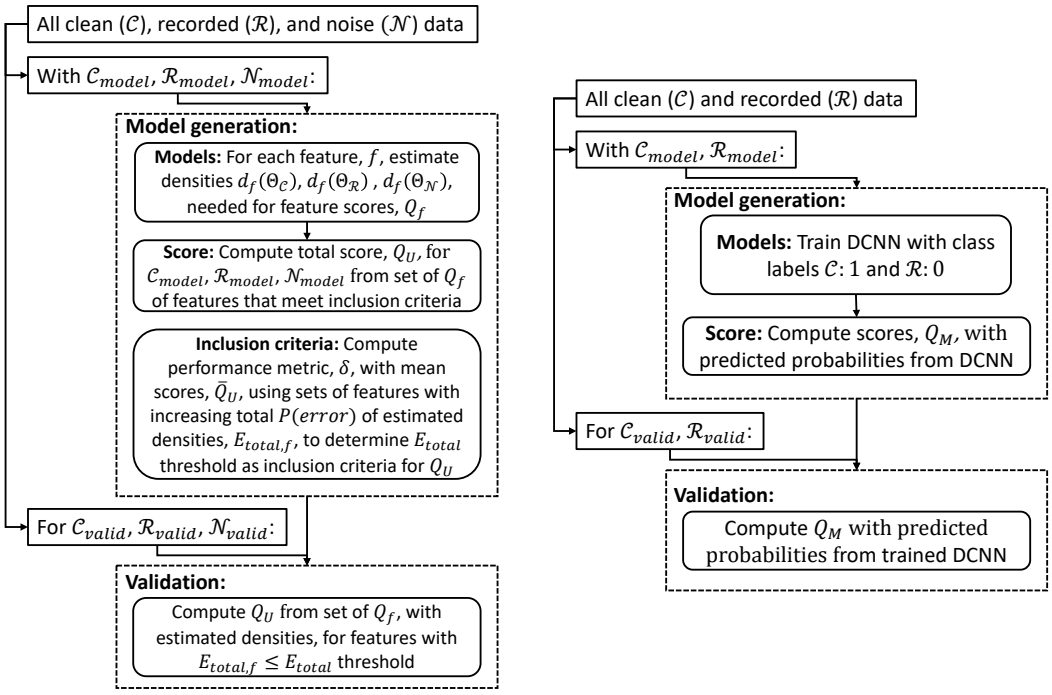
Further, we validate and apply our scoring metric to evaluate the effectiveness of different artifact removal algorithms. We hypothesize that data cleaned with other artifact removal algorithms will obtain higher scores than before they were processed. Methods to compare the performance of EEG artifact removal algorithms have not been well developed and currently rely on either visual inspection or synthetic data [63]. Since ocular artifacts are the most common noise targeted by artifact removal algorithms for EEG, we score recorded data with noise present and data after being processed by different ocular artifact removal algorithms [68]. These scoring methods could be used to evaluate the effectiveness of noise removal algorithms by comparing scores of EEGs processed by different methods.

5.2 Methods

We first introduce a scoring method for cases when the noise source can be recorded directly by the same modality (Section 5.2.1). We next describe a scoring method when the noise source cannot be measured directly through the same recording modality (Section 5.2.2). Both methods are designed to generate a score $0 \leq Q \leq 1$ (Q_U for data with unimodal source and Q_M for data with multimodal sources). A score of zero would imply that the data is entirely noise or from sources not of interest, while a score of one would mean that the data is entirely from the desired electrophysiological source. Figure 5.1 shows a high-level processing pipeline of model generation and validation of scoring methods for data with unimodal (Figure 5.1a) and multimodal (Figure 5.1b) sources. Finally, we outline two different noise removal algorithms and how our scoring method could be used to evaluate their performance (Section 5.2.3).

5.2.1 Scoring data with unimodal source

In cases where the noise source can be measured directly from the same recording modality (referred to as 'unimodal method'), it is possible to compare quantitative features of both the signal of interest and the noise. After computing these features and their corresponding scores, we detail how to identify which features would be most effective to score data quality.



(a) (b)

Figure 5.1 Flowcharts of model generation and validation of scoring methods for data with (a) unimodal and (b) multimodal sources.

5.2.1.1 Scoring method for data with unimodal source

We begin by computing features for data without noise, hereafter referred to as clean data ($\mathcal{C}$), recorded data with noise, referred to as recorded data ($\mathcal{R}$), and the noise source, referred to as noise data ($\mathcal{N}$).

With the collection of quantitative features for each recording, we fit a distribution for each feature for each type of data, (clean, recorded, and noise) through kernel density estimations (KDE) (Section 1.6). Our KDE based models are computed using Gaussian kernels and Scott's rule for bandwidth size. For each source of data and feature, f, a set of parameters, Θ, and subsequently distributions $d_f(\Theta_C)$, $d_f(\Theta_R)$, and $d_f(\Theta_N)$, are estimated.

To obtain a sub-score from each feature, we use the Bayesian decision critical value, v_f^*, which minimizes the probability of error between each set of estimated distributions, $d_f(\Theta_{\mathcal{C}})$ and $d_f(\Theta_{\mathcal{N}})$ [32]. For each feature used, if the mean value of $\mathcal{N}$ is less than the mean value of $\mathcal{C}$, a sub-score using that recording's feature's value, v_f, is obtained by

$$Q_f(v_f) = \begin{cases} \frac{1}{2} + \frac{1}{2}P(v_f > x_i \mid x_i \in \mathcal{C}), \text{ if } v_f \geq v_f^* \\ \\ \frac{1}{2}P(v_f \leq x_i \mid x_i \in \mathcal{N}), \text{ if } v_f < v_f^* \end{cases} \tag{5.1}$$

In essence, if $v_f \geq v_f^*$ then $Q_f(v_f)$ represents the proportion of values in $\mathcal{C}$ less than v_f scaled between $[0.5, 1]$, or if $v_f < v_f^*$ then $Q_f(v_f)$ represents the proportion of values in $\mathcal{N}$ less than v_f scaled between $[0, 0.5]$. If the mean value of $\mathcal{N}$ is greater than the mean value of $\mathcal{C}$, then the inequalities in Equation 5.1 are reversed appropriately. To obtain the final quality score for our unimodal method a set of features for a recording, V,

$$Q_U(V) = \frac{1}{F}\sum_{f=1}^{F} Q_f(v_f), \; v_f \in V \text{ for features } f \text{ that meet inclusion criteria,} \tag{5.2}$$

where F is the number of computed features used to obtain the overall score.

We next develop a method to identify an inclusion criteria for which features would be best for determining Q_U. We compute three probabilities of errors for each feature for each set of data as follows:

$$P(error|\mathcal{C}, f) = P\left(d_f(x_i|\Theta_{\mathcal{C}}) < d_f(x_i|\Theta_{\mathcal{N}}) \mid x_i \in \mathcal{C}\right) \tag{5.3}$$

$$\begin{aligned} P(error|\mathcal{R}, f) =& P\left(d_f(x_i|\Theta_{\mathcal{C}}) < d_f(x_i|\Theta_{\mathcal{N}}) \mid x_i \in \mathcal{R}\right) \\ =& 1 - P\left(d_f(x_i|\Theta_{\mathcal{N}}) \leq d_f(x_i|\Theta_{\mathcal{C}}) \mid x_i \in \mathcal{R}\right) \end{aligned} \tag{5.4}$$

$$P(error|\mathcal{N}, f) = P\left(d_f(x_i|\Theta_{\mathcal{N}}) < d_f(x_i|\Theta_{\mathcal{C}}) \mid x_i \in \mathcal{N}\right) \tag{5.5}$$

Ideally, the clean and noise distributions should be completely separable, with a probability of error of zero, while the distribution of recorded data should be a combination of values from the clean and noise distribution and thus should have a probability of error of 0.5 between both the clean and noise distributions. Thus, we can evaluate the utility of each feature by computing the error of each estimated distribution from the ideal error, referred to as total error. We define the total error for each feature as

$$E_{total,f} = P(error|\mathcal{C}, f) + |0.5 - P(error|\mathcal{R}, f)| + P(error|\mathcal{N}, f) \tag{5.6}$$

where $0 \leq E_{total,f} \leq 2.5$.

Lower $E_{total,f}$ represent features best suited for scoring signal quality and only features with low enough $E_{total,f}$ should be used. Finally, we formulate a metric to

determine the inclusion criteria of how low $E_{total,f}$ of all features should be, E_{total} threshold. We define a measure of the error from the ideal solution,

$$\delta = \left|0.75 - \overline{Q}_U(\mathcal{C})\right| + \left|0.50 - \overline{Q}_U(\mathcal{R})\right| + \left|0.25 - \overline{Q}_U(\mathcal{N})\right| \tag{5.7}$$

where $\overline{Q}_U$ is the mean $Q_U(V)$ across all data of each type, $V_\mathcal{C} \in \mathcal{C}$, $V_\mathcal{R} \in \mathcal{R}$, and $V_\mathcal{N} \in \mathcal{N}$, features from clean data, recorded data, and noise data, respectively.

To interpret Q_U from this method effectively we wish to have the mean of $Q_U(V_\mathcal{C})$, $\overline{Q}_U(\mathcal{C})$, be 0.75 and have range between 0.5 and 1, the mean of $Q_U(V_\mathcal{R})$, $\overline{Q}_U(\mathcal{R})$, to be 0.50 and have range between 0.25 and 0.75, and the mean of $Q_U(V_\mathcal{N})$, $\overline{Q}_U(\mathcal{N})$, to be 0.25 and have range between 0 and 0.50. These constraints and parameters make it such that ideally there will be no overlap between $\mathcal{C}$ and $\mathcal{N}$ while $\mathcal{R}$ will overlap approximately half with $\mathcal{C}$ and half with $\mathcal{N}$.

Therefore, to obtain an optimal E_{total} threshold, we calculate δ with increasing values of E_{total} thresholds to observe when δ begins to increase. From this analysis, we only use features with $E_{total,f}$ lower than the determined threshold value as an inclusion criteria for Equation 5.2.

5.2.1.2 Data with unimodal source

As the unimodal approach is data-driven, we present here data used to generate scoring parameters and subsequently validate this method. We focus on noise from eye-movement, EOG, since for high density EEGs they are generally captured directly by electrodes placed near the eyes.

A data set obtained from the University of Houston contained EEG and EOG recordings (sampled at 100 Hz) as well as motion capture from eleven subjects walking on a treadmill for six minutes [72]. Of these eleven subjects, eight were used for this approach because of the availability and consistency of data. These EEG were recorded with a 64 electrode array following labels from the extended 10-20 system (Section 1.1) with: channel FT9 used as ground; FT10 for reference; TP9, TP10, PO9, and PO10 for EOG; and the 58 remaining channels for EEG. For three subjects: T7 was used as ground; T8 for reference; TP9, TP10, FT9, and FT10 were used for EOG; and the 58 remaining channels for EEG.

A robust noise removal method developed by the University of Houston research group directly used available noise sources to remove them from the recorded EEG [73]. This noise removal algorithm targeting EOG noise used an H^∞ filtering formulation (Section 1.4.2) since it guarantees robustness where small modeling errors and external noise do not cause large estimation errors [52]. This algorithm and subsequent study used four EOG channels directly recorded as reference disturbance input. The strength and effectiveness of this algorithm are shown to out-perform other common ocular artifact removal techniques [72].

The H^∞ EOG cleaning algorithm was applied to the six minutes of recording for the eight subjects. We then separated data from EEG channels and EOG channels. This gave us 58 channels with six minutes of recording of both clean EEG data and recorded EEG, as well as four channels with six minutes of EOG data. For EEG, we used 30 features on segmented data of one-minute epochs since these features have been shown to be stable with these higher epoch lengths in previous

quantitative EEG studies [95]. This yielded $\mathcal{C}_{Eye}$ and $\mathcal{R}_{Eye}$ with $n = 2784$ (58 channels $\times$ 6 minutes $\times$ 8 subjects $= 2784$), and $\mathcal{N}_{Eye}$ with $n = 192$ (4 channels $\times$ 6 minutes $\times$ 8 subjects $= 192$), where each sample was of size 30×1 (30 features), for our method scoring data with unimodal source.

5.2.1.3 Model generation for data with unimodal source: EEG with ocular artifacts

To obtain scoring models for this data we separated 90% of the data randomly to generate the models and reserved 10% of the data to test and validate the results. We present ten-fold cross-validated values from ten generated models from different samplings of 90% of the data, denoted with the subscript *'model'*, that resulted in $\mathcal{C}_{Eye,model}$ and $\mathcal{R}_{Eye,model}$, both with $n = 2506$, and $\mathcal{N}_{Eye,model}$, with $n = 173$. We report the mean results as well as the mean standard deviation across samples for the ten-fold cross-validation.

As mentioned, we computed 30 features (Section 1.3) for both EEG and EOG data. For spectral features, the Fourier transforms were taken on the pre-processed recording after which various spectral features were computed. The power spectral density (PSD) (Section 1.2) of frequency bands commonly analyzed in the EEG were estimated using the periodogram. The ranges of the frequency bands applied in this study were as follows: δ(delta) : $1 - 4$Hz, θ(theta) : $4 - 8$Hz, α(alpha) : $8 - 12$Hz, μ(mu) : $12 - 16$Hz, β(beta) : $16 - 25$Hz, γ(gamma) : $25 - 40$Hz [125]. Both absolute powers and relative powers were computed, with relative power equal to the power

in a frequency band divided by the total power. The entropy of the periodogram, and entropy of the normalized periodogram, were found using the Shanon entropy definition [14]. In addition to the spectral features, the following time domain features, directly from the pre-processed EEG signal, were computed: entropy of the normalized signal, mean thresholded Lempel-Ziv complexity (LZC), minimum value, maximum value, median, mean, variance, standard deviation, skew, kurtosis, curve length, energy, non-linear energy, sixth power, sum, mobility, complexity.

To identify the appropriate E_{total} threshold for the set of Q_f that will be used to calculate Q_U (Equation 5.2), we analyze the relationship between the E_{total} threshold (Equation 5.6) and $\overline{Q}_U$ along with their associated δ (Equation 5.7) in Figure 5.2.

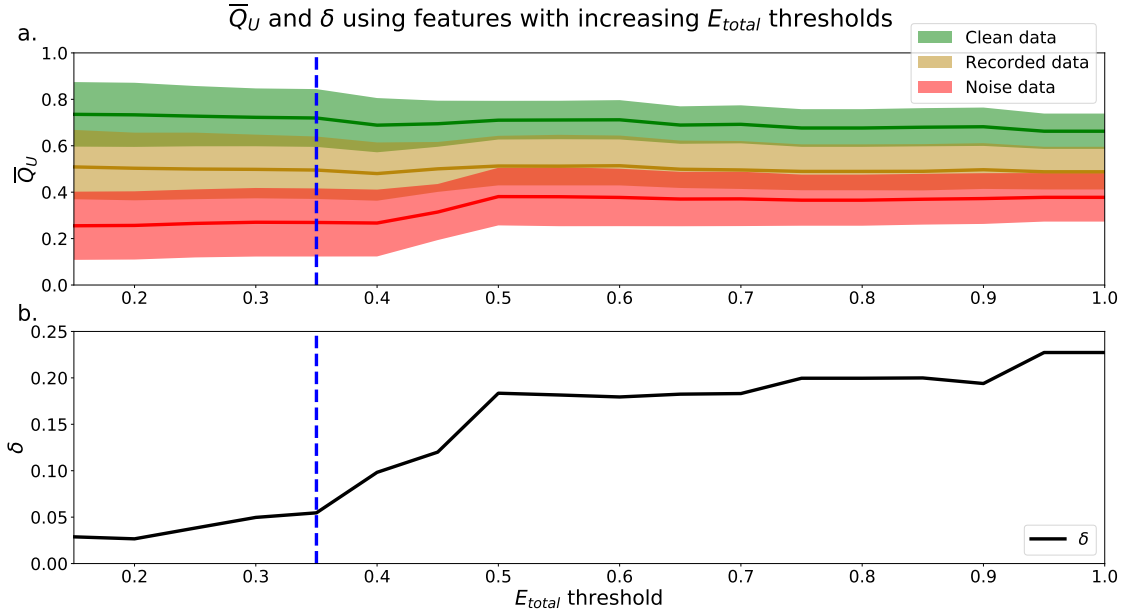


Figure 5.2 (a) $\overline{Q}_U$ means (lines) and standard deviations (shaded area) using features with increasing E_{total} thresholds. Green represents $\overline{Q}_U(\mathcal{C}_{Eye,model})$, yellow $\overline{Q}_U(\mathcal{R}_{Eye,model})$, and red $\overline{Q}_U(\mathcal{N}_{Eye,model})$. (b) δ values using $\overline{Q}_U$ obtained from increasing E_{total} thresholds. (a)-(b) Blue dashed-line marks a threshold of $E_{total} = 0.35$, where δ begins to increase more rapidly.

We see from Figure 5.2a that using scores from features with lower $E_{total,f}$ yielded better performing mean scores across data, $\overline{Q}_U$. Scores of $\mathcal{N}_{Eye,model}$ were lower and closer to 0.25, $\mathcal{R}_{Eye,model}$ were closer to 0.50, and $\mathcal{C}_{Eye,model}$ were higher and closer to 0.75. Further, from Figure 5.2b, the lower the E_{total} threshold applied, the lower the corresponding δ value. The best performing and most informative $\overline{Q}_U$ values were obtained with an inclusion criteria of either a threshold of $E_{total} \leq 0.20$ or $E_{total} \leq 0.35$. Table 5.1 shows our probabilities of errors (Equations 5.3-5.5) and $E_{total,f}$ (Equation 5.6) for each of the 30 features, where features with $E_{total,f} \leq 0.35$ are shaded and in bold. Further, to show how these features were distributed and data types appropriately mixed and separated, we show in Figure 5.3 the estimated distributions of $\mathcal{C}_{Eye,model}$, $\mathcal{R}_{Eye,model}$, and $\mathcal{N}_{Eye,model}$ for features with $E_{total,f} \leq 0.35$.

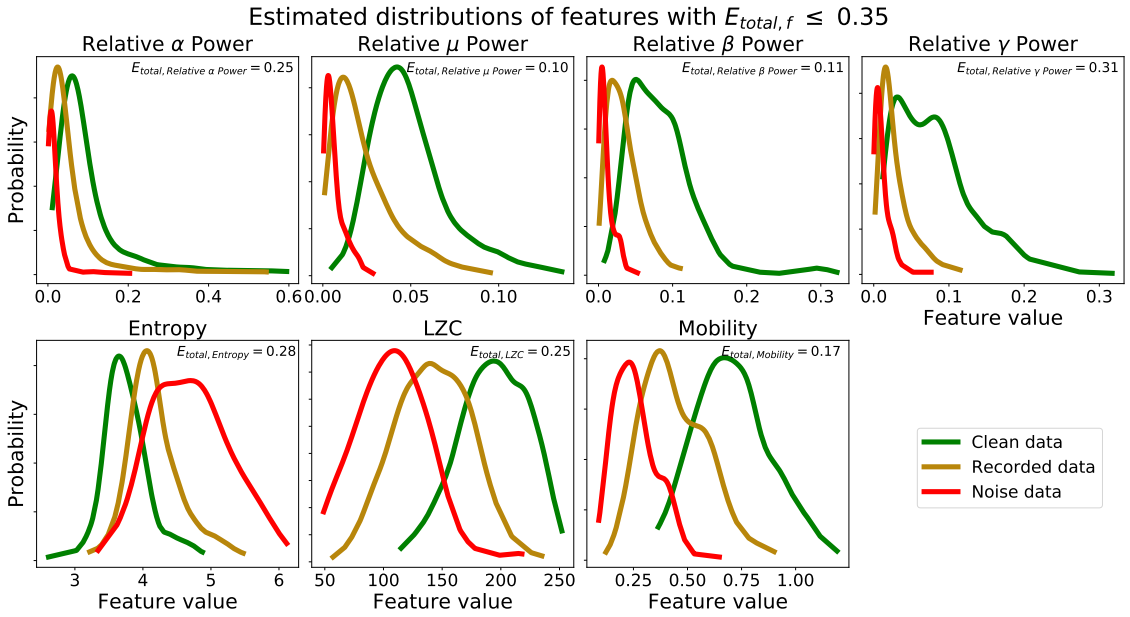


Figure 5.3 Estimated distributions clean, recorded, and noise data for features with $E_{total,f} \leq 0.35$. $E_{total,f}$ for set of estimated distributions of each feature are also shown.

	qEEG features	$P(error\|\mathcal{C}, f)$	$P(error\|\mathcal{R}, f)$	$P(error\|\mathcal{N}, f)$	$E_{total,f}$
Spectral	Relative δ Power	0.12	0.18	0.73	1.17
	Relative θ Power	0.14	0.50	0.33	0.47
	Relative α Power	**0.11**	**0.55**	**0.09**	**0.25**
	Relative μ Power	**0.04**	**0.54**	**0.02**	**0.10**
	Relative β Power	**0.06**	**0.54**	**0.02**	**0.11**
	Relative γ Power	**0.15**	**0.60**	**0.05**	**0.31**
	Absolute δ Power	0.02	0.17	0.36	0.71
	Absolute θ Power	0.06	0.13	0.43	0.86
	Absolute α Power	0.23	0.23	0.64	1.14
	Absolute μ Power	0.23	0.25	0.72	1.20
	Absolute β Power	0.62	0.68	0.24	1.04
	Absolute γ Power	0.84	0.87	0.10	1.31
	Spectral Entropy	0.17	0.63	0.06	0.36
I.T.	**Entropy**	**0.14**	**0.51**	**0.13**	**0.28**
	LZC	**0.10**	**0.59**	**0.06**	**0.25**
Statistical	Minimum	0.07	0.37	0.32	0.52
	Maximum	0.10	0.47	0.31	0.44
	Median	0.00	0.86	0.12	0.48
	Mean	0.00	0.87	0.09	0.46
	Variance	0.07	0.26	0.26	0.57
	SD	0.11	0.41	0.17	0.37
	Skew	0.15	0.42	0.39	0.63
	Kurtosis	0.11	0.21	0.51	0.91
Signal Shape	Curve Length	0.29	0.27	0.61	1.13
	Energy	0.00	0.31	0.47	0.66
	Non-linear Energy	0.19	0.18	0.65	1.15
	Sixth Power	0.00	0.27	0.59	0.83
	Sum	0.00	0.87	0.09	0.46
	Mobility	**0.07**	**0.56**	**0.04**	**0.17**
	Complexity	0.27	0.47	0.40	0.70

Note: I.T. - Information theoretic, LZC - Lempel-Ziv complexity, SD - Standard deviation

Table 5.1 Probabilities of error ($P(error)$) and total error for estimated densities of each feature ($E_{total,f}$) from $\mathcal{C}_{Eye,model}$, $\mathcal{R}_{Eye,model}$, and $\mathcal{N}_{Eye,model}$ used in unimodal method. Features with $E_{total,f} \leq 0.35$ shaded and in bold.

We see from Table 5.1 that the three features with the lowest $E_{total,f}$ were relative μ power, relative β power, and mobility, followed by relative α power, relative γ power, entropy, and LZC. Visually, we see that features with $E_{total,f} \leq 0.35$ show the following similar traits (Figure 5.3): (1) the clean data and noise data distributions had little overlap, (2) the mean value of the recorded data distributions were close to the critical value (Equation 5.1), and (3) all data distributions were generally smooth and had a single mode.

We verify numerically in Table 5.2 the appropriate E_{total} inclusion criteria threshold by using sets of features with increasing E_{total} thresholds.

Data	$\overline{Q}_U$ ($E_{total,f} \leq 0.20$)	$\mathbf{\overline{Q}_U}$ ($\mathbf{E_{total,f} \leq 0.35}$)	$\overline{Q}_U$ ($E_{total,f} \leq 0.50$)
$\mathcal{C}_{Eye,model}$	0.73 ± 0.14	$\mathbf{0.72 \pm 0.13}$	0.71 ± 0.08
$\mathcal{R}_{Eye,model}$	0.50 ± 0.15	$\mathbf{0.49 \pm 0.14}$	0.51 ± 0.13
$\mathcal{N}_{Eye,model}$	0.26 ± 0.15	$\mathbf{0.27 \pm 0.15}$	0.38 ± 0.12
δ	0.03	**0.06**	0.18

Table 5.2 Mean scores $\overline{Q}_U \pm$ standard deviation and δ values of unimodal method across model generation cross-validation using features with incremental E_{total} thresholds. Values with chosen threshold, $E_{total,f} \leq 0.35$, in bold.

We find that increasing the E_{total} threshold from 0.20 to 0.35 and including features with $0.20 < E_{total,f} \leq 0.35$ did not have much of an impact in performance since δ only increased by 0.03. However, further including features with $0.35 < E_{total,f} \leq 0.50$ did seem to affect score more negatively since δ further increased by 0.12. Thus, to include more features and capture more characteristics of the signals we set the inclusion criteria of features' Q_f to use when computing Q_U (Equation 5.2) to features with $E_{total,f} \leq 0.35$. We can visualize estimated distributions of

Q_U and see in Figure 5.4 that distributions of $\mathcal{C}_{Eye,model}$ and $\mathcal{N}_{Eye,model}$ scores were well separated with wider distributions and intersected at $Q_U = 0.53$, while the estimated distribution of Q_U of $\mathcal{N}_{Eye,model}$ had a more narrow distribution with relatively symmetric decreasing tails centered at 0.47.

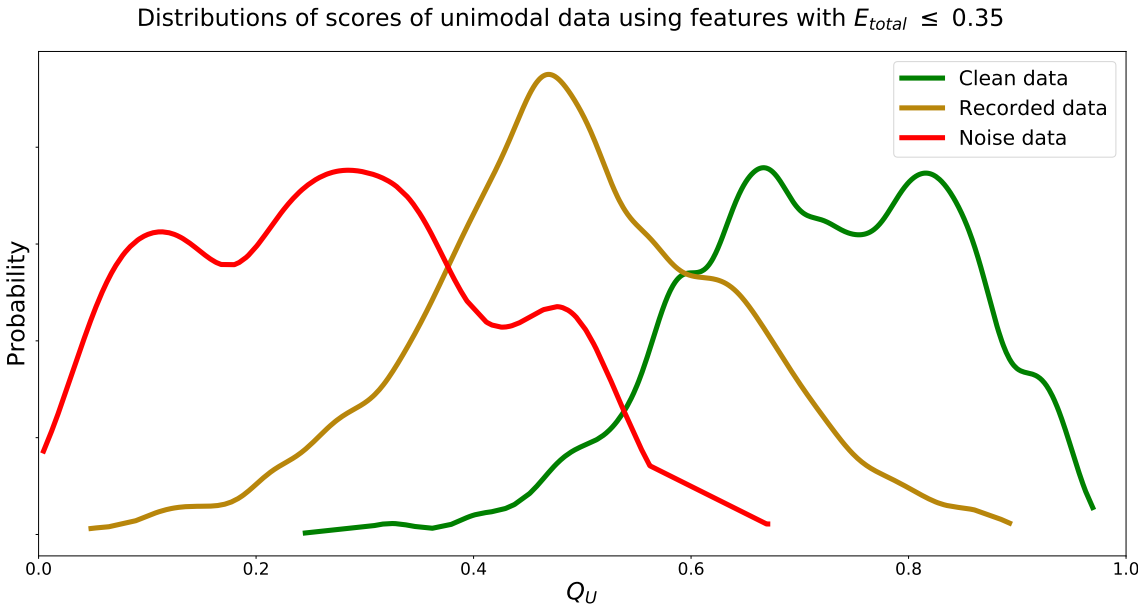


Figure 5.4 Estimated distributions of Q_U for $\mathcal{C}_{Eye,model}$, $\mathcal{R}_{Eye,model}$, and $\mathcal{N}_{Eye,model}$ using features with $E_{total,f} \leq 0.35$. Distributions of Q_U of $\mathcal{C}_{Eye,model}$ and $\mathcal{N}_{Eye,model}$ intersect at $Q_U = 0.53$. Distribution of Q_U of $\mathcal{R}_{Eye,model}$ centered at 0.47.

5.2.2 Scoring data with multimodal sources

In cases when the noise source cannot be captured directly from the same recording modality and require multiple recording modalities (referred to as 'multimodal method'), it may not be possible to directly compare distributions of quantitative features of both the signal of interest and noise. Therefore, we must compare clean signals and recorded signals with noise present to formulate models to identify differences.

5.2.2.1 Scoring method for data with multimodal sources

This problem can be formulated as a two-class classification machine learning problem. In one class we have clean data ($\mathcal{C}$) and in the second class we have recorded data with noise present ($\mathcal{R}$). For classification these two data are assigned numeric labels, $\{\mathcal{C} : 1,\ \mathcal{R} : 0\}$. The deep learning classifier can then find the difference between the two data sets which here is the presence of the noise. Once trained, new data can be classified with a probabilistic prediction using Softmax functions in the last layer. The closer the probability is to zero, the more similar the signal is to the noise source, while probabilities closer to one would represent predicted signals without noise.

The deep learning model (Section 1.7.2) would traditionally select the class with the highest probabilities as the prediction. To score the data from the deep learning models, we used the prediction probabilities directly (Equation 5.8). Here, $P_{\mathcal{C}}$ is the predicted probability of input data being part of the clean data class and $P_{\mathcal{R}}$ is the predicted probability of input data being part of the recorded data class with noise present. Then the scoring function for our multimodal method is defined as

$$Q_M = \begin{cases} \frac{1}{2} + \frac{P_{\mathcal{C}} - P_{\mathcal{R}}}{2}, & \text{if } P_{\mathcal{C}} \geq P_{\mathcal{R}} \\ \\ \frac{1}{2} - \frac{P_{\mathcal{R}} - P_{\mathcal{C}}}{2}, & \text{if } P_{\mathcal{C}} < P_{\mathcal{R}} \end{cases} \tag{5.8}$$

The motivation of this definition is such that if $P_{\mathcal{C}} = 1,\ P_{\mathcal{R}} = 0$ then $Q_M = 1$,

if $P_{\mathcal{C}} = 0$, $P_{\mathcal{R}} = 1$ then $Q_M = 0$, and if $P_{\mathcal{C}} = 0.5$, $P_{\mathcal{R}} = 0.5$ then $Q_M = 0.5$. Further, when differences between the probabilities are larger, Q_M should be made higher when $P_{\mathcal{C}} >> P_{\mathcal{R}}$ and lower when $P_{\mathcal{C}} << P_{\mathcal{R}}$. This is desired since larger differences between $P_{\mathcal{C}}$ and $P_{\mathcal{R}}$ would imply that the predictions of the model are more confident and therefore scores should be adjusted accordingly.

5.2.2.2 Data with multimodal sources

To generate trained models and subsequently validate the multimodal approach, we focused on EEG with noise from motion since it cannot be recorded directly from EEGs and are only present in EEG as embedded noise.

We again used the data set obtained from the University of Houston that contained EEG recordings (sampled at 100 Hz) as well as motion capture from eleven subjects walking on a treadmill for six minutes (Section 5.2.1.2) [72]. EEG was recorded from the same electrodes and configuration referred to above with data from all eleven subjects available, where channel FT9 was used as ground, FT10 for reference, TP9, TP10, PO9, and PO10 were used for EOG, and the 58 remaining channels for EEG. Further, the experimental protocol had subjects walk on a treadmill at one, two, three, and four miles-per-hour.

Another noise removal algorithm was used to remove motion artifacts [72]. This algorithm also used an H^{∞} filter formulation with Voltera series and time-varying weight assumption. Unlike EOG data which was directly measured from the same modality, the reference signal used to identify the motion artifacts in

EEG signals were 3-axis acceleration values, after gravity compensation, using the quaternion of IMUs.

We further supplemented this data with recorded EEG data from another study. EEG was recorded from 20 subjects while walking around an art exhibit [75]. These EEG were recorded with a 32 electrode array labeled in accordance with the extended 10-20 system where, P09 and P019 for EOG, and Fp1, Fp2, F7, F8, T7, T8, TP9, TP10, P7, and P8 were generally unused and contained mostly zero-values, and the remaining 20 channels had consistently available EEG. Each trial began with a baseline wall stair of approximately one-minute. Afterwards, subjects walked around an art exhibit for at least seven minutes.

To obtain a robust model that scored EEG quality based on the presence of motion artifacts we combined data from both these sources to obtain a $\mathcal{C}_{Motion}$ that represented EEG data from recordings where motion was removed through an artifact removal algorithm and recordings where motion was known to not be present. Similarly, we combined data from both sources to obtain a $\mathcal{R}_{Motion}$ that represented EEG data from recordings where motion was present under different circumstances, in both controlled environments with different walking speeds, and in an uncontrolled setting where subjects walked through an art exhibit. Since data from our second source ([75]) only had 20 EEG channels available, we used the same 20 channels from our first data source ([72]). Each recording from both sources were segmented into 30-second epochs for our multimodal method. Combining these we obtained $\mathcal{C}_{Motion}$ with $n = 568$ (12 30-second segments from 6-minutes $\times$ 4 walking speeds $\times$ 11 subjects + 2 30-second segments from 1-minute of baseline $\times$ 20 subjects

= 568), and $\mathcal{R}_{Motion}$ with n = 808 (12 30-second segments from 6-minutes × 4 walking speeds × 11 subjects + 14 30-second segments from 7-minutes of walking × 20 subjects = 808), where each sample was of size 3000 × 20 (30-second segments sampled at 100Hz across 20 channels), for our method scoring data with multimodal source.

5.2.2.3 Model generation for data with multimodal sources: EEG with motion artifacts

To obtain scoring models for this data we separated 90% of the data randomly to generate the models and reserved 10% of the data to test and validate the results. We present ten-fold cross-validated values from ten generated models from different balanced samplings of 90% of the data, denoted with the subscript *'model'*, that resulted in $\mathcal{C}_{Motion,model}$ and $\mathcal{R}_{Motion,model}$, each with $n = 512$. We report the mean results as well as the mean standard deviation across samples for the ten-fold cross-validation. For this application we used deep learning models used in previous research that used multi-channel EEGs as input and output class predictions [123].

We show in Table 5.3 predicted probabilities and associated scores (Equation 5.8) using model training data from our deep learning-based scoring for our multimodal scoring method.

Data	Probability of $\mathcal{C}_{Motion}$	Probability of $\mathcal{R}_{Motion}$	$\overline{Q}_M$
$\mathcal{C}_{Motion,model}$	0.79 ± 0.13	0.18 ± 0.12	0.81 ± 0.13
$\mathcal{R}_{Motion,model}$	0.28 ± 0.26	0.67 ± 0.26	0.30 ± 0.26

Table 5.3 Mean predicted probabilities ± standard deviation and corresponding mean scores, $\overline{Q}_M$, of multimodal method across model generation cross-validation.

We see that the recorded data was in fact scored well below 0.5 while data without walking artifacts were scored well above 0.5, instilling confidence in the trained deep learning models. We visualize estimated distributions of Q_M and see in Figure 5.5 that the distribution of $\mathcal{C}_{Motion,model}$ scores was negatively skewed, the distribution $\mathcal{R}_{Motion,model}$ scores was positively skewed, and both distributions intersected at $Q_M = 0.62$.

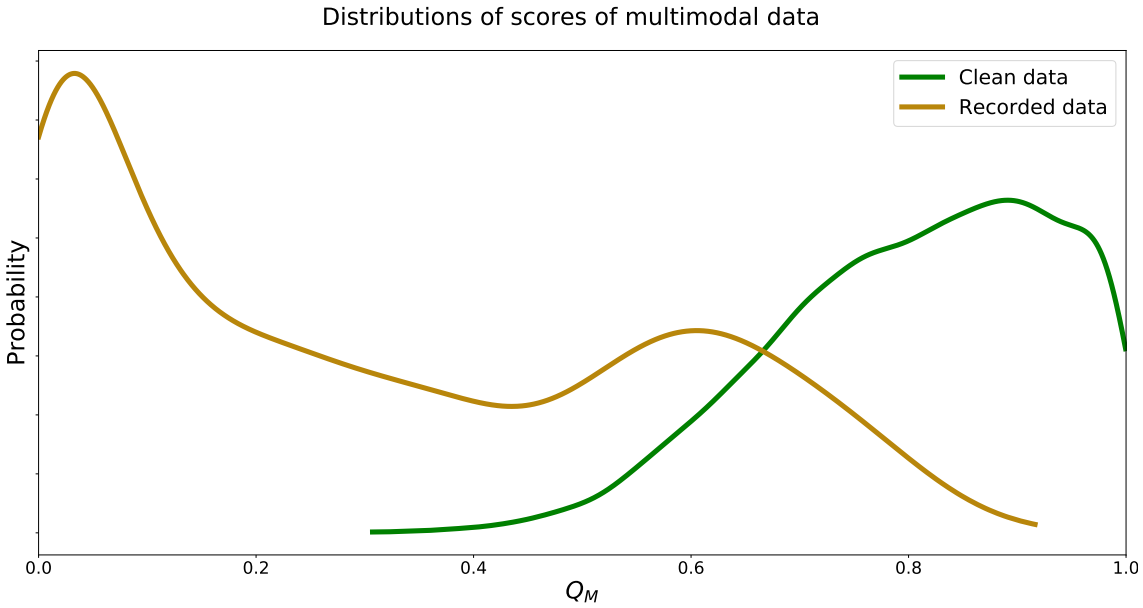


Figure 5.5 Estimated distributions of Q_M for $\mathcal{C}_{Motion,model}$ and $\mathcal{R}_{Motion,model}$. Distributions of Q_M of $\mathcal{C}_{Motion,model}$ and $\mathcal{R}_{Motion,model}$ intersect at $Q_M = 0.62$.

5.2.3 Evaluating artifact removal algorithms: Removing ocular artifacts from EEG

Using scoring methods introduced in this study, we can evaluate the efficacy of artifact removal algorithms (Section 1.4) that target a specific type of noise. The most common type of noise that artifact removal algorithms target in EEG

recordings is from ocular activity. Therefore, we use our unimodal method (Section 5.2.1) to evaluate these types of algorithms. A MATLAB software toolbox that is commonly used in EEG processing is EEGLAB. We used two artifact removal algorithms with different approaches that aim to remove ocular artifacts. Though different, both methods rely on independent component analysis (ICA) which is the most common method used for removing ocular artifacts [68].

Using EEGLAB (version 14.1.1) and available plug-ins, we applied the independent component artifact classification multiple artifact rejection algorithm (MARA) to $\mathcal{R}_{Eye}$ [143], [62]. This method uses trained classifiers to identify components from ICA that are artifactual for rejection. The FastICA method was used for obtaining ICA components and components identified by MARA were automatically rejected [57].

We also applied automatic artifact removal (AAR) (version 1.3) targeting EOG removal using blind source separation (BSS) to $\mathcal{R}_{Eye}$ [69], [70]. AAR with BSS was applied with all defaults, using the SOBI algorithm in MATLAB which has been shown to be effective for BSS [9], [118]. To automatically apply AAR with BSS to EOG, the method identifies ICA components that represent EOG noise by removing components with the lowest mean fractal dimension values [45].

To evaluate and compare the effectiveness of these artifact removal algorithms we obtained the average score of $\overline{Q}_U(\mathcal{R}_{Eye})$ and compared the average scores of the data after being processed with MARA, $\overline{Q}_U(\mathcal{C}_{Eye-MARA})$, and ARR, $\overline{Q}_U(\mathcal{C}_{Eye-AAR})$.

5.3 Results

We show mean results as well as the mean standard deviation across samples for our unimodal and multimodal scoring methods for the data sets specified. For both methods, we report the ten-fold cross-validated values from ten generated models with the remaining 10% unseen data, denoted with the subscript *'valid'*. For the unimodal method, we scored $\mathcal{C}_{Eye,valid}$ and $\mathcal{R}_{Eye,valid}$, both with $n = 278$, and $\mathcal{N}_{Eye,valid}$, with $n = 19$. For the multimodal method, we scored $\mathcal{C}_{Motion,valid}$ and $\mathcal{R}_{Motion,valid}$, each with $n = 56$. These validation results are presented below in Section 5.3.1 (unimodal method) and Section 5.3.2 (multimodal method). Section 5.3.3 shows results using all data and validation data.

5.3.1 Scoring data with unimodal source: EEG with ocular artifacts

Applying the unimodal method to data not used for model development we scored the data (Equation 5.2) and present cross-validation results using features that met the inclusion criteria determined for features, $E_{total,f} \leq 0.35$ (Table 5.4).

Data	$\overline{Q}_U$ ($E_{total,f} \leq 0.35$)
$\mathcal{C}_{Eye,valid}$	0.73 ± 0.11
$\mathcal{R}_{Eye,valid}$	0.50 ± 0.14
$\mathcal{N}_{Eye,valid}$	0.23 ± 0.14
δ	0.04

Table 5.4 Mean scores $\overline{Q_U} \pm$ standard deviation and δ values of unimodal method results across cross-validation using features $E_{total,f} \leq 0.35$.

We see that the models developed performed well, obtaining $\delta = 0.04$ from

ideal mean score characteristics. The $\overline{Q}_U$ and δ found using the unseen validation data were similar to those from data used generating these models. Further, estimated distributions of Q_U of validation data followed closely those shown in Figure 5.4 with distributions of Q_U of $\mathcal{C}_{Eye,valid}$ and $\mathcal{N}_{Eye,valid}$ intersecting at 0.52.

5.3.2 Scoring data with multimodal sources: EEG with motion artifacts

We next show results of our multimodal data scoring method. We show the cross-validated predicted probabilities for each data type as well as the quality score (Equation 5.8) using our deep learning-based scoring method on unseen data (Table 5.5).

Data	Probability of $\mathcal{C}_{Motion}$	Probability of $\mathcal{R}_{Motion}$	$\overline{Q}_M$
$\mathcal{C}_{Motion,valid}$	0.77 ± 0.13	0.19 ± 0.12	0.79 ± 0.13
$\mathcal{R}_{Motion,valid}$	0.28 ± 0.27	0.67 ± 0.28	0.31 ± 0.28

Table 5.5 Mean predicted probabilities $\pm$ standard deviation and corresponding mean scores, $\overline{Q}_M$, of multimodal method results across cross-validation.

We see that the predicted probabilities and $\overline{Q}_M$ from unseen data were similar to those from the model generation data. The model was able to score recorded data with noise well below 0.5 while also scoring clean data well above 0.5. Estimated distributions of Q_M of unseen validation data followed closely those shown in Figure 5.5 with distributions of Q_M of $\mathcal{C}_{Eye,valid}$ and $\mathcal{R}_{Eye,valid}$ intersecting at 0.57. We note that mean cross-validated standard deviations of $\overline{Q}_M(\mathcal{R}_{Motion})$ data were twice as large as $\overline{Q}_M(\mathcal{C}_{Motion})$ data, which we discuss in Section 5.4.

Since the data used was a combination of data of both baseline recordings without walking and data where walking artifacts were removed from data, we wished to determine what the deep learning models identified as important in scoring EEG quality from motion artifacts. We applied easyPEASI, a method that identifies frequency bands that were important to the deep learning models (Chapter 4).

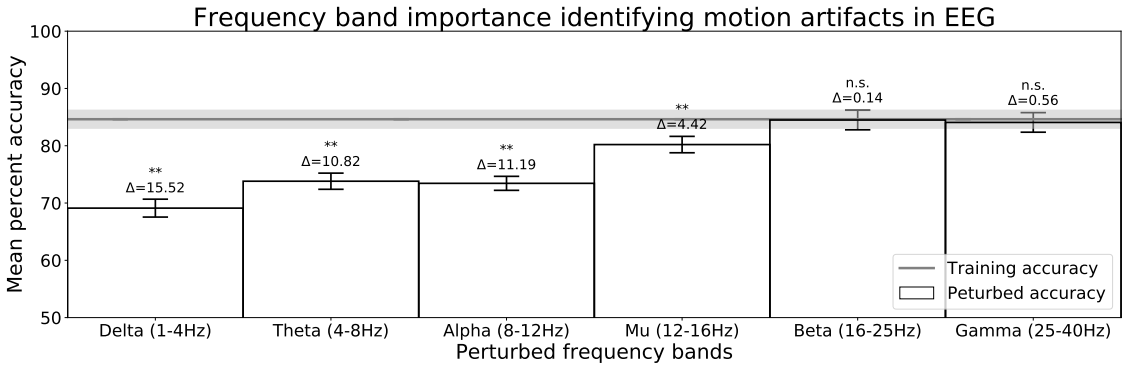


Figure 5.6 easyPEASI results for models comparing the differences between EEG with walking artifacts against EEG without walking and cleaned data. Each bar represents the classification accuracy from the DCNN model when a particular frequency band was perturbed using the easyPEASI. Lines and shaded area at $84.63\% \pm 01.68$ are the mean and standard deviation of training accuracies with no perturbed data. Δ values are the difference between the mean training accuracy and the mean perturbed accuracy. P-value significance thresholds set at: n.s.: not significant, *: $P < .01$, **: $P < .001$.

From Figure 5.6 we see that the lower frequencies were found to be important in scoring EEG quality which is consistent with known artifact properties [68].

5.3.3 Evaluating algorithms removing ocular artifacts from EEG

As an extension and further application of our unimodal scoring method we score $\mathcal{R}_{Eye}$ data with artifacts removed by two different methods, MARA and AAR (Section 5.2.3). To compare their effectiveness we score all $\mathcal{R}_{Eye}$ data (Section

5.2.1.2), with $n = 2784$, and validation $\mathcal{R}_{Eye,valid}$ data (Section 5.3.1), with $n = 278$ (Table 5.6).

Data	$\overline{Q}_U$
$\mathcal{C}_{Eye-MARA}$	0.72 ± 0.12
$\mathcal{C}_{Eye-AAR}$	0.60 ± 0.11
$\mathcal{R}_{Eye}$	0.50 ± 0.14
$\mathcal{C}_{Eye,valid-MARA}$	0.72 ± 0.10
$\mathcal{C}_{Eye,valid-AAR}$	0.61 ± 0.10
$\mathcal{R}_{Eye,valid}$	0.50 ± 0.14

Table 5.6 Comparison of mean scores $\overline{Q}_U \pm$ standard deviation of unimodal method across cross-validation of all data and validation data processed by MARA and AAR methods.

We see that for both sets of data, $\overline{Q}_U(\mathcal{C}_{Eye-MARA}) > \overline{Q}_U(\mathcal{C}_{Eye-AAR})$. That is, data processed with MARA, $\mathcal{C}_{Eye-MARA}$, was scored higher than data processed with AAR, $\mathcal{C}_{Eye-AAR}$. This means that MARA may have been more effective than AAR at removing ocular artifacts from EEG. Further, as hypothesized, data processed by both artifact removal algorithms resulted in data with mean scores higher than $\overline{Q}_U(\mathcal{R}_{Eye})$.

5.4 Discussion

We present in this work two novel methods to score electrophysiological signal quality. These methods were separated by the differences between the source of the signal noise and thus how it could be characterized. Specifically, we applied our method to quantify signal quality when the noise source could be recorded from the same modality to EEG with ocular artifacts. In such cases, the same quantitative features can generally be computed on both the signal of interest (EEG) and noise

source (EOG), and compared directly to each other.

For other sources of artifacts, such as motion, supplemental physiological measurement tools would be needed to measure the motion directly. In these cases, it is not usually possible to compare quantitative features computed on both the signal of interest and noise source directly. For example, in our application scoring EEG with motion artifacts, the value of entropy of an EEG channel may not be directly comparable to the entropy of acceleration in the X-axis from an IMU. Therefore, we apply a method which only requires signals that are free of noise and recordings with noise present.

With both methods, we generate models to score signal quality of EEG with either ocular or motion artifacts. Though in research settings high density EEG headsets, which usually have electrodes to measure EOG, are commonly used, low density EEGs are becoming more common and have been shown to be effective for many applications [71]. Even when high density EEG sets are used, other artifact sources, like motion, are not generally directly captured. These data-driven models, once generated, can be used to evaluate the quality of EEG and potential presence of noise from artifacts of other data without the need to directly record noise sources.

For our unimodal method, in addition to the scoring models, we also formulate criteria to quantify features that are effective for scoring EEG. Results show that the E_{total} measure and δ performance metric do meaningfully represent a feature's ability to score EEG in a unimodal data source setting. We determine that features should be used only if they have a $E_{total,f} \leq 0.35$. From analyses shown in Figure 5.2 and Table 5.2, we see that features with $E_{total,f} \leq 0.35$ performed better than

those with higher E_{total}. Incorporating features with $0.35 < E_{total,f} \leq 0.50$ along with features with lower $E_{total,f}$ decreased performance.

In general, our unimodal method was able to generalize to data not used in generating the scoring models. Results in Table 5.4 and score distributions matched values found from generating these models in Table 5.2 and shown in Figure 5.4. For our multimodal method, we were able to score data from two different data sets well. Combined data that had artifacts removed and baseline data when subjects were known to be still were both scored as generally clean data. Both in the model generation and validation of this method, we found that the cross-validated standard deviations of recorded data scores, $\overline{Q}_M(\mathcal{R}_{Motion})$, were twice as large as standard deviations of clean data scores, $\overline{Q}_M(\mathcal{C}_{Motion})$ (Tables 5.3 and 5.5). This may have been because the data sources used had many different types of walking. Some subjects walked at various speeds, varying from one to four miles-per-hour, while others walked through an art exhibit at their own pace, even potentially with stops to view art pieces which may explain the second mode with higher scores seen in Figure 5.5 [72], [75].

Examining the application of our unimodal method to evaluate the effectiveness of artifact removal algorithms, we see that MARA was more effective than AAR in removing artifacts, obtaining higher scores for the processed recorded data. Strictly speaking, this means that MARA removed ocular artifacts more similarly to the H^∞ method used to develop our models than AAR did. Further, we see that these results were consistent both across all data, including those used to generate our models, and the independent validation data (Table 5.6). These results present

a significant advancement to evaluating artifact removal algorithms by providing quantitative measures on real EEG recordings as opposed to qualitative evaluations or using synthetic data [63].

Both methods were applied to EEG to score different recordings with different noise artifacts. However, future work may apply multiple models to the same recordings to obtain several quality scores that may be combined to assess an overall data quality score. We also note that though the unimodal method can only be applied when the noise source is available from the same modality, the multimodal method can be applied to data with unimodal source. Our multimodal approach, which is deep learning-based, requires a large amount of data to generate accurate models. If more data were available, we may attempt to apply our multimodal method to score data from a unimodal source, in this case using only $\mathcal{C}_{Eye}$ and $\mathcal{R}_{Eye}$ to score these data with Q_M.

Comparing these approaches, we see that our multimodal method required only an appropriate deep learning model designed to classify the input signal. These deep learning models identify signal features important to distinguish the two input classes, clean signals and signals with noise, automatically. Our unimodal method on the other hand, required more manual selection of both the features of the signal to compute and feature importance for scoring signal quality. However, as opposed to our multimodal method, our unimodal method allows for us to identify which specific quantitative features of the signal of interest were important for scoring signal quality with respect to the targeted noise.

Both our unimodal and multimodal methods present advancements and im-

provements from existing methods by evaluating EEG signal quality with continuous scores. Previous methods have generally evaluated signal quality by classifying EEG into discrete quality categories [30], [56], [48]. Our scoring models allow for rapid evaluation of signal quality of EEG data. Future work may expand the type of data used to generate these models, such as including signals known to have no ocular artifacts present. To further validate scores generated by these models, data could be obtained or generated with known levels of noise. Signals with more noise introduced should result in lower scores. Our analyses presented signals of each type in aggregate, averaging across samples, future analyses may investigate recordings more specifically to identify the level of noise in signals.

Chapter 6: Conclusions and future work

6.1 Conclusions

We began by providing a brief background on the advanced methods used throughout the research chapters presented. Altogether, the research chapters presented here all advance quantitative methods to analyze and interpret neural activity from electroencephalography (EEG).

Our contributions begin by developing methods to evaluate the utility of commonly used quantitative EEG features for different applications. We proposed a non-parametric method based on counts of non-significant statistical tests using a large data set. We evaluated feature values' dependencies on epoch length of EEG recordings used as well as their relationship between intra- and inter-subject consistency. This work provides data-driven methods of identifying EEG features and their spatial characteristics ideal for various EEG applications, and determining future EEG feature consistencies using an existing EEG data set. These types of features could also be useful in establishing baseline features for a standard EEG signal and subsequently detect variations away from those baselines.

With features of EEG computed and associated clinical reports available from an unprecedentedly large data base we sought to apply machine learning methods

to determine if leveraging these data could identify novel classifications. We subsequently applied textual analysis on clinical reports associated with EEG recordings to obtain patient characteristics for the quantitative EEG data. Given the clinical nature of the data set and novelty we focused on classifying subject's medications based on their EEG. Kernel support vector machine (kSVM) models have a well-established history of good performance as a feature-based classifier in EEG applications. However, recently, deep convolutional neural network (DCNN) models have shown state-of-the-art results and superb effectiveness in EEG applications. We identified the two most common medications taken by subjects in the data set, two anticonvulsants: Dilantin (generic phenytoin) and Keppra (generic levetiracetam). As novel EEG-based classifications, we compared kSVM and DCNN models' effectiveness to determine medications taken from solely from EEG. Results showed a first step in determining the utility of machine learning in drug classification based on neurological activity. By investigating the abilities of learning algorithms to differentiate individuals taking different medications through EEG, this work is a step on the road to the broader utility of machine learning in biomarker development.

Though the ability to accurately classify subject's medications present a potential tool for pharamacodynamics, as a clinically relevant application it would also be informative to understand the underlying neurological differences between classified groups. With feature-based approaches such as kSVM, there are several methods to identify feature importance in classifications among features used. On the other hand, deep learning models identify important features automatically as part of the end-to-end algorithm. However, understanding what was learned by

these models can be difficult, particularly for EEG-driven DCNNs. To address this, we developed easyPEASI as a simple method to identify important frequency bands manipulating input data only and without the need to investigate the model itself. We validated easyPEASI and applied it to EEG-based medication-use classifications to identify relevant frequency bands. Beyond identifying frequency bands that are important in medication classifications, easyPEASI shows promise as method of obtaining explainable artificial intelligence and interpretable models from EEG-driven deep learning through a simpler more accessible method perturbing only input data.

Until this point, our work used a large data set with well segmented recordings from clinical settings to identify important characteristics of EEG. However, EEG is usually corrupted by noise which can degrade the effectiveness of diagnostic tools and brain-machine interfaces. Analyzing artifacts present in EEG recordings is an active area of research and often the most difficult processing step in analysis and interpretation. We therefore used data from experiments where artifactual data and artifact-free data were available. From these data we validated scoring models formulated to identify signal quality for data with ocular and motion noise sources with a continuous score ranging from zero to one. Our scoring models allow for rapid evaluation of signal quality of EEG data. Models are easily extendable to other physiological measurements where noise is often present. This work also provides a novel data-driven method to evaluate and compare quantitatively the effectiveness of noise removal algorithms utilizing real data.

This dissertation successfully achieves our aims to *(1)* develop population consistency metrics and apply them to commonly analyzed quantitative EEG features,

(2) apply and compare feature-based and deep learning-based classifications to identify, solely through EEG, medication profiles of subjects taking anticonvulsants, *(3)* propose and validate a new method to identify frequency band of importance in EEG-driven deep learning classifications, and *(4)* develop methods to score signal quality of electrophysiological signals, specifically EEG with noise from ocular and motion artifacts.

6.2 Future work

There are several potential ways results and methods from our works might be followed up with or applied in future research.

Our consistency metrics may be applied to new quantitative EEG features that are developed to better understand their utility. Furthermore, since differences between intra-subject and inter-subject consistencies can quantify a qEEG feature's discriminability between subjects, they may also be used to inform feature selection in future machine learning classification applications.

As such, future work may investigate which specific features were chosen in feature-based medication classifications and their clinical implications. Though we investigated deep learning classifications through easyPEASI, future development of extensions to easyPEASI could be done to identify brain regions that are important in classification tasks. Regarding the applications investigated in this work, investigating the differences between subjects with abnormal EEG taking anticonvulsants against subjects with normal EEG taking no medications may potentially determine

anticonvulsants' efficacy.

Finally, with respect to our data quality scoring methods, future work may expand the type of data used to generate these models, such as including signals known to have no ocular artifacts present. Our analyses presented signals of each type in aggregate, averaging across samples, future analyses may also investigate recordings more specifically to identify the level of noise in signals.

We hope results and methods presented in this dissertation are useful in future research beyond the potential works described here and help further advance the field of computational neuroscience.

(This page intentionally left blank)

Appendix A: EEG Spectral Connectivity Analysis in a Large Clinical Population

Overview

This appendix presents work that was published as a conference proceedings as part of the 9th IEEE/EMBS Neural Engineering (NER) conference in 2019 [93].

We applied coherence-based connectivity analyses to EEG data of clinically determined normal and abnormal recordings from the Temple University Health University EEG Corpus to identify differences in connectivity between populations.

Motivation: Recent studies, usually with less than 50 subjects, have shown the utility of brain connectivity in the diagnosis and understanding of various neurological conditions. A better understanding of the baseline brain connectivity in resting state on a larger group of subjects could improve the efficacy of medical devices that use neurological biomarkers for treatment or diagnosis of a medical condition.

Approach: We obtained coherence-based connectivity networks by averaging coherence measures for all subjects in populations to construct default mode networks. We then computed network connectivity parameters to quantify fea-

tures of these networks from each subject. We also computed Fiedler vectors using coherence based networks, which represented groups of nodes in the connectivity adjacency matrix identified through spectral graph methods.

Application: We applied these methods to identify differences between subjects with normal or abnormal EEG recordings. We established default mode networks for the different populations on several frequency bands. We found that frequency bands differed across the populations more than specific graph properties. Further, we found that there was an increased level of connectivity in the population with abnormal EEG recordings.

Research impact: This work represents a novel application of network theory to the largest available EEG data set. We show network theory methods applied to EEG are able to distinguish between normal and abnormal EEG recordings. These results may lead to neural connectivity-based diagnostic aides.

A.1 Introduction

Brain connectivity is the pattern of links between neurons, neuronal populations, or anatomically separated brain regions. Brain function and how information is shared throughout the brain are dependent on these connections. Recent studies have shown the utility of brain connectivity in the diagnosis and understanding of various neurological conditions such as epilepsy, schizophrenia, Alzheimer's, alexithymia, and dyslexia [31], [33], [35], [60], [46]. Connectivity in the brain has also been shown to change with age [41]. Electroencephalography (EEG), which measures brain electrical signals from the scalp, is a common non-invasive, relatively inexpensive, neuro-monitoring technique that can be used to assess brain connectivity. Unlike some EEG biomarkers, connectivity based analyses explicitly utilize spatial information present in EEG. A better understanding of the baseline brain connectivity in resting state, or default mode network (DMN), could improve the efficacy of medical devices that use neurological biomarkers for treatment or diagnosis of a medical condition [20].

The DMN has mainly been studied through fMRI or PET imaging [111], [18]. With EEG, previous studies have also investigated the DMN between eyes-open and eyes-closed [20]. However, as with most previous EEG work, the number of subjects used to identify connectivity differences between populations has generally been below 50. With the availability of big data resources it is now possible to expand these types of analyses to a larger group of subjects and examine neural connectivity under noisy, less controlled conditions. These type of data also provide a more het-

erogeneous population than typically analyzed. In this study, EEG recordings were used to study neural connectivity in patients both with and without neurological disorders. To analyze neural connectivity we use spectral coherence (SC) [16]. SC is a mathematical method that describes how similar neuronal oscillatory activity of two or more sensors, or brain regions, are with each other. With the demographic and clinical information about each patient, connectivity results were studied across age, sex, clinical impression, and medications taken. However, we report here only on the work and results related to differences in clinical impressions of the EEG. The aim of this study is to compare connectivity properties of a normative and abnormal population obtained from a large clinical data set.

A.2 Methods

To investigate connectivity differences in these populations we applied three evaluation methods to labeled graphs obtained from analyzing the Neural Engineering Data Consortium (NEDC) EEG Corpus from Temple University Health, version 1.0.0 (Figure A.1). This data set contained 23,257 EEG recording sessions (1.1TB) from 13,551 subjects. Each session was accompanied by a report containing clinical impressions and patient characteristics [51].

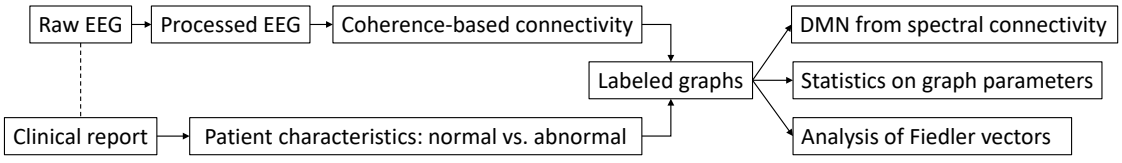


Figure A.1 Data processing and analysis flowchart.

A.2.1 Data set: Defining normal and abnormal

Patient information and clinical impressions of the EEG recordings were obtained through textual analysis of the clinical reports. Since the recordings in the NEDC database were not all acquired with the same EEG system, the channel layouts were not all consistent. A subset of channels common across most recording sessions was identified and used in subsequent analyses. From a standard 10-20 electrode montage, the subset of channels used were: Fp1, Fp2, F3, F4, C3, C4, P3, P4, O1, O2, F7, F8, T3, T4, T5, T6, Fz, Cz, Pz.

Five minute recordings were used for the connectivity analyses based on recording lengths used in previous DMN studies [31], [46], [41]. From each session that contained at least six minutes of EEG data, minutes 1-6 were used across the same 19 channels. The first minute of each recording was excluded to reduce artifacts present in the EEG [123].

The NEDC also released a subset of patient sessions that have been reviewed and labeled to be clinically normal or abnormal with 99% inter-rater agreement [83]. In order to expand the labeled data set, we developed and applied an automated labeling method based on textual analysis of the clinical reports to the entire NEDC. The automated labeling method matched the known labels with 98.8% accuracy, instilling confidence in the accuracy of the labeling in the expanded data set [94]. Using this expanded data set, we further defined the criteria for the "normal" and "abnormal" populations. The normal population was defined as subjects whose EEG recordings were labeled as normal and were not taking any medications at the

time of recording, resulting in a total of 1,167 recordings across 944 unique subjects. The abnormal population was defined as subjects whose EEG recordings were labeled as abnormal and were taking a clinically relevant anti-epileptic medication [36]. The mean age of the abnormal population, initially 50.39, was significantly higher than the mean age of the normal population (Table A.1). To ensure a similar age distribution within each population the oldest subjects in the abnormal data set were removed sequentially until the mean age was the same across the normal and abnormal populations, resulting in 2,940 abnormal recordings across 1,197 unique subjects. The patient demographics extracted from the reports for the two populations, along with the mean and standard deviation of their ages, are shown in Table A.1.

	Normal	Abnormal
n	1,167	2,940
Male	508	1,418
Female	549	1,347
Sex not determined	110	175
Mean Age	34.12±18.90	34.12±11.41

Table A.1 Normal and abnormal population demographics

A.2.2 Pre-processing and spectral coherence

If necessary, recordings were down-sampled to 100Hz. The data were then filtered through a 0.5Hz to 50Hz band pass filter, and re-referenced using a common average. To establish connectivity we used SC which was defined as

$$C_{xy}(\omega) = \frac{|S_{xy}(\omega)|^2}{S_{xx}(\omega)S_{yy}(\omega)} \tag{A.1}$$

where cross-spectral density $S_{xy}(\omega)$ is the Fourier transform of the cross-correlation function, $R_{xy}(t)$, of $x(t)$ and $y(t)$. That is,

$$S_{xy}(\omega) = \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} x(\tau) \cdot y(\tau + t) d\tau \right] e^{-j\omega t} dt = \int_{-\infty}^{\infty} R_{xy}(t) e^{-j\omega t} dt \tag{A.2}$$

$0 \leq C_{xy}(\omega) \leq 1$ where a value of zero implied there was no spectral correlation and a value of one implied a direct spectral correlation. We computed average coherence values for frequency bands commonly used in EEG analysis: δ(delta) : $1 - 4$Hz, θ(theta) : $4 - 8$Hz, α(alpha) : $8 - 12$Hz, μ(mu) : $12 - 16$Hz, β(beta) : $16 - 25$Hz, γ(gamma) : $25 - 40$Hz [125]. Based on coherence values of two electrode sites (i.e. vertices), we connected them with edges and formed a graph. In this study we say $x \sim y$, x is connected to y, if $C_{xy}(\omega) \geq 0.3$. This threshold was based on the notion that the coherence can be interpreted as a correlation and correlations below 0.3 are typically negligible [91]. To establish a population DMN, we averaged graphs in each population for each frequency band.

A.2.3 Graph parameters

With our connected graphs, we analyzed five parameters of the graph generated for each EEG recording session for a given frequency band. A graph, G, was defined by a set of vertices, V, and edges, E. We defined the degree of $x \in V$, d_x, to be the number of edges connected to vertex x, and the cardinality of V, $|V|$, as the number of vertices in a graph. Further, let A be the adjacency matrix, a $|V| \times |V|$

matrix, and D the degree matrix, a diagonal $|V| \times |V|$ matrix, where

$$A(i,j) = \begin{cases} w_{ij} : \ x_i \sim x_j, i \neq j \\ 0 : \ \text{otherwise} \end{cases} \qquad D(i,i) = \sum_{j=1}^{|V|} A(i,j) \tag{A.3}$$

Note in this definition of A, we considered undirected graphs (i.e. $A(i,j) = A(j,i)$) and did not allow vertices to connect to themselves. In a weighted graph, w_{ij} was the coherence value between the two vertices. In an unweighted graph, $w_{ij} = 1, \forall w_{ij} > 0$. In this study we maintained the coherence values as the weights of the edges. Using A and D we computed three graph connectivity parameters [31].

(i) The number of edges in a graph, $|E|$.

(ii) The graph density,

$$\partial = \frac{2|E|}{|V|(|V|-1)} \tag{A.4}$$

where $\partial = 1$ meant all possible connections have been made.

(iii) The Rich Club coefficient,

$$\phi(k) = \frac{2e_{>k}}{v_{>k}(v_{>k}-1)} \tag{A.5}$$

where $v_{>k}$ was the number of vertices $x \in V$ with $d_x \geq k$ and $e_{>k}$ was the number of edges between those vertices. If $\phi(k)$ was high that meant that the graph had many connections with vertices with a high d_x.

Further, to analyze spectral graph properties we defined the Graph Laplacian,

L, as

$$L = D - A \tag{A.6}$$

From L, two spectral graph parameters were computed [31].

(iv) The algebraic connectivity of a graph, defined as the second smallest eigenvalue of L, λ_2.

(v) The energy of a graph,

$$\Lambda = \sum_{i=1}^{|V|} |Re(\lambda_i)| \tag{A.7}$$

the sum of the absolute values of the real components of the eigenvalues of L.

To determine how different the graphs were between populations we applied a non-parametric Kruskal-Wallis test to each of the five graph parameters computed on each of the six frequency bands.

A.2.4 Fiedler vectors

Using the Graph Laplacian defined in Equation A.6, the eigenvector corresponding to the smallest eigenvalue of L is constant. However, the eigenvector corresponding to the second smallest eigenvalue, known as the Fiedler vector, represents clusters of vertices and partitions of how the graph is connected. The Fiedler vector shows how EEG electrodes group based on connectivity and how they differ across the populations. When computing the spectrum of the graph, any vertices that were not connected to any other vertices were removed. Disconnected graphs were analyzed so long as each subgraph had more than one vertex.

A.3 Results

A.3.1 Spectral coherence connectivity

Figure A.2 shows an example of the averaged DMN across the normal population where the strength of the connections were the coherence values for the α band. Though the majority of the connections fall between 0.3 and 0.5, the strongest connections were present in the occipital-temporal and frontal lobes, between T6-O2, (F3,F4)-Fz, and Fp1-Fp2. The δ, θ, and μ bands show similar connections without the stronger (F3,F4)-Fz connections but with the addition of stronger connections between T5-O1 and O1-O2. The β and γ band DMNs had a significant decrease in the number of connections. In general, the strength of connections and connectivity of the graph decreased as the frequency range of the bands increased. Specifically, the connectivity between the frontal and occipital-temporal regions began to separate as the frequency increased until it was no longer present in the γ band. Furthermore, while there were generally more connections in the DMNs of the abnormal population, the connections present in the DMNs of both populations were generally stronger in the normal population.

A.3.2 Graph parameters

Comparing the normal and abnormal populations, we compute the five graph parameters for each of the six frequency bands. Table A.2 shows the mean and standard deviation of each of the 30 sets across each of the normal and abnormal

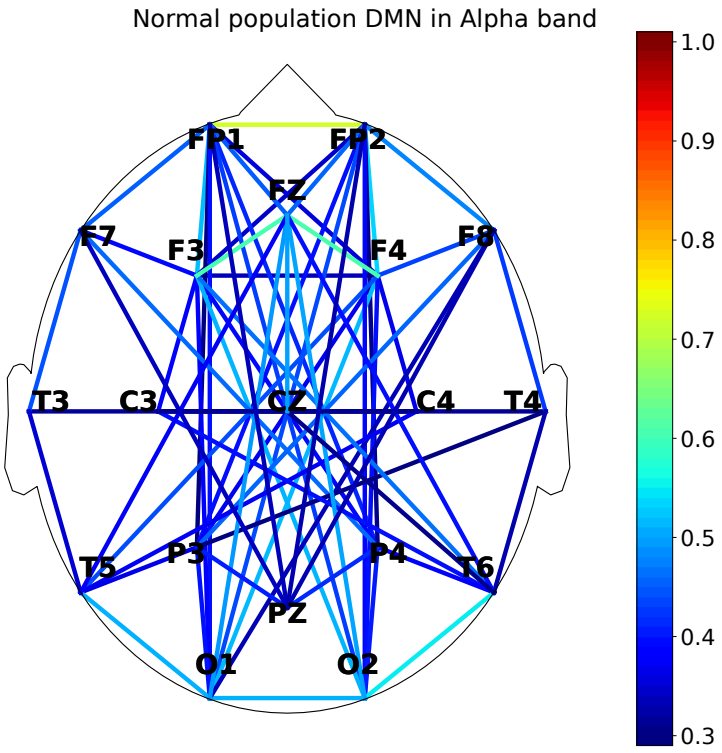


Figure A.2 Color shows connection strength through coherence. Strongest connections seen between T6-O2, (F3,F4)-Fz, and Fp1-Fp2 ($n = 1,167$).

populations. Applying a Kruskal-Wallis test we bold the pairs which were found to be significantly different between the two populations. Table A.2 shows the frequency bands were more influential than the graph parameters for whether the two groups would be statistically different. The θ, α, and γ bands showed most or all graph parameters to differ significantly with $p < 0.0001$, unless otherwise noted.

A.3.3 Fiedler vectors

The Fiedler vectors of the Graph Laplacian are shown in Figure A.3. The colors represent the eigenvector's values at each vertex. Strong positive and negative values, as well as near-zero values represented prominent groups of vertices of the

Parameter	Population	δ (Delta)	θ (Theta)	α (Alpha)	μ (Mu)	β (Beta)	γ (Gamma)
# Edges - $\|E\|$	Normal	62.29 ± 31.63	**51.81*** ± 26.48**	**63.43*** ± 29.24**	42.06 ± 27.37	**29.26*** ± 26.21**	**23.81*** ± 27.04**
	Abnormal	63.00 ± 29.02	**58.00*** ± 27.26**	**56.82*** ± 29.50**	45.24 ± 31.50	**35.01*** ± 32.27**	**29.07*** ± 33.04**
Density - ∂	Normal	0.36 ± 0.18	**0.30*** ± 0.15**	**0.37*** ± 0.17**	0.25 ± 0.16	**0.17*** ± 0.15**	**0.14*** ± 0.16**
	Abnormal	0.37 ± 0.17	**0.34*** ± 0.16**	**0.33*** ± 0.17**	0.26 ± 0.18	**0.20*** ± 0.19**	**0.17*** ± 0.19**
Rich Club - $\overline{\phi}$	Normal	**0.62*** ± 0.17**	0.52* ± 0.17	**0.60*** ± 0.17**	0.48 ± 0.18	0.42* ± 0.19	0.4** ± 0.20
	Abnormal	**0.59*** ± 0.17**	0.54* ± 0.17	**0.55*** ± 0.18**	0.48 ± 0.19	0.44* ± 0.20	0.44** ± 0.22
Connectivity - λ_2	Normal	**0.47*** ± 1.50**	**0.43*** ± 1.37**	**0.48*** ± 0.90**	0.30 ± 1.45	0.21 ± 1.44	**0.19*** ± 1.61**
	Abnormal	**0.52*** ± 1.31**	**0.54*** ± 1.47**	**0.51*** ± 1.57**	0.44 ± 1.82	0.32 ± 1.79	**0.30*** ± 1.89**
Energy - Λ	Normal	64.96 ± 47.39	**50.21*** ± 38.53**	**63.06*** ± 36.67**	39.61 ± 38.54	**27.48*** ± 37.64**	**23.91*** ± 41.36**
	Abnormal	65.03 ± 44.57	**58.06*** ± 42.56**	**56.84*** ± 44.63**	44.96 ± 48.36	**35.37*** ± 49.03**	**30.82*** ± 51.37**

Table A.2 Mean ± stand. dev. of graph parameters per frequency band across populations - **Bold** ***: $p < 0.0001$; **: $p < 0.0015$; *: $p < 0.05$

Fiedler vector. The scale of the color bars depended on the range of the eigenvector to visualize these groups since each Fiedler vector shown was derived from a different graph. The differences between the clusters of the eigenvector's values, rather than the values themselves, were analyzed between the two populations. In θ, α, μ, and β bands, the clusters in the abnormal population are more central and globally connected than in the normal population.

A.4 Discussion

This study explored differences in connectivity in resting EEG data between a normative and abnormal population from a large clinical data set. Unlike previous works where subjects' EEGs were recorded under known conditions and commonly performing instructed tasks, it is not known which activities or tasks the subjects were performing during EEG recordings used in this study. However, we note that

an ever-increasing volume of available human EEG data will share this task-agnostic characteristic, and so it is important to understand these types of data. Spectral graph parameters λ_2 and Λ values were higher in the abnormal population as previously observed in Dasgupta et al. [31]. There was also generally greater variability in these two parameters in the abnormal population [31]. However, our parameter results show there were generally more connections in the abnormal population's graphs as compared to previous work [31]. The Fiedler vectors of the Graph Laplacian showed a similar trend as well. This may have been from differences in how the graphs were formed. Alternatively, it is possible that these differences are from abnormal neural activity propagating throughout the brain given the the clinical nature of this data set and the fact that individuals included in this abnormal subset were actively taking anti-epileptic drugs.

These results present connectivity in the sensor space of EEG which has applications in medical devices and analytic tools. However, further analysis is needed to better understand the neurophysiological implications and source connectivity. Future work will investigate connectivity correlates with other patient characteristics such as age and sex. Graph theoretic neural connectivity methods will also be explored to better model machine learning architectures and as features in classification applications.

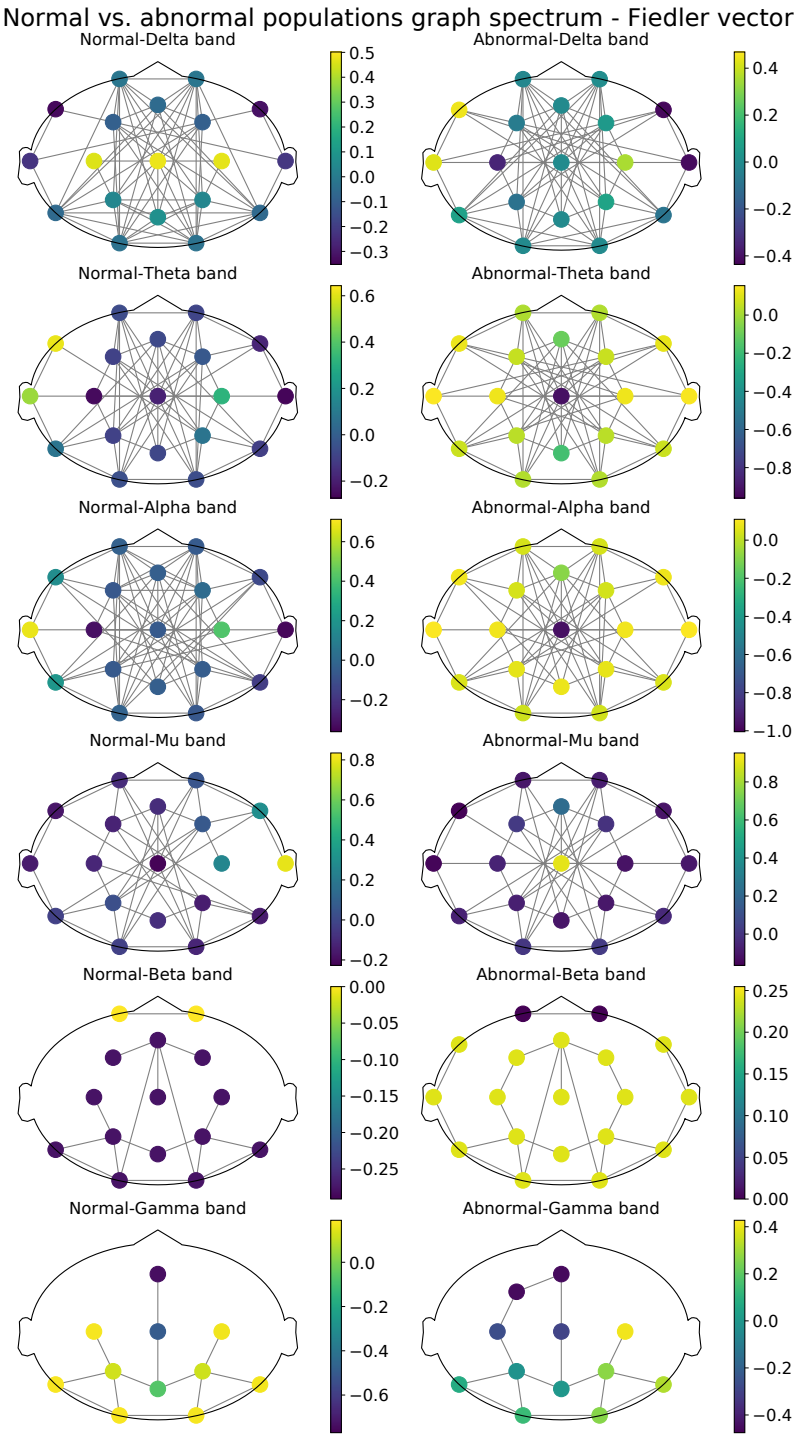


Figure A.3 Fiedler vectors of the graph plotted for each frequency band for Normal ($n = 1,167$) and Abnormal ($n = 2,940$) populations. Color bar scales are different to permit visualization of clusters of positive, negative and near-zero groups based on values of the Fiedler vector.

Bibliography

[1] Aapo Hyvärinen and Juha Karhunen and Erkki Oja. *Independent Component Analysis*. John Wiley & Sons, 2001.

[2] Bassel Abou-Khalil. Levetiracetam in the treatment of epilepsy. *Neuropsychiatric disease and treatment*, 4(3):507 – 523, 2008. doi: https://doi.org/10.2147/ndt.s2937.

[3] K. K. Ang, Z. Y. Chin, H. Zhang, and C. Guan. Filter Bank Common Spatial Pattern (FBCSP) in Brain-Computer Interface. *IEEE International Joint Conference on Neural Networks, 2008. IJCNN 2008. (IEEE World Congress on Computational Intelligence)*, pages 2390–2397, 2008.

[4] Simon Authier, Marcus S. Delatte, Mary-Jeanne Kallman, Joanne Stevens, and Carrie Markgraf. Eeg in non-clinical drug safety assessments: Current and emerging considerations. *Journal of Pharmacological and Toxicological Methods*, 81:274–285, 2016.

[5] Yang Bai, Liang Zhenhu, and Xiaoli Li. A permutation Lempel-Ziv complexity measure for EEG analysis. *Biomedical Signal Processing and Control*, 19:102–114, 2015.

[6] Dobkin D.P. Barber, C.B. and H.T. Huhdanpaa. The Quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software*, 22(4):469–483, 1996.

[7] M. S. Bartlett. Smoothing Periodograms from Time-Series with Continuous Spectra. *Nature*, 161:686–687, 1948.

[8] M. S. Bartlett. Periodogram Analysis and Continuous Spectra. *Biometrika*, 37(1-2):1–16, 1950.

[9] A. Belouchrani, K. Abed-Meraim, J. . Cardoso, and E. Moulines. A blind source separation technique using second-order statistics. *IEEE Transactions on Signal Processing*, 45(2):434–444, Feb 1997. ISSN 1941-0476. doi: 10.1109/78.554307.

[10] John J. Benedetto. *Harmonic Analysis and Applications*. CRC Press, 1996.

[11] Hans Berger. Fifty Years of EEG: Fiftieth Anniversary of Hans Berger's Publication of the Electroencephalogram His First Records in 1924-1931. *Archives of Psychiatry and Neurological Sciences*, 227:279–300, 1979.

[12] M. Bewernitz and H. Derendorf. Electroencephalogram-based phramacodynamic measures: a review. *International journal of clinical pharmacology and therapeutics*, 50(3):162–184, 2012.

[13] Christopher Bishop. *Pattern Recognition and Machine Learning*. Springer, 2009.

[14] Susana Blanco, Arturo Gara, and Diego Coulombie. Comparison of Frequency Bands Using Spectral Entropy for Epileptic Seizure Prediction. *ISRN Neurology*, 2013, 2013. ID 287327.

[15] Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik. A Training Algorithm for Optimal Margin Classifiers. *Proceedings of the fifth annual workshop on Computational learning theory*, pages 144–152, 1992.

[16] Susan M. Bowyer. Coherence a Measure of the Brain Network: Past and Present. *Neuropsychiatric Electrophysiology*, 2(1), 2016.

[17] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[18] Broyd et al. Default-Mode Brain Dysfunction in mental disorders: A Systematic Review. *Neuroscience and Biobehavioral Reviews*, 33(3):279–296, 2009. doi: 10.1016/j.neubiorev.2008.09.002.

[19] Christopher J. C. Burges. A Tutorial on Support Vector Machines for Pattern Recognition. *Data Mining and Knowledge Discovery*, 2:121–167, 1998.

[20] Chen et al. EEG Default Mode Network in the Human Brain: Spectral Regional Field Powers. *NeuroImage*, 41(2):561–574, 2008.

[21] Z. Y. Chin, K. K. Ang, C. Wang, C. Guan, and H. Zhang. Multi-class filter bank common spatial pattern for four-class motor imagery BCI. *Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 571–574, 2009.

[22] Travers Ching, Daniel S. Himmelstein, Brett K. Beaulieu-Jones, Alexandr A. Kalinin, Brian T. Do, Gregory P. Way, Enrico Ferrero, Paul-Michael Agapow, Michael Zietz, Michael M. Hoffman, Wei Xie, Gail L. Rosen, Benjamin J. Lengerich, Johnny Israeli, Jack Lanchantin, Stephen Woloszynek, Anne E. Carpenter, Avanti Shrikumar, Jinbo Xu, Evan M. Cofer, Christopher A. Lavender, Srinivas C. Turaga, Amr M. Alexandari, Zhiyong Lu, David J. Harris, Dave DeCaprio, Yanjun Qi, Anshul Kundaje, Yifan Peng, Laura K. Wiley, Marwin H. S. Segler, Simina M. Boca, S. Joshua Swamidass, Austin

Huang, Anthony Gitter, and Greene Casey S. Opportunities and obstacles for deep learning in biology and medicine. *Journal of the Royal Society Interface*, 15(141), 2018. doi: http://doi.org/10.1098/rsif.2017.0387.

[23] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs). *ICLR*, 2016.

[24] GBD 2016 Neurology Collaborators. Global, regional, and national burden of neurological disorders, 1990–2016: a systematic analysis for the global burden of disease study 2016. *The Lancet Neurology*, 18(5):459 – 480, 2019. ISSN 1474-4422. doi: https://doi.org/10.1016/S1474-4422(18)30499-X. URL `http://www.sciencedirect.com/science/article/pii/S147444221830499X`.

[25] Corinna Cortes and Vladimir N. Vapnik. Support-Vector Networks. *Machine Learning*, 20:273–297, 1995.

[26] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. John Wiley & Sons, 2006.

[27] Alexander Craik et al. Deep learning for Electroencephalogram (EEG) classification tasks: A review. *Journal of Neural Engineering*, 16(3), 2019.

[28] G. Cybenko. Approximation by Superposition of a Sigmoidal Function. *Mathematics of Control, Signals, and Systems*, 2:303–314, 1989.

[29] Maryann D'Allessandro, Rosana Esteller, George Vachtesevanos, Arthur Hinson, Javier Echauz, and Brian Litt. Epileptic Seizure Prediction Using Hybrid Feature Selection Over Multiple Intracranial EEG Electrode Contacts: A Report on Four Patients . *IEEE Transactions on Biomedical Engineering*, 50(5): 603–615, 2003. doi: 10.1109/tbme.2003.810706.

[30] Ian Daly et al. What does clean EEG look like? In *Engineering in Medicine and Biology Society (EMBS), 2012 34th International Conference of the*, pages 3963–3966. IEEE, 2012.

[31] Abhijit Dasgupta, Ritankar Das, Losiana Nayak, and Rajat De. Analyzing epileptogenic brain connectivity networks using clinical EEG data. *IEEE International Conference on Bioinformatics and Biomedicine*, 2015.

[32] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification*. John Wiley & Sons, 2 edition, 2000.

[33] Zack Dvey-Aharon, Noa Fogelson, Abraham Peled, and Nathan Intrator. Connectivity maps based analysis of EEG for advanced diagnosis of schizophrenia attributes. *PLoS ONE*, 2017.

[34] D.W. Scott. *Multivariate Density Estimation: Theory, Practice, and Visualization*. John Wiley & Sons, 1992.

[35] Marjolein M. A. Engels, Cornelis J. Stam, Wiesje M. van der Flier, Philip Scheltens, Hanneke de Waal, and Elisabeth C. W. van Straaten. Declining Functional Connectivity and Changing Hub Location in Alzheimer's disease: an EEG study. *BMC Neurology*, 145(15), 2017.

[36] "Epilepsy Society". List of anti-epileptic drugs. https://www.epilepsysociety.org.uk/list-anti-epileptic-drugs, 2018. Accessed: 2018-09-17.

[37] Jean-Marc Fellous, Guillermo Sapiro, Andrew Rossi, Helen Mayberg, and Michele Ferrante. Explainable artificial intelligence for neuroscience: Behavioral neurostimulation. *Frontiers in Neuroscience*, 13:1346, 2019. ISSN 1662-453X. doi: 10.3389/fnins.2019.01346. URL `https://www.frontiersin.org/article/10.3389/fnins.2019.01346`.

[38] R. Feltcher. *Practical Methods of Optimization*. John Wiley & Sons, 1987.

[39] Thalia Fernandez, Thalia Harmony, Mario Rodriguez, Alfonso Reyes, Erzsebet Marosi, and Jorge Bernal. Test-restest reliability of EEG spectral parameters during cognitive tasks: I Absolute and relative power. *International Journal of Neuroscience*, 68:255–261, 2009.

[40] Alexander Fingelkurts, Andrew Fingelkurts, Victor Ermolaev, and Alexander Kaplan. Stability, reliability and consistency of composition of brain oscillations. *International Journal of Psychophysiology*, 59:116–126, 2006.

[41] Jessica I. Fleck, Julia Kuti, Jeffery Mercurio, Spencer Mullen, Katherine Austin, and Pereira Olivia. The Impact of Age and Cognitive Reserve on Resting-State Brain Connectivity. *Frontiers in Aging Neuroscience*, 9(392), 2017.

[42] Theo Gasser, Petra Bächer, and Hans Steinberg. Test-retest reliability of spectral parameters of the EEG. *Electroencephalography and Clinical Neurophysiology*, 60:312–319, 1985.

[43] Donald L. Gilbert, Gopalan Sethuraman, Uma Kotagal, and C. Ralph Buncher. Meta-analysis of EEG test performance shows wide variation among studies. *Neurology*, 60:564–570, 2002.

[44] Gene H. Golub and Charles F. Van Loan. *Matrix Computations (3rd Ed.)*. Johns Hopkins University Press, USA, 1996. ISBN 0801854148.

[45] G. Gomez-Herrero, W. D. Clercq, H. Anwar, O. Kara, K. Egiazarian, S. V. Huffel, and W. V. Paesschen. Automatic removal of ocular artifacts in the eeg without an eog reference channel. In *Proceedings of the 7th Nordic Signal Processing Symposium - NORSIG 2006*, pages 130–133, June 2006. doi: 10.1109/NORSIG.2006.275210.

[46] González et al. Graph Analysis of EEG Resting State Functional Networks in Dyslexic Readers. *Clinical Neurophysiology*, 127:3165–3175, 2016.

[47] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016.

[48] Fanny Grosselin et al. Quality Assessment of Single-Channel EEG for Wearable Devices. *Sensors*, 19(601), 2019.

[49] Steinn Gudmundsson, Thomas Runarsson, Sven Sigurdsson, Gudrun Eiriksdottir, and Kristinn Johnsen. Reliability of quatitative EEG features. *Clinical Neurophysiology*, 118:2162–2171, 2007.

[50] Steinn Gudmundsson, Thomas Runarsson, and Sven Sigurdsson. Test-restest reliability and feature selection in physiological time series classification. *Preprint*, 2010.

[51] A. Harati, S. Lopez, I. Obeid, M. Jacobson, S. Tobochnik, and J. Picone. The TUH EEG Corpus: A Big Data Resource for Automated EEG Interpretation. *In Proceedings of the IEEE Signal Processing in Medicine and Biology Symposium*, page 1–5, 2014.

[52] B. Hassibi and T. Kailath. H-infinity adaptive filtering. *Proceeding of the 1995 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 949–952, May 1995.

[53] Monson H. Hayes. *Statistical Digital Signal Processing and Modeling*. John Wiley & Sons, 1996.

[54] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. *arXiv: 1512.03385*, 2015.

[55] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989.

[56] Bin Hu et al. Signal Quality Assessment Model for Wearable EEG Sensor on Prediction of Mental Stress. *IEEE Transactions on Nanobioscience*, 14(5): 553–561, 2013.

[57] A. Hyvarinen. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Transactions on Neural Networks*, 10(3):626–634, May 1999. ISSN 1941-0093. doi: 10.1109/72.761722.

[58] A. Hyvärinen and E. Oja. Independent component analysis: algorithms and applications. *Neural Networks*, 13(4):411 – 430, 2000. ISSN 0893-6080. doi: https://doi.org/10.1016/S0893-6080(00)00026-5. URL `http://www.sciencedirect.com/science/article/pii/S0893608000000265`.

[59] Antonio Ibáñez-Molina, Sergio Iglesias-Parro, Maria Soriano, and José Aznarte. Multiscale Lempel-Ziv complexity for EEG measures. *Clinical Neurophysiology*, 126:541–548, 2015.

[60] Imperatori et al. Default Mode Network Alternations in Alexithymia: An EEG Power Spectra and Connectivity Study. *Nature-Scientific Reports*, 6 (36653), 2016.

[61] S. Ioe and C. Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *Proceedings of The 32nd International Conference on Machine Learning*, pages 448–456, 2015.

[62] Franziska Horn Eric Waldburger Carsten Allefeld Irene Winkler, Stephanie Brandl and Michael Tangermann. Automatic Classification of Artifactual ICA-Components for Artifact Removal in EEG Signals. *Journal of Neural Engineering*, 11(035013), 2014.

[63] Md Kafiul Islam, Amir Rastegarnia, and Zhi Yang. Methods for artifact detection and removal from scalp EEG: A review. *Clincal Neurophysiology*, 46 (4-5):287–305, 2016. doi: doi.org/10.1016/j.neucli.2016.07.002.

[64] H. H. Jasper. The Ten-Twenty Electrode System of the International Federation. *Electroencephalography and Clinical Neurophysiology*, 10:371–375, 1958.

[65] D. C. Javitt, K. M. Spencer, G. K. Thaker, G. Winterer, and M Hajós. Neurophysiological biomarkers for drug development in schizophrenia. *Nature reviews. Drug discovery*, 7(1):68–83, 2008.

[66] Natalia Jaworska, Sara de la Salle, Mohamed-Hamza Ibrahim, Pierre Blier, and Verner Knott. Leveraging machine learning approaches for predicting antidepressant treatment response using electroencephalography (eeg) and clinical data. *Frontiers in Psychiatry*, 9:768, 2019. ISSN 1664-0640. doi: 10.3389/fpsyt.2018.00768. URL `https://www.frontiersin.org/article/10.3389/fpsyt.2018.00768`.

[67] O. Jensen, P. Goel, N. Kopell, M. Pohja, R. Hari, and B. Ermentrout. On the human sensorimotor-cortex beta rhythm: Sources and modeling. *NeuroImage*, 26(2):347 – 355, 2005. ISSN 1053-8119. doi: https://doi.org/10.1016/j.neuroimage.2005.02.008. URL `http://www.sciencedirect.com/science/article/pii/S1053811905001060`.

[68] Xiao Jiang, Gui-Bin Bian, and Zean Tian. Removal of Artifacts from EEG Signals: A Review. *Sensors*, 19(987), 2019. doi: doi.org/10.3390/s19050987.

[69] T-P Jung, S Makeig, C Humphries, TW Lee, MJ McKeown, V Iragui, and TJ Sejnowski. Removing Electroencephalographic Artifacts by Blind Source Separation. *Psychophysiology*, 37:163–178, 2000.

[70] T-P Jung, S Makeig, W Westerfield, J Townsend, E Courchesne, and TJ Sejnowski. Removal of eye activity artifacts from visual event-related potentials in normal and clinical subjects. *Clinical Neurophysiology*, 111:1745–1758, 2000.

[71] Anders Bach Justesen, Mette Thrane Foged, Martin Fabricius, Christian Skaarup, Nizar Hamrouni, Terje Martens, Olaf B. Paulson, Lars H. Pinborg, and Sándor Beniczky. Diagnostic yield of high-density versus low-density eeg: The effect of spatial sampling, timing and duration of recording. *Clinical Neurophysiology*, 130(11):2060 – 2064, 2019. ISSN 1388-2457. doi: https://doi.org/10.1016/j.clinph.2019.08.007. URL `http://www.sciencedirect.com/science/article/pii/S1388245719311939`.

[72] Atilla Kilicarslan and Jose Luis Contreras Vidal. Characterization and real-time removal of motion artifacts from eeg signals. *Journal of Neural Engineering*, 16(056027), 2019. doi: 10.1088/1741-2552/ab2b61.

[73] Atilla Kilicarslan, Robert G. Grossman, and Jose Luis Contreras Vidal. A robust adaptive denoising framework for real-time artifact removal in scalp eeg measurements. *Journal of Neural Engineering*, 13(026013), 2016. doi: 10.1088/1741-2560/13/2/026013.

[74] D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, 2014.

[75] Kimberly Kontson, Murad Megjhani, Justin Brantley, Jesus Cruz-Garza, Sho Nakagome, Dario Robleto, Michelle White, Eugene Civillico, and Jose Contreras-Vidal. 'your brain on art': Emergent cortical dynamics during aesthetic experiences. *Frontiers in Human Neuroscience*, 9:626, 2015. ISSN 1662-5161. doi: 10.3389/fnhum.2015.00626. URL `https://www.frontiersin.org/article/10.3389/fnhum.2015.00626`.

[76] A. Korotcov, V. Tkachenko, D. P. Russo, and S. Ekins. Comparison of deep learning with multiple machine learning methods and metrics using diverse drug discovery data sets. *Molecular pharmaceutics*, 14(12):4462–4475, 2017. doi: 10.1021/acs.molpharmaceut.7b00578.

[77] William H. Kruskal and W. Allen Wallis. Use of Ranks in One-Criterion Variance Analysis. *Journal of the American Statistical Association*, 47(260): 583–621, 1952.

[78] Chi Qin Lai et al. Literature survey on applications of electroencephalography (EEG). *AIP Conference Proceedings*, 2016(020070), 2018.

[79] H. O. Lancaster. *The Chi-squared distribution*. Wiley, 1969.

[80] Lawhern, Vernon J. and Solon, Amelia J. and Waytowich, Nicholas, R. and Gordon, Stephen M. and Hung, Chou P. and Lance, Brent J. EEGNet: A

Compact Convolutional Neural Network for EEG-based Brain-Computer Interfaces. *arXiv:1611.08024v4*, 2018.

[81] A. Lenartowicz and S. K. Loo. Use of EEG to Diagnose ADHD. *Current Psychiatry Reports*, 16(11):498, 2014.

[82] Hallie Levine. From aha! moment to fda approval: The fascinating story behind a new treatment for bladder cancer. *jnj.com*, Apr 2019. URL `https://www.jnj.com/innovation/fda-approves-new-bladder-cancer-treatment`.

[83] S. Lopez. Automated Identification of Abnormal EEGs. *Temple University*, 2017.

[84] F. Lotte, M. Congedo, A. Lecuyer, F. Lamarche, and B. Arnaldi. A review of classification algorithms for EEG-based brain-computer interfaces. *Journal of Neural Engineering*, 4(2), 2007.

[85] F. Lotte, L. Bougrain, A. Cichocki, M. Clerc, M. Congedo, A. Rakotomanonjy, and F. Yger. A review of classification algorithms for EEG-based brain-computer interfaces: a 10 year update. *Journal of Neural Engineering*, 15 (031005), 2018.

[86] S. López, G. Suarez, D. Jungreis, I. Obeid, and J. Picone. Automated identification of abnormal adult eegs. In *IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, Dec 2015. doi: 10.1109/SPMB.2015.7405423.

[87] José Maltez, Lars Hyllienmark, Vadim Nukulin, and Tom Brismar. Time course and variability of power in different frequency bands of EEG during resting conditions. *Clinical Neurophysiology*, 34:195–202, 2004.

[88] Eric Maris and Robert Oostenveld. Nonparametric statistical testing of EEG- and MEG-data. *Journal of Neuroscience Methods*, 164:177–190, 2007.

[89] Hidenori Matsuzaki. Adaptive second-order volterra filtering based on an h^{infty} criterion. *SICE Journal of Control, Measurement, and System Integration*, 9(1):26–32, 2016. doi: 10.9746/jcmsi.9.26.

[90] Kimford J. Meador, David W. Loring, Alan Boyd, Javier Echauz, Suzette LaRoche, Naymee Velez-Ruiz, Pearce Korb, William Byrnes, Deanne Dilley, Simon Borghs, Marc De Backer, Tyler Story, Peter Dedeken, and Elizabeth Webster. Randomized double-blind comparison of cognitive and eeg effects of lacosamide and carbamazepine. *Epilepsy & Behavior*, 62:267–275, 2016.

[91] M. M. Mukaka. A Guide to Appropriate Use of Correlation Coefficient in Medical Research. *Malawi Medical Journal: The Journal of Medical Association of Malawi*, 24(3), 2012. URL `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3576830/`.

[92] Naereen. Lempel-ziv complexity for a binary sequence, in naive python code., 2017. URL `https://pypi.python.org/pypi/Lempel-Ziv_Complexity`.

[93] D. O. Nahmias and K. L. Kontson. EEG Spectral Connectivity Analysis in a Large Clinical Population. In *2019 9th International IEEE/EMBS Conference on Neural Engineering (NER)*, pages 41–44, March 2019. doi: 10.1109/NER.2019.8716884.

[94] D. O. Nahmias, K. L. Kontson, and E. F. Civillico. Learning EEG: Identification of novel electroencephalogram classifications and variability of baseline features in a large clinical dataset. In *2017 International Symposium on Wearable Robotics and Rehabilitation (WeRob)*, pages 1–2, Nov 2017. doi: 10.1109/WEROB.2017.8383856.

[95] David O. Nahmias, Kimberly L. Kontson, David A. Soltysik, and F. Civillico, Eugene. Consistency of quantitative electroencephalography features in a large clinical data set. *Journal of Neural Engineering*, 16(066044), 2019. doi: doi.org/10.1088/1741-2552/ab4af3.

[96] Makus Näpflin, Marc Wildi, and Johannes Sarnthein. Test-restest reliability of EEG spectra validates a statistical signature of persons. *Clinical Neurophysiology*, 118:2519–2524, 2007.

[97] Makus Näpflin, Marc Wildi, and Johannes Sarnthein. Test-restest reliability of EEG spectra during a working memory task. *NeuroImage*, 43:687–693, 2008.

[98] F. Nasse, C. Thurau, and G. A. Fink. A review of classification algorithms for EEG-based brain-computer interfaces: a 10 year update. *International Conference on Computer Analysis of Images and Patterns*, pages 83–90, 2009.

[99] Jennifer J. Newson and Tara C. Thiagarajan. Eeg frequency bands in psychiatric disorders: A review of resting state studies. *Frontiers in Human Neuroscience*, 12:521, 2019. ISSN 1662-5161. doi: 10.3389/fnhum.2018.00521. URL `https://www.frontiersin.org/article/10.3389/fnhum.2018.00521`.

[100] Robert Oostenvelda and Peter Praamstrac. The five percent electrode system for high-resolution EEG and ERP measurements. *Clinical Neurophysiology*, 112:713–719, 2001.

[101] Alan V. Oppenheim and Ronald W. Schafer. *Discrete-Time Signal Processing*. Prentice-Hall, 3 edition, 2009.

[102] Natasha Padfield, Jaime Zabalza, Huimin Zhao, Valentin Masero, and Jinchan Ren. Eeg-based brain-computer interfaces using motor-imagery: Techniques and challenges. *Sensors*, 19(6)(1423), 2019. doi: 10.3390/s19061423.

[103] Emanuel Parzen. On estimation of a probability density function and mode. *Ann. Math. Statist.*, 33(3):1065–1076, 09 1962. doi: 10.1214/aoms/1177704472. URL `https://doi.org/10.1214/aoms/1177704472`.

[104] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. *NIPS-W*, 2017. URL `https://pytorch.org`.

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

[106] Elvira Pirondini, Martina Coscia, Jesus Minguillon, José del R. Millán, Dimitri Van De Ville, and Silvestro Micera. EEG topographies provide subject-specific correlates of motor control. *Nature, Scientific Reports*, 7(13229), 2017.

[107] MR Poynton, BM Choi, YM Kim, IS Park, GJ Noh, SO Hong, YK Boo, and SH Kang. Machine learning methods applied to pharmacokinetic modelling of remifentanil in healthy volunteers: A multi-method comparison. *Journal of International Medical Research*, 37(6):1680–1691, 2009. doi: 10.1177/147323000903700603.

[108] J. Proakis, Charles M. Rader, Fuyun Ling, and Christo Nikias. *Advanced Digital Signal Processing*. Macmillan, 1992.

[109] John G. Proakis and Dimitris G. Manolakis. *Digital Signal Processing*. Pearson, 4 edition, 2006.

[110] V. O. Puntmann. How-to guide on biomarkers: biomarker definitions, validation and applications with examples from cardiovascular disease. *Postgraduate Medical Journal*, 85:538–545, 2009.

[111] Raichle et al. A default mode of the brain. *PNAS*, 98(2):676–682, 2001.

[112] Alvin Rajkomar, Jeffrey Dean, and Isaac Kohane. Machine learning in medicine. *New England Journal of Medicine*, 380(14):1347–1358, 2019. doi: 10.1056/NEJMra1814259. URL `https://www.nejm.org/doi/full/10.1056/NEJMra181425`. PMID: 30943338.

[113] Normadiah M. Razali and Yap B. Wah. Power Comparisons of Shapiro-Wilk, Kolmogorov-Smirnov, Lilliefors and Anderson-Darling Tests. *Journal of Statistical Modeling and Analytics*, 2(1):21–33, 2011.

[114] John A. Rice. *Mathematical Statistics and Data Analysis*. Duxbury, 3 edition, 2010.

[115] M. Rogawski and W Löscher. The neurobiology of antiepileptic drugs. *Nature Review Neuroscience*, 5:553 – 564, 2004. doi: https://doi.org/10.1038/nrn1430.

[116] Murray Rosenblatt. Remarks on some nonparametric estimates of a density function. *Ann. Math. Statist.*, 27(3):832–837, 09 1956. doi: 10.1214/aoms/1177728190. URL https://doi.org/10.1214/aoms/1177728190.

[117] Yannick Roy et al. Deep learning-based electroencephalography analysis: a systematic review. *Journal of Neural Engineering*, 16(051001), 2019. doi: 10.1088/1741-2552/ab260c/.

[118] Guillermo Sahonero-Alvarez and Humberto Calderon. A comparison of sobi, fastica, jade and infomax algorithms. *Proceedings of the 8th International Multi-Conference on Complexity, Informatics and Cybernetics (IMCIC 2017)*, pages 17–22, 2017.

[119] M.C. Salinsky, B.S. Oken, and L. Morehead. Test-retest in EEG frequency analysis. *Electroencephalography and Clinical Neurophysiology*, 79:382–392, 1991.

[120] W. Samek, A. Binder, G. Montavon, S. Lapuschkin, and K. Müller. Evaluating the visualization of what a deep neural network has learned. *IEEE Transactions on Neural Networks and Learning Systems*, 28(11):2660–2673, Nov 2017. ISSN 2162-2388. doi: 10.1109/TNNLS.2016.2599820.

[121] Udit Satija et al. A review of Signal Processing Techniques for Electrocardiogram Signal Quality Assessment. *IEEE Reviews in Biomedical Engineering*, 11:36–52, 2018.

[122] J. Sayers. The world health report 2001—mental health: new understanding, new hope. *World Health Organization Bulletin*, 79:1085–1085, 2001. doi: https://doi.org/10.1016/j.clinph.2019.08.007. URL http://www.sciencedirect.com/science/article/pii/S1388245719311939.

[123] R. Schirrmeister, L. Gemein, Eggensperger, F. Hutter, and T. Ball. Deep learning with convolutional neural networks for decoding and visualization of EEG pathology. *The IEEE Signal Processing in Medicine and Biology Symposium*, December 2017. doi: 10.1109/SPMB.2017.8257015.

[124] B Schölkopf, K. Sung, C. Burges, F. Girosi, P. Niyogi, T. Poggio, , and V. Vapnik. Comparing support vector machines with gaussian kernels to radial basis function classifiers. *IEEE Transactions in Signal Processing*, 45:2758–2765, 1997.

[125] Donald L. Schomer and Fernando H. Lopes da Silva. *Niedermeyer's Electroencephalography: Basic Principles, Clinical Applications, and Related Fields*. Lippincott Williams & Wilkins, 6 edition, 2010.

[126] S. S. Shapiro and M.B. Wilk. An Analysis of Variance Test for Normality. *Biometrika*, 52(3/4):591–611, 1965.

[127] J. Shen, C. Zhang, B. Jiang, J. Chen, Z. Song, J.and Liu, and W. K.. Ming. Artificial intelligence versus clinicians in disease diagnosis: Systematic review. *JMIR medical informatics*, 7(3)(e10010), 2019. doi: 10.2196/10010.

[128] Jessica Smith. U.s. fda approves ibrance (palbociclib) for the treatment of men with hr , her2- metastatic breast cancer. *pfizer.com*, Apr 2019. URL `https://www.pfizer.com/news/press-release/press-release-detail/u_s_fda_approves_ibrance_palbociclib_for_the_treatment_of_men_with_hr_her2_metastatic_breast_cancer`.

[129] S. M. Snyder, T. A. Rugino, M. Hornig, and M. A. Stein. Integration of an EEG biomarker with a clinician's ADHD evaluation. *Brain and Behavior*, 5 (4):5–6, 2015.

[130] David Soltysik. Reproducibility of EEG Power Spectral Density. *Proceedings of the 22nd Annual Meeting of the Organization of Human Brain Mapping*, 1 (1736), June 2016.

[131] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[132] Adbulhamit Subasi and M. Ismail Gursoy. EEG signal classification using PCA, ICA, LDA and support vector machines. *Expert Systems with Applications*, 37:8659–8666, 2010.

[133] J. Tang, R. Liu, Y. L. Zhang, M. Z. Liu, M. J. Hu, Y. F.and Shao, and W. Zhang. Application of machine-learning models to predict tacrolimus stable dose in renal transplant recipients. *Nature: Scientific reports*, 7(42192):1680–1691, 2017. doi: 10.1038/srep42192.

[134] William O. Tatum IV, Aatif M. Husain, Selim R. Bendadis, and Peter W. Kaplan. *Handbook of EEG Interpretation*. Demos, 2008.

[135] Pierre Thodoroff, Joelle Pineau, and Andrew Lim. Learning robust features using deep learning for automatic seizure detection. *CoRR*, abs/1608.00220, 2016. URL `http://arxiv.org/abs/1608.00220`.

[136] Trevor Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer, 2 edition, 2009.

[137] P. P. Vaidyanathan. *Multirate Systems and Filter Banks*. Prentice Hall, 1993.

[138] Michel J.A.M. vanPutten, Sebastian Olbrich, and Martijn Arns. Predicting sex from brain rhythms with deep learning. *Nature: Scientific Reports*, 8 (3069), 2018.

[139] Vladimir N. Vapnik. Estimation of Dependences Based on Empirical Data [in Russian]. *Nauka*, 1979.

[140] Vladimir N. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, 1995.

[141] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, CJ Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake Vand erPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1. 0 Contributors. SciPy 1.0–Fundamental Algorithms for Scientific Computing in Python. *arXiv e-prints*, art. arXiv:1907.10121, Jul 2019.

[142] Peter D. Welch. The use of Fast Fourier Transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms. *IEEE Transactions on Audio and Electroacoustics*, AU-15(2):70–73, 1967.

[143] Irene Winkler, Stefan Haufe, and Michael Tangermann. Automatic Classification of Artifactual ICA-Components for Artifact Removal in EEG Signals. *Behavioral and Brain Functions*, 7(30), 2011.

[144] World Health Organization. *Neurological Disorders: Public Health Challenges*. World Health Organization, 2006. ISBN 9789241563369.

[145] Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 818–833, Cham, 2014. Springer International Publishing. doi: https://doi.org/10.1007/978-3-319-10590-1_53.

[146] L. Zhao, Y. P. Wu, J. L. Qi, Y. Q. Liu, K. Zhang, and W. L. Li. Efficacy of levetiracetam compared with phenytoin in prevention of seizures in brain injured patients: A meta-analysis. *Medicine*, 97(48), 2018. doi: 10.1097/MD.0000000000013247.

[147] Y. T. Zhou and R. Chellappa. Stereo matching using a neural network. *International Conference on Acoustics, Speech, and Signal Processing*, 1988.