Evaluating the Effectiveness of Preprocessing Methods on Motor Classification Scores in EEG Data

Connor Browne

A thesis
submitted in partial fulfillment of the
requirements for the degree of

Master of Science

University of Washington

2023

Committee:

Erika Parsons

William Erdly

Pierre Mourad

Peng Du

Program Authorized to Offer Degree:
Computer Science and Software Engineering

University of Washington

Abstract

Evaluating the Effectiveness of Preprocessing Methods on Motor Classification Scores in EEG Data

Connor Browne

Chair of the Supervisory Committee:
Erika Parsons
School of Science, Technology, Engineering & Mathematics

Classification of motor tasks is of significant interest in brain-computer interfacing today. Electroencephalograph data contains a large amount of noise obfuscating the signal associated with these motor tasks. Various preprocessing techniques exist to increase the signal-to-noise ratio allowing for more accurate classifications. The effectiveness of these techniques varies between motor tasks and in different environments. There is a need to evaluate these different techniques in many different environments and with different motor tasks. This thesis investigates the effectiveness of several preprocessing techniques and classification models for classifying four different motor imagery tasks from EEG data. Specifically, Frequency Filtering, ICA, and CSP are evaluated using Naive Bayes, kNN, Linear SVM, RBF SVM, LDA, Random Forest, and a MLP Neural Network. To control for the environment, data was collected from student volunteers in short sessions designed to demonstrate either eye blinking, eye rolling, jaw clenching, or neck turning. Each task had its own procedure for the session. Motor tasks in data were evaluated for frequency and amplitude commonalities using continuous wavelet transforms and Fourier transforms. Preprocessing Techniques were then iteratively applied to these datasets and evaluated using an ML model. The evaluation metrics used were Accuracy, F1, Precision, and Recall. Results showed that the combination of Frequency Filtering, ICA, and CSP with the Naive Bayes and Random Forest models

yielded the highest accuracy and F1 for all motor tasks. These findings contribute to the field of EEG signal processing and could have potential applications in the development of brain-computer interfaces. It also directly contributes to a greater project in spatial neglect rehabilitation by providing novel insights to common artifacts in EEG data, as well as to the creation of a framework for data processing in real-time and offline.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I would like to express sincere appreciation to Prof. William Erdly, Prof. Pierre Mourad, and Prof. Peng Du for accepting my request to be on the committee for this thesis and for their valuable feedback. Finally, I offer my enduring thanks to Prof. Erika Parsons for accepting me into the research team and without whose support and guidance, this research would not have been possible.

Chapter 1

INTRODUCTION

1.1 Background

Electroencephalography (EEG) is a non-invasive technique used to record the electrical activity of the brain. EEG data is used in a wide range of applications, including clinical diagnosis, cognitive neuroscience, and brain-computer interfaces. One important application of EEG data is in the classification of motor tasks, such as eye blinking, eye rolling, neck turning, and jaw clenching.

Other techniques for capturing motor task information exist such as electromyography (EMG), but they have distinct characteristics from EEG and do not capture its cognitive aspect while still being accessible. EEG does not require any surgical procedures or insertion of electrodes into the muscles, making it more comfortable and less invasive for the participants. EEG captures the collective electrical activity of large populations of neurons in the brain. It provides a comprehensive view of brain dynamics and allows for the analysis of brain processes related to motor tasks beyond the specific muscle groups being activated. This broader perspective can be valuable for studying complex motor processes, such as motor planning, coordination, and cognitive aspects of motor tasks. While EMG provides direct information about muscle contractions and force exertion, EEG provides insights into the neural processes and cognitive aspects underlying motor control. Overall EEG provides the greatest use-case for this research and the most avenues for further development.

EEG can be used to capture information about brain pathology, including spatial neglect, through the analysis of neural activity patterns. EEG can be used to measure event-related potentials (ERP), which are changes in electrical activity in the brain in response to specific events or stimuli. By comparing ERPs between individuals with spatial neglect and healthy

controls, researchers can identify differences in neural processing associated with spatial attention and awareness. EEG allows for the analysis of different frequency bands of brain oscillations, such as alpha, beta, theta, and gamma brain waves. Spatial neglect has been associated with abnormal patterns of oscillatory activity, particularly in the parietal and frontal regions of the brain. EEG can be used to examine functional connectivity between different brain regions. Spatial neglect is believed to involve disruptions in the functional connectivity networks responsible for spatial attention and awareness. EEG-based neurofeedback training can be used as a therapeutic approach for individuals with spatial neglect. By providing real-time feedback of their own brain activity, individuals can learn to self-regulate their brain states and improve spatial attention and awareness. While this work does not examine brain pathology directly it may be used later to provide information on preprocessing techniques and healthy controls.

Classification of motor tasks from EEG data has been the subject of extensive research in recent years. Various preprocessing techniques, such as frequency filtering (FF), independent component analysis (ICA), and common spatial pattern (CSP) have been employed to increase the signal-to-noise ratio (SNR) of EEG data and improve the classification accuracy. EEG data contains a significant amount of noise, including EMG contamination caused by the propagation of muscle signals into EEG data. Reducing this noise is of significant interest to brain computer interfacing (BCI).

Moreover, different classification models, including naive bayes, k-nearest neighbors (kNN), linear support vector machines (SVM), radial basis function (RBF) SVM, linear discriminant analysis (LDA), random forests, and multi-layer perceptron (MLP) neural networks, have been used to classify motor tasks from EEG data.

Evaluation metrics such as accuracy, F1 score, precision, and recall have been used to assess the performance of these classification models.

In this work, we aim to investigate the effectiveness of various preprocessing techniques, such as FF, ICA, and CSP, on the classification scores of motor tasks in EEG data, specifically eye blinking, eye rolling, neck turning, and jaw clenching. We also aim to compare the

performance of different classification models, including naive bayes, kNN, linear SVM, RBF SVM, LDA, random forest, and MLP neural network, using these preprocessing techniques.

1.2 Research Problem

EEG signals can be used to identify patterns of neural activity associated with various cognitive and motor tasks, making it a valuable tool in neuroscience and brain-computer interface (BCI) research. However, EEG signals are often contaminated by various sources of noise, such as muscle artifacts, eye movements, and environmental interference. These artifacts can significantly reduce the signal-to-noise ratio (SNR) of the EEG data and make it difficult to accurately classify brain states or control a BCI system. Below is a sample of EEG data demonstrating its 7 channels, waveform, and associated noise.

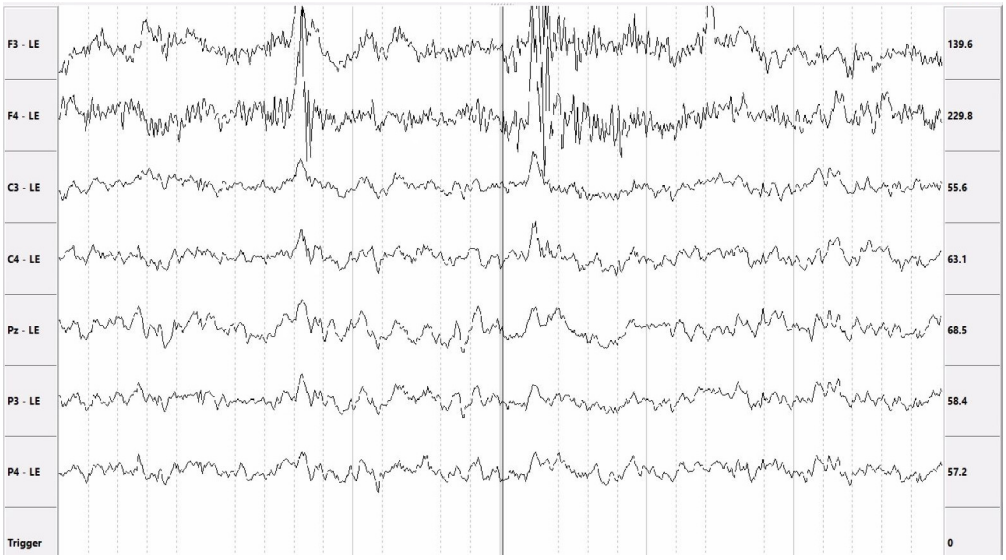

Figure 1.1: Sample EEG data with seven channels

Various preprocessing techniques, such as frequency filtering (FF), independent component analysis (ICA), and common spatial patterns (CSP), have been proposed to mitigate the effects of these artifacts and improve the SNR of EEG data. However, the effectiveness of these preprocessing techniques varies depending on the specific task and type of artifact.

Their effectiveness can also vary with the changing hardware and how they are fitted. Below is a sample of jaw clench data transformed with ICA showing how noisy signals can be separated onto independent components.

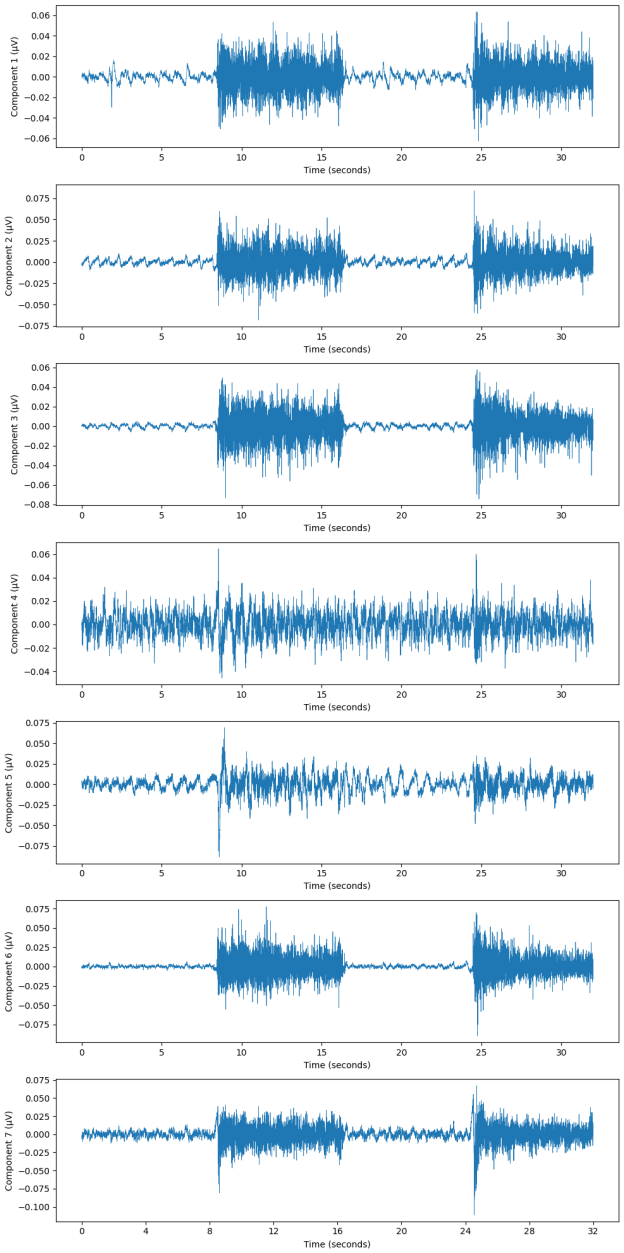

Figure 1.2: Sample Jaw Clench data transformed with ICA

Therefore, the research problem addressed in this thesis is to evaluate the effectiveness of various preprocessing techniques, including FF, ICA, and CSP, in improving the SNR of EEG data and the scores of classifying different motor tasks, specifically eye blinking, eye rolling, neck turning, and jaw clenching. The research aims to provide a better understanding of the optimal preprocessing pipeline for improving EEG data quality and accuracy in BCI applications.

1.3 Objectives

The main objective of this study is to evaluate the effectiveness of various preprocessing techniques and classification models for classifying motor tasks from EEG data. Specifically, the study aims to:

- Design data collection experiments to structure collected data for each selected motor task.
- Collect EEG data from student volunteers while they perform motor tasks of jaw clenching, eye blinking, eye rolling, and neck turning.
- Preprocess the collected EEG data using the FF, ICA, and CSP techniques.
- Train and test the naive bayes, kNN, linear SVM, RBF SVM, LDA, random forest, and a multi-layer perceptron neural network models on the preprocessed data to classify the motor tasks.
- Evaluate the performance of each classification model using the accuracy, F1, precision, and recall metrics.
- Compare the performance of the different preprocessing techniques and classification models to determine the most effective combination for classifying motor tasks from EEG data.

- Determine the statistical power of each comparison to accept or reject the null hypothesis.

The ultimate goal of this study is to contribute to the field of EEG signal processing and provide insights into the development of brain-computer interfaces. This work expands knowledge of preprocessing techniques and their effectiveness of increasing SNR. It widens tested motor tasks to include less common tasks often stripped out as noise. It potentially provides additional structured public data for motor evoked potentials. This work directly contributes to an umbrella project in Brain-Computer Interfacing that seeks to rehabilitate spatial neglect patients following strokes. This work seeks to contribute to the spatial neglect project by detecting high-level artifacts and brain-data that will later contribute to the elimination of noise, testing the selected preprocessing techniques for their viability in filtering for noise in the project, and creating a framework for analyzing artifacts in EEG data.

1.4 Scope and Limitations

The purpose of this experiment is to evaluate the effectiveness of three preprocessing techniques (frequency filtering, independent component analysis, and common spatial patterns) and seven machine learning models (naive bayes, kNN, linear SVM, RBF SVM, LDA, random forest, and multi-layer perceptron neural network) on classification scores. The evaluation will be done only with the selected metrics. It does not attempt to evaluate other preprocessing techniques, nor does it seek to evaluate other machine learning models.

Data for the experiments is collected from University of Washington Bothell student volunteers, this potentially influences generalizability to other groups. The experiments do not perform hyperparameter tuning for either preprocessing technique configurations or machine learning models either over all tasks, or for each singular task. These limitations primarily effect external validity and the generalizability of these findings to other environments.

Chapter 2

RELATED WORK

In this chapter we discuss the existing work related to motor classification and brain-computer interfacing (BCI) that supports our research and thesis. We present the current state of motor classification and BCI. Finally, we discuss how our work differs from the previous work done in motor classification and BCI.

2.1 Literature Review

Evaluating the effectiveness of preprocessing techniques is a part of motor classification, which falls into the field of brain-computer interfacing (BCI) [16]. BCI is seen as the process of interpreting brain signals into a software output with or without the actual movement of muscles [24]. The field of BCI has seen significant growth in the last three decades [9]. Despite this growth the same experimental design has been utilized with data collection followed by feature analysis, feature extraction, model training, and finally classification of motor tasks. Two issues continue to plague the field of BCI: inter-subject variability, and data noise. Today a variety of preprocessing techniques and machine learning models are utilized to study a wide range of motor tasks and attempt to remediate these issues. The most prominently utilized preprocessing techniques are frequency filtering, independent component analysis, common spatial patterns, and wavelet transforms [24]. Meta-analyses have found these preprocessing techniques to be effective. Studies have also confirmed more channels in the EEG system leads to greater results in classification [24].

Recently significant research has been conducted on deep learning methods of classification and feature extraction for motor tasks [19]. Though many opportunities in BCI research remain. One such opportunity is the use of BCI to assist in rehabilitation of movement dis-

orders in stroke patients [25]. Publicly available datasets are therefore of great importance to researchers and the quality of those datasets are of paramount importance. Many available datasets in BCI have been on arm and hand related movement [8]. Due to the need for stability in data, a common form of dataset is needed. This is not a requirement that has been met for every form of motor task. Despite the growth of research and availability of data, the classification rate of motor tasks has not effectively grown [24], nor has the use of publicly available data [8].

2.2 Related Work

Other works have validated the the effectiveness frequency filtering (FF), independent component analysis (ICA), and common spatial patterns (CSP). One such study evaluated the effectiveness of band-pass filtering and ICA on classification of arm and foot motor imagery (MI) tasks. Their findings were that frequency filtering and combinations of ICA were effective in increasing signal to noise ratio [7]. A different study tested the effectiveness of preprocessing techniques on classifying the act of mentally swimming. They found that the effectiveness of preprocessing in increasing classification rate varied by the type of machine learning models, and for SVM models there was statistically no benefit at all. [18]. Yet another study evaluated the effectiveness of all three preprocessing techniques on classification of foot,hand, and tongue motor tasks [4]. They found the combination of all three techniques was the most effective.

Prior Work The work of evaluating preprocessing techniques and classifying motor tasks began earlier with work in detecting and classifying jaw clenches in raw EEG data [6]. This work focused real-time classification using blocked data and a continuous Fourier transform (CFT) to evaluate an amplitude threshold on low frequency data in the C3 channel. Groundwork for the data collection experiments, as well as the data processing pipeline were made here. A weakness in this work was its singular focus on jaw clench data that was not adaptable to other data types without manual control over artifact rejection.

Spatial Neglect Rehabilitation This study takes place under the auspices of the spatial neglect rehabilitation research project. Spatial neglect is a neuropsychological condition affecting up to 82% of stroke patients [3]. Spatial neglect manifests as an "inability to perceive, report and orient to sensory events towards one side of space" [11]. Rehabilitation from spatial neglect is a tedious and not-well-understood process with unreliable results. The goal of the project is the development of a robotic tool to probe and aid patient rehabilitation efforts. For this purpose the robot utilizes edge computing in the form of a base-station to monitor patient EEG and physiological readings. This allows the robot to understand when the patient comprehends its presence, and when they do not. Figure 2.1 shows the overall design for the project. To process the EEG data requires an applied machine learning approach that aims to classify both brain-wave related events and event-related potentials in the form of eye movement. Potential candidates for important brain waves are alpha and beta waves. Further physiological data may be incorporated into the project over time, which will necessitate evaluation of the data against the EEG readings for corollary purposes.

The pipeline for classification of motor tasks for the spatial neglect rehabilitation project is an open question with many solutions and few concrete answers. This study aims to address several base questions of the project, such as detecting any specific event, and to aid it in reaching a state that the EEG ML solution could be applied to actual spatial neglect patients. Further questions in the project surround the comparability of students to spatial neglect patients in terms of EEG readings, and how detectable the various manifestations of the condition are in the specific EEG headset model being utilized. The project is a multi-disciplinary team spanning mechanical engineering to neurology with multiple teams of people focusing on different aspects of the project. This study will collaborate with the edge computing and data analytic portion of the greater project with further possible cross-over.

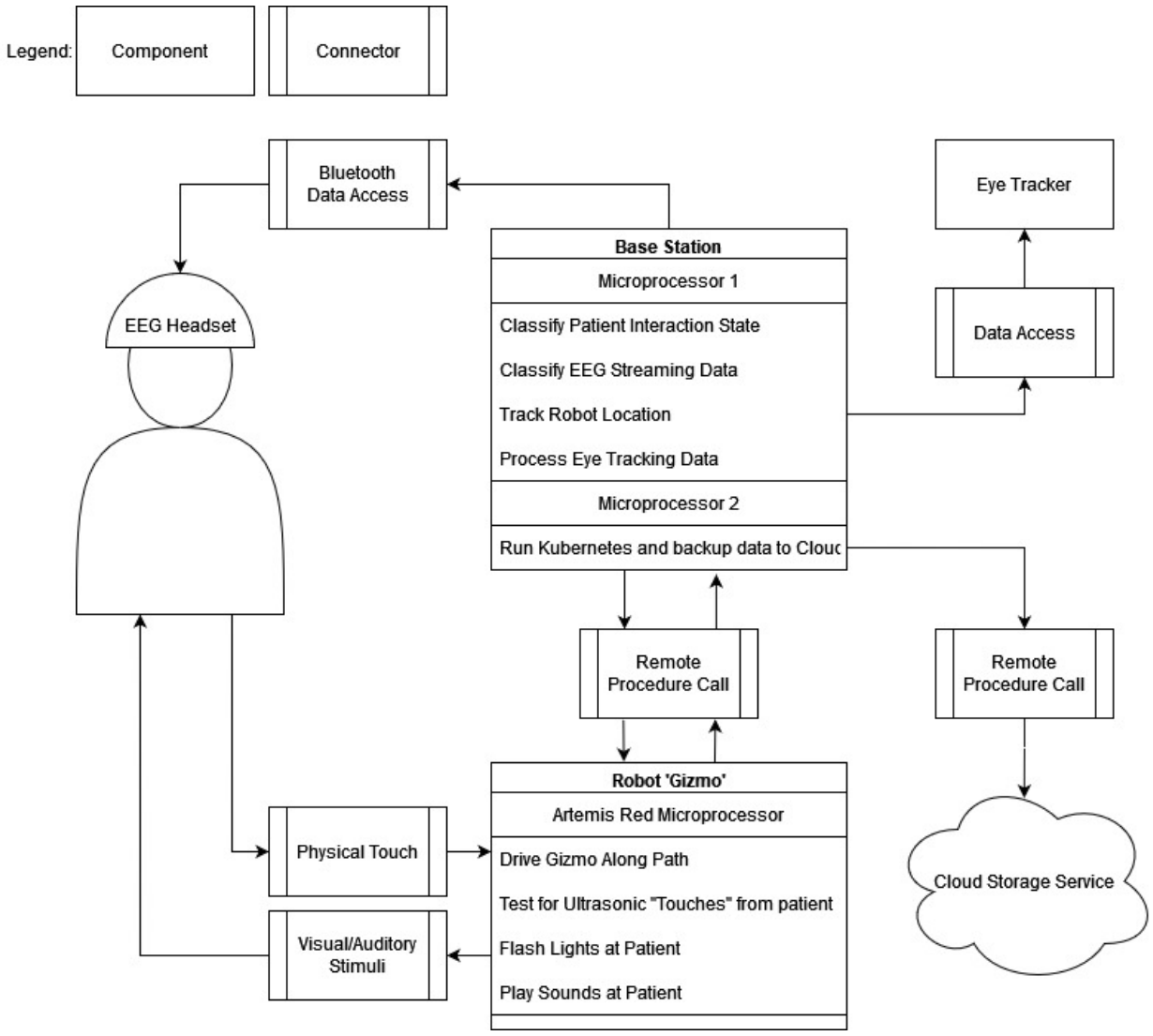

Figure 2.1: Spatial Neglect Rehabilitation Program Design Diagram in Component Connector Format

Chapter 3
TECHNICAL BACKGROUND

In this chapter we discuss the technical background for this work related to motor classification, brain-computer interfacing (BCI), statistics, and machine learning.

3.1 Brain-Computer Interface Background

For a century neuroscience has been researching and discovering how the human mind interprets and reacts to signals and motor actions. Signals from the brain effecting the EEG waveform broadly fall into two categories, brain waves generated by cyclical neuron activation, and event-related potentials [10]. Brain waves primarily consist of alpha, beta, theta, and delta waves, with less common waves being gamma and mu waves. Figure 3.1 below demonstrates some examples of common brain waves. These waves are primarily measured by frequency and amplitude. Further, each wave is associated with a different mental and physical state, for example alpha waves are associated with a restful state with little mental activity, while beta waves are associated with intense activity or concentration.

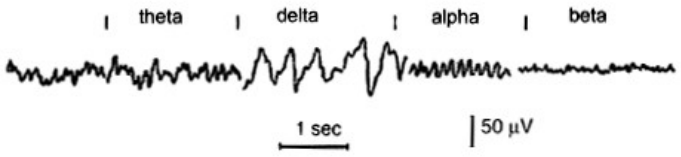

Figure 3.1: Examples of common brain waves in EEG data [21]

Event-related potentials originate from either sensory-evoked potentials (SEP) or motor-evoked potentials (MEP) and can vary in intensity and temporal delay from its triggering

event. Some examples of ERPs can be a light flashing, a pinch, an eye blink, or a jaw clench. Various brain-health pathology, such as epilepsy, can significantly alter an EEG waveform. Further, brain-waves can vary with age as certain wave types are more prevalent in children compared to adults and visa-versa. Figure 3.2 below demonstrates the mapping of senses and motor actions to the brain cortex. It is a cortical homunculi, also known as sensory or motor homunculi, a visual representation of the human body mapped onto the brain's cortex. It depicts the distorted proportions of body parts based on their representation in the somatosensory and somatomotor areas of the brain.

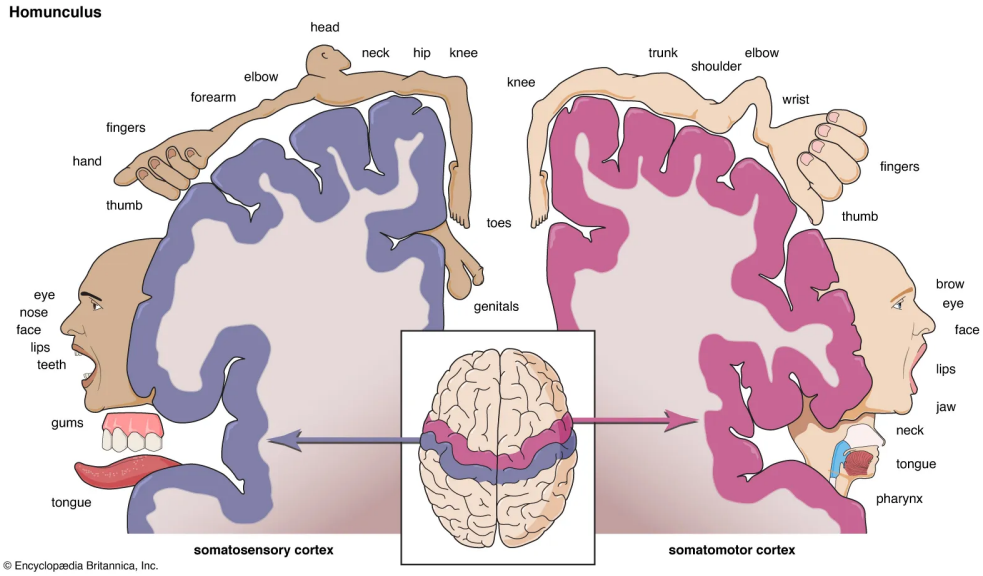

Figure 3.2: Cortical Homunculus Map from [15]

Today the most common method for reading this brain state is Electroencephalography (EEG) An EEG device reads cortical brain activity in the form of electrical potentials from synapses firing on the surface of the brain [13]. The data from an EEG are recorded as a waveform in micro-volts and time in an electroencephalogram. An EEG device does this through the use of electrodes located on an individual's head such as in figure 4.2 on page 26. These electrodes may be dry, wet, sub-dermal, or cortical in placement and type. The

placement of these electrodes is standardized to allow any trained neurologist to interpret the EEG data. Given all these sources of information the EEG waveform is a complex composite of signals in healthy adults. The only time an EEG will read straight, is in brain-death.

It is important to distinguish muscle signals from MEPs. A muscle signal typically refers to the electrical activity recorded directly from muscles using electromyography (EMG). EMG measures the electrical signals generated by muscle fibers during contraction and relaxation. Muscle signals provide information about the activation and behavior of the muscles themselves, reflecting muscle activity, force production, and motor unit recruitment. When muscles are active, there is a phenomenon called EMG contamination, where the electrical activity generated by the muscles can interfere with the EEG signal. During muscle contractions or movements, the electrical activity of the muscles can produce artifacts that are picked up by the EEG electrodes. These muscle artifacts can contaminate the EEG signal and make it challenging to isolate and analyze the pure brain activity. In this work, we are analyzing EMG contamination in the EEG signal. This does not mean cortical changes associated with muscle activation are not present in the EEG data, but they are being covered up by the muscle signals.

The process of collecting data in motor brain-computer interfacing typically follows a paradigm of short sessions where a volunteer receives instructions through the session. These instructions typically follow a pattern with a leadup to action (fixation cross), a cue, motor block, then rest gap as demonstrated in figure 3.3 below. These instructions cycle throughout the session [4, 8]. The type of cue, and how it is delivered varies significantly. Following the motor action, the brain state does not immediately return to a resting state and requires a period to reset.

Frequency Filtering

Frequency filtering is a widely used technique in motor task preprocessing that aims to extract specific frequency ranges of interest from the recorded signals. It is used to attenuate

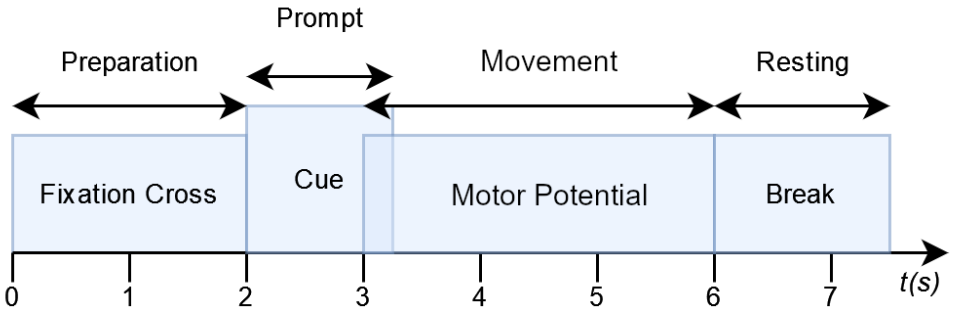

Figure 3.3: Cycle of Motor Potential

or enhance specific frequency bands. Frequency filtering helps to enhance the signal-to-noise ratio and highlight relevant neural activity by selectively filtering out noise or unwanted frequency ranges. Different types of filters, such as high-pass, low-pass, band-pass, or notch filters, can be employed based on the specific research objectives and characteristics of the EEG data. The most popular form of frequency filtering is band-pass filtering during pre-processing [24]. In these instances it allows researchers to focus on specific frequency ranges associated with different brain wave activities, such as alpha, beta, theta, or gamma waves, enabling a more targeted analysis of brain dynamics and functional connectivity.

Independent Component Analysis

Independent component analysis (ICA) is a powerful signal processing technique used in EEG data analysis to attempt to divide mixed signals into their underlying independent components. As EEG signals are a mix of neural activity, physiological artifacts, and environmental noise, ICA can effectively separate these signals by assuming that the components are statistically independent. Decomposing the signals into independent components has allowed other studies to identify and isolate neural signals of interest, such as brain activity related to specific cognitive processes or motor tasks. This technique has proven valuable in removing artifacts, such as eye blinks, muscle activity, and electrooculography (EOG) artifacts, from the EEG data, thereby enhancing the quality and interpretability of the signals

[5].

Common Spatial Patterns

Common spatial patterns (CSP) is a preprocessing method widely used in BCI in EEG data analysis that tries to enhance the discriminative power of motor task signals for specific tasks or conditions. The goal of CSP is to find spatial filters that maximize the differences in power or activity between two classes of EEG signals, such as motor tasks. By applying CSP, researchers can identify spatial patterns of brain activity that are most relevant to the desired task. The technique works by computing a set of spatial filters that transform and project the EEG signals to highlight the most discriminative brain regions or patterns. It is worth noting that 8 channels is considered the minimum for using CSP [24]. Generally, it allows researchers to enhance the signal-to-noise ratio, improve the classification accuracy, and extract meaningful features from EEG data. This enables a deeper understanding of the underlying brain dynamics and facilitates the development of more effective neurofeedback and brain-controlled systems [4].

Nyquist Frequency

The act of interpreting a complex waveform, and breaking it down into constituent frequencies is one of sampling and testing for each frequency at each time-step in the signal. The Nyquist frequency is a fundamental concept in signal processing that determines the maximum frequency that can be accurately represented in a digital signal [1]. According to the Nyquist-Shannon sampling theorem, the sampling rate must be at least twice the highest frequency of interest in the signal to avoid aliasing and loss of information. If the Nyquist frequency is not satisfied, higher frequencies can fold back and appear as lower frequencies, leading to distortion and inaccurate representation of the original signal.

When detecting frequencies in a signal, it is essential to consider the Nyquist frequency in relation to the sample rate. If the frequency of interest exceeds half of the sample rate, aliasing can occur, and the frequency will be misrepresented or undetectable. Therefore, to

accurately capture higher frequencies, a higher sample rate is required to ensure that the Nyquist frequency is appropriately addressed.

3.2 Statistical and Machine Learning Background

Basic Statistical Measures

In statistics there are several important measures that describe groups of data. These measures are the mean, variance, and standard deviation of data. While there are more basic measures, they are not utilized in this work. These values describe the centrality and spread of groups of data and are defined below.

Mean: The sum of a set of numbers divided by the count of numbers. Known also as the arithmetic mean or arithmetic average it is the most commonly utilized measure of central tendency for a set of numbers.

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i = \frac{x_1 + x_2 + ... + x_n}{n}[2] \tag{3.1}$$

Variance: A measure of dispersion for a set of numbers, variance helps describe the relative positions of values. It is also known as the standard deviation squared.

$$\mathrm{Var}(X) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2 \tag{3.2}$$

Standard Deviation: Another measure of dispersion for a set of numbers, standard deviation describes the spread of positions from a central value. It is commonly used as it is in the same units as the original values, is sensitive to outliers in the data set, and is consistent to other statistical measures.

$$\mathrm{SD}(X) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{3.3}$$

Sum of Squares

The sum of squares is used to measure different aspects of the variance of a set of data points. It is a vital technique for measuring the effect an independent variable upon the dependent variable. Two forms of the sum of squares are utilized in this work in the formula of partial eta squared: the total sum of squares, and the sum of squares of the effect of interest. The total sum of squares equation below calculates the sum of the squared differences between each observation and the mean, this represents the total variation in the data. In equation 3.4 n is the total number of samples in the data set, x_i is a single data point, and $\bar{x}$ is the mean value of the data set.

$$SS_{\text{total}} = \sum_{i=1}^{n} (x_i - \bar{x})^2 \tag{3.4}$$

Unlike the total sum of squares, the sum of squares of the effect of interest measures the amount variance that is attributable to the influence of a specific effect or variable upon a set of data points. Where each group represents a different configuration of the independent variable, it calculates the sum of the squared differences between each group mean and the overall mean, representing the variation specifically attributable to the effect of interest. In equation 3.5 k is the number of groups, n_i is the number of samples in a specific group, $\bar{x}_i$ is the mean value of that group, and $\bar{x}$ is the mean of the entire data set.

$$SS_{\text{effect}} = \sum_{i=1}^{k} n_i (\bar{x}_i - \bar{x})^2 \tag{3.5}$$

Partial Eta Squared

Quantifying the change in a variable caused by an effect of interest is of significant interest to empirical research and education [17]. A growing number of journals require that effect sizes are reported for data. An effect in this case is the change caused by variation of treatments in an experimental design. Specifically for this work it is the variation of preprocessing techniques and machine learning models separately. The size of an effect is a measure of the proportion of change in a dependent variable that is attributable to the independent

variable. Equation 3.6 defines partial eta squared and uses the definitions defined above in equations 3.4 and 3.5. The use of the sum of squares accounts for variance that is not attributable to the effect of interest by taking the variance in the overall data set in addition to the variance caused by the effect. Partial eta squared is commonly utilized with Analysis of Variance (ANOVA). As this work utilizes one-way ANOVA and Tukey HSD the reported effect size will be partial eta squared. In these methods partial eta squared is the same as eta squared and generalized eta-squared [22].

$$\eta_p^2 = \frac{SS_{\text{effect}}}{SS_{\text{total}}}, 0 \leq \eta_p^2 \leq 1 \tag{3.6}$$

Statistical Level of Significance

To establish if the cause and effect of an experiment covary the most commonly used method is null hypothesis significance testing (NHST) [20]. In experimental design, the null hypothesis states that the difference between the population means of the samples before and after effect are zero. Often this results in either the null hypothesis being true or the experimental hypothesis being true. To establish if the post-effect groups are significantly different, an experimenter may utilize a t-test and denote the probability that the difference obtained would have occurred by chance where no actual between-group difference exists. This has often become known as the statistical level of significance set at $p < 0.05$, therefore known as the chance that no actual difference exists. This concept is shown in figure 3.4.

The usage of the level of significance as being "significant" or "not significant" does have consequences as it implies the effect size is entirely zero or not instead of existing on a scale. To address this, effect sizes such as partial eta squared are reported to describe the proportion of variance of the sample means attributable to the effect. The most common of these significance tests is the t-test which is designed to evaluate how distinguishable the two different sample groups of data are. Most commonly setup as before and after effect (or control and treatment groups.) In this work though, there are 7 different preprocessing configurations and 7 different machine learning models, this would require 14 different t-

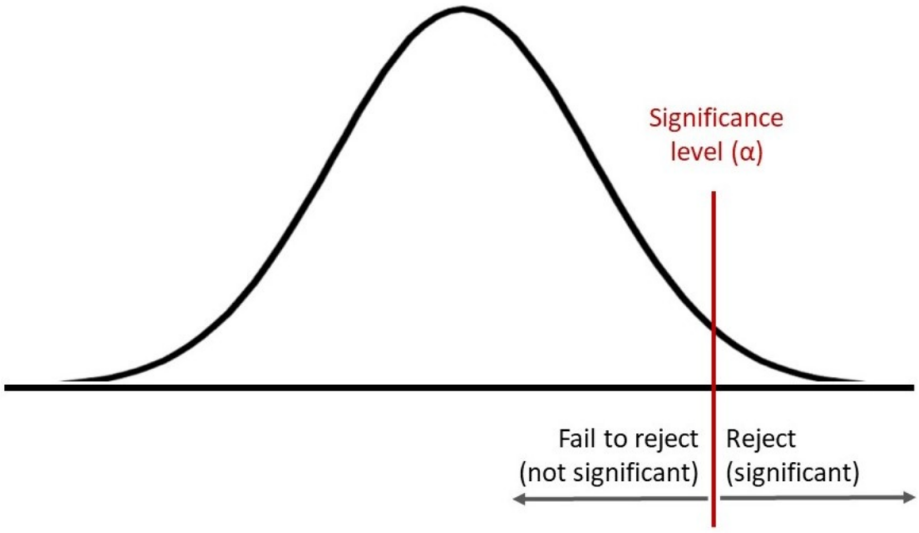

Figure 3.4: A depiction of the use of significance level α to address the null hypothesis. In this example, α is the area under the curve of the normal distribution to the right of where it is. A common α value of 0.05 means that 5% of the represented data is the area under the curve. Thus there is a 5% chance that the sample means are actually the same, and that they appear different due to random chance.

tests for each set. Thus a more nuanced approach is necessary to adequately address the null hypothesis in this work.

Analysis of Variance

To evaluate level of significance and reject the null hypothesis where multiple independent groups exist would require a large number of t-tests, therefore this work utilizes a two stage process starting with the analysis of variance (ANOVA) test. ANOVA seeks to evaluate statistical power on all groups simultaneously and thus describe how closely grouped all of the groups are. It accomplishes this using sum of squares to calculate variances across groups. Breaking down the sum of squares components into totals, effects, and errors allows ANOVA to generate a F-statistic value and generate a p-value for all the groups based upon that value. The equations for the sum of squares of error (also known as the residual sum of squares) and the F-statistic are below. In equation 3.7 k is the number of groups, n_i is the number of observations in the $i-th$ group, Y_{ij} is the $j-th$ observation in the $i-th$ group, and $\bar{Y}_i$ is the mean of the $i-th$ group. After calculating the F-statistic it is used to calculate the p-value using a F-distribution with $r-1, n_t-1$ degrees of freedom. Utilizing ANOVA allows for the evaluation of the significance of overall results without the use of numerous t-tests while still being mathematically rigorous. This step of the two-stage process does not evaluate the significance of differences between individual groups however, requiring the second stage in the process.

$$SS_{\text{error}} = \sum_{i=1}^{k} \sum_{j=1}^{n_i} (Y_{ij} - \bar{Y}_i)^2 \tag{3.7}$$

$$F^* = \frac{MS_{\text{between}}}{MS_{\text{within}}} = \frac{SS_{\text{effect}}/(r-1)}{SS_{\text{error}}/n_t - r} \tag{3.8}$$

Tukey's Honestly Significant Difference

To complement ANOVA and determine the level of significance for pairwise comparisons between groups, Tukey's Honestly Significance Difference (HSD) is utilized in this work. Tukey HSD calculates the q critical value using the studentized range distribution, a derivation of the t-distribution. It does this for each pair of groups being compared. The exact form used in this work utilizes equations 3.9 and 3.10. MS_w is the mean sum of squares of the error. The p-value for each pairwise comparison is approximated using the studentized range distribution as well. These approximations are known to be conservative if samples sizes do not match between groups. Tukey's HSD test controls the family-wise error rate, ensuring that the overall probability of making a Type I error is kept at the desired level, even with multiple comparisons.

$$t = \frac{x_i - x_j}{\sqrt{2 * MS_w/n}} \tag{3.9}$$

$$Q(\sqrt{2}|t_i|, r, N - r) \tag{3.10}$$

Box and Whisker Plots

Visually displaying the centrality, spread, and important statistical measures is the primary purpose of the box and whiskers plot. It displays information about its central value using the median and variance with the interquartile range as shown in figure 3.5. Based upon the normal distribution, it displays information regarding the lower and upper quartiles of the data, which represent the points where 25% of data lie outside the range. Using this information, the size of the box can be used to show how centrally located the distribution of data is around its median with larger boxes representing greater variance. Additionally, the location of the median within the box displays the skew of the data. The skew represents if more data points fall to the left or right of the median. Outliers in the data are represented as dots outside the whiskers of the plot, they are considered outliers by being so far away

from the center of the distribution. All of these traits are displayed in a single box which can be displayed alongside other boxes. These boxes allow multiple distributions of data and their statistical values to be compared at a glance.

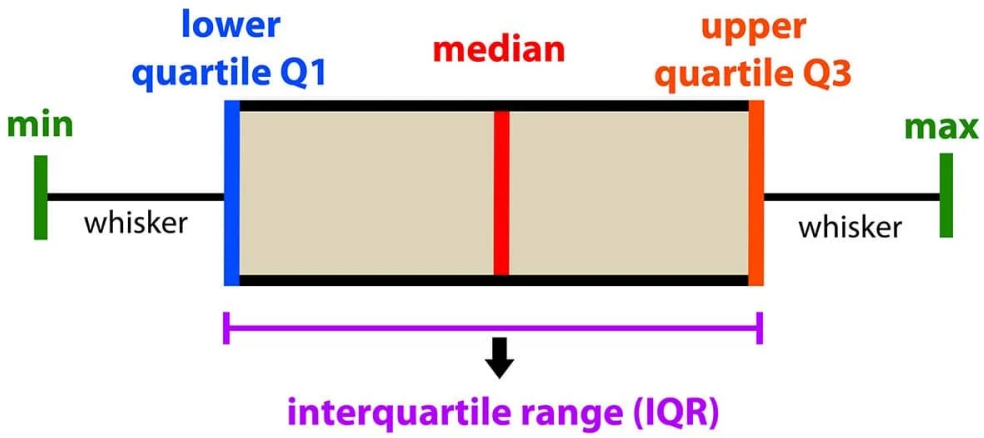

Figure 3.5: The layout of a Box and its Whiskers in a Box and Whiskers Plot. Each labelled part contributes to the relevance of the represented data.

Machine Learning and Decision Boundaries

Machine Learning (ML) is a subset of Artificial Intelligence (AI) and uses algorithms and models to enable computers to learn and make predictions or decisions without explicit programming. It involves training models on data and using statistical techniques to extract patterns, and enable predictions or classifications. ML occurs in two distinct phases, training and testing. During the training phase, a model is created that is based upon the data fed to it. ML falls into several categories known as supervised, unsupervised or semi-supervised. In supervised ML the label of the data being classified is known during training. In unsupervised ML the label is unknown during training. Each ML model generates and utilizes decision boundaries to classify or make decisions based on input data as shown in figure 3.6. These

boundaries are generated during training. The goal of ML is to utilize these models to generate classifications based on the shape of either previously known data or the inherent shape of currently known data.

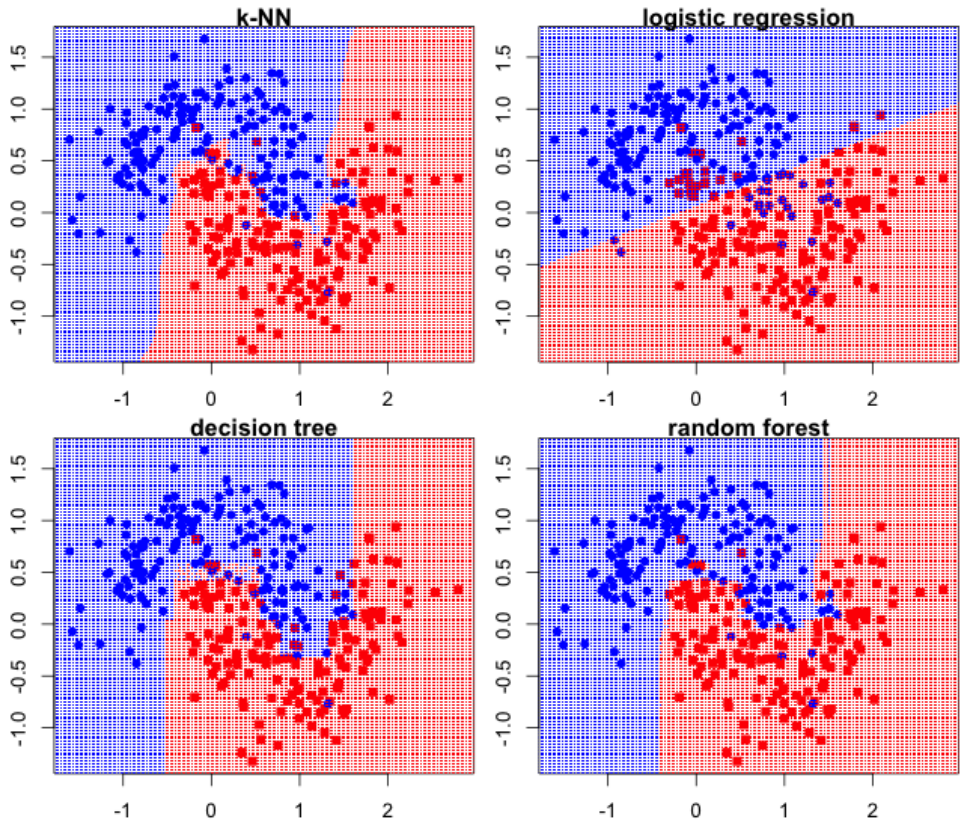

Figure 3.6: Examples of Machine Learning (ML) decision boundaries. ML relies upon the statistical distribution of data to generate a decision boundary separating classes of data from one another. The shape of these boundaries varies model to model, and may be constant, linear, or n-dimensional. These boundaries are created during model training and utilized to generate future classifications or decisions.

Chapter 4

METHODS

4.1 Data Collection

The data for this study was collected from student volunteers at the University of Washington Bothell. All participants provided informed consent prior to the start of the experiment. Data was collected using two programs, the first being DSI Streamer from Wearable Sensing [23], and a custom Python program that orchestrates the session and records data to a .csv. DSI Streamer is used to connect to the EEG headset and collect data wirelessly for each session. It is also used to calibrate each electrode and ensure a solid connection. Finally it acts as a TCP/IP node that the Python program can connect to and collect data from. Panel images and known bugs for DSI Streamer are located in Appendix E. Figure 4.1 shows the layout of each data collection session. The experiment involved recording EEG data while the participants performed four different motor tasks: jaw clenching, eye blinking, eye rolling, and neck turning. Each participant completed multiple sessions each lasting around 56 seconds, during which they were instructed to perform a single task according to a predetermined procedure.

For jaw clenching, participants were instructed to clench their jaw tightly for 8 seconds, followed by a 8-second rest period. This cycle was repeated 3 times for a total of 56 seconds. For eye blinking, participants were instructed to blink their eyes rapidly for 8 seconds, followed by a 8-second rest period. This cycle was repeated 3 times for a total of 56 seconds. For eye rolling, participants were instructed with two different procedures. The first procedure was to roll their eyes upward and around for 8 seconds, followed by a 8-second rest period. This cycle was repeated 3 times for a total of 56 seconds. The second procedure was to roll their eyes every 4 seconds. This cycle was repeated a total of 12 times for a total of

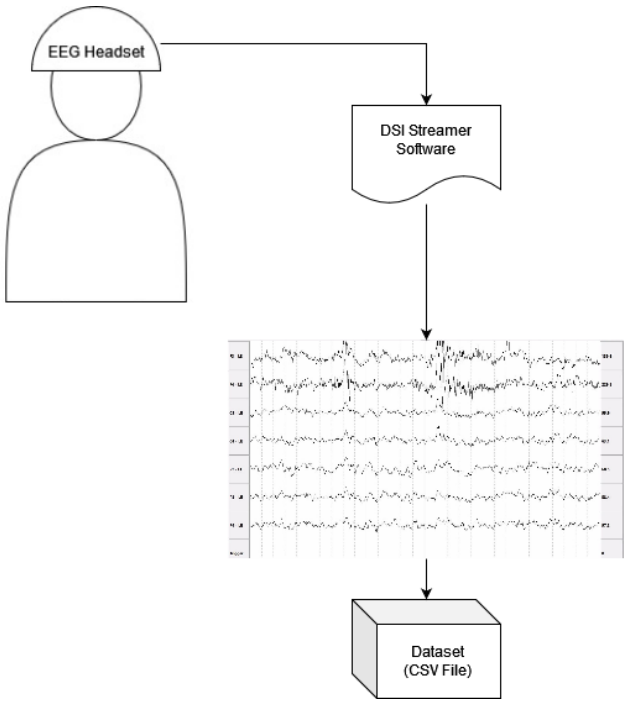

Figure 4.1: Data Collection Diagram. Data is wirelessly transmitted from the EEG headset to DSI Streamer. It is then transmitted locally from DSI Streamer to a Python program that stores the data in a .csv file.

56 seconds. For neck turning, participants were instructed to turn their head to the left and right once every 8 seconds. This cycle was repeated 6 times for a total of 56 seconds. For arm movement, participants were instructed to move their arm horizontally for 8 seconds, followed by a 8-second rest period. This cycle was repeated 3 times for a total of 56 seconds.

During each session, the participant was seated comfortably in a quiet room and was verbally instructed with commands for each task. EEG data was recorded using a 7-channel WearableSensing DSI-7 dry EEG system with a sampling rate of 300 Hz. The electrode layout of the DSI-7 is shown in Figure 4.2 below. The data was saved in a standard comma separated value (CSV) file for further analysis.

To analyze and understand each task type, several methods of preliminary data analysis were performed including plotting raw channel signals, spectral-power plots in windowed slices of data, and scalograms generated with continuous-wavelet transformations (CWT).

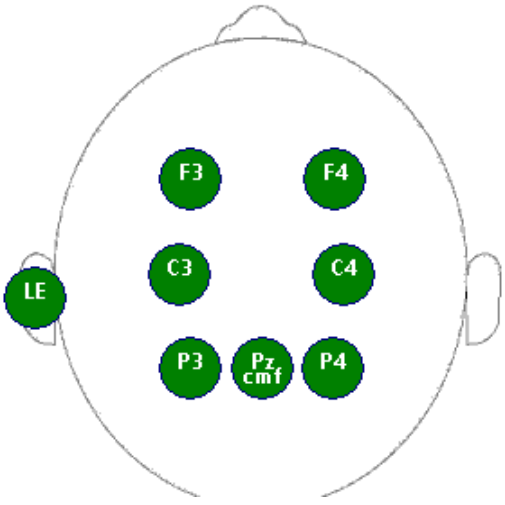

Figure 4.2: DSI-7 Electrode Layout. LE is a reference node that acts as a conglomerate of the other electrodes. It is often removed from classification processing.

4.2 Preprocessing Techniques

The raw EEG data collected during the motor task data sessions was preprocessed using a combination of FF, ICA, and CSP. Figure 4.3 below shows the complete form of the data processing and classification pipeline for this experiment.

First, the raw EEG data was passed through a high-pass filter with a cut-off frequency of 0.5 Hz to remove any baseline drift. The data was then segmented into epochs of 8 seconds duration, each starting at the onset of the motor task and ending as the task ceased. These epochs alternated between positive and negative labels representing the presence or absence of a motor task in the dataset.

Next a combination of either FF, ICA, or CSP was applied to the data. 7 combinations of these techniques were iterated through for each trio of datasets from no preprocessing techniques to all three of them. More details for each of these techniques can be found in chapter 3, section 3.1.

FF was applied to remove any remaining artifacts and noise from the data. This technique involves setting a threshold value for each channel, below which all data is rejected. The threshold values were determined using a visual inspection of the data, and any data below the threshold was removed. For this work the threshold was set at 5 and 100 Hz.

ICA was applied to remove any remaining artifacts and identify independent components associated with the motor tasks. This technique involves decomposing the EEG data into independent components, each representing a specific source of activity.

CSP was applied to enhance the discriminative power of the EEG signals by identifying spatial filters that maximize the differences between the classes. This technique involves calculating the covariance matrix of the EEG signals for each class and then computing spatial filters that maximize the differences between the classes. The spatial filters were applied to the EEG data to create a new feature set for classification. Due to an error in the module providing CSP, principal component analysis (PCA) with 7 components was utilized prior to CSP in the pipeline.

Overall, these preprocessing techniques were chosen to increase the signal-to-noise ratio of the EEG data and improve the classification accuracy of the motor tasks. The preprocessed data was then used for further analysis and classification.

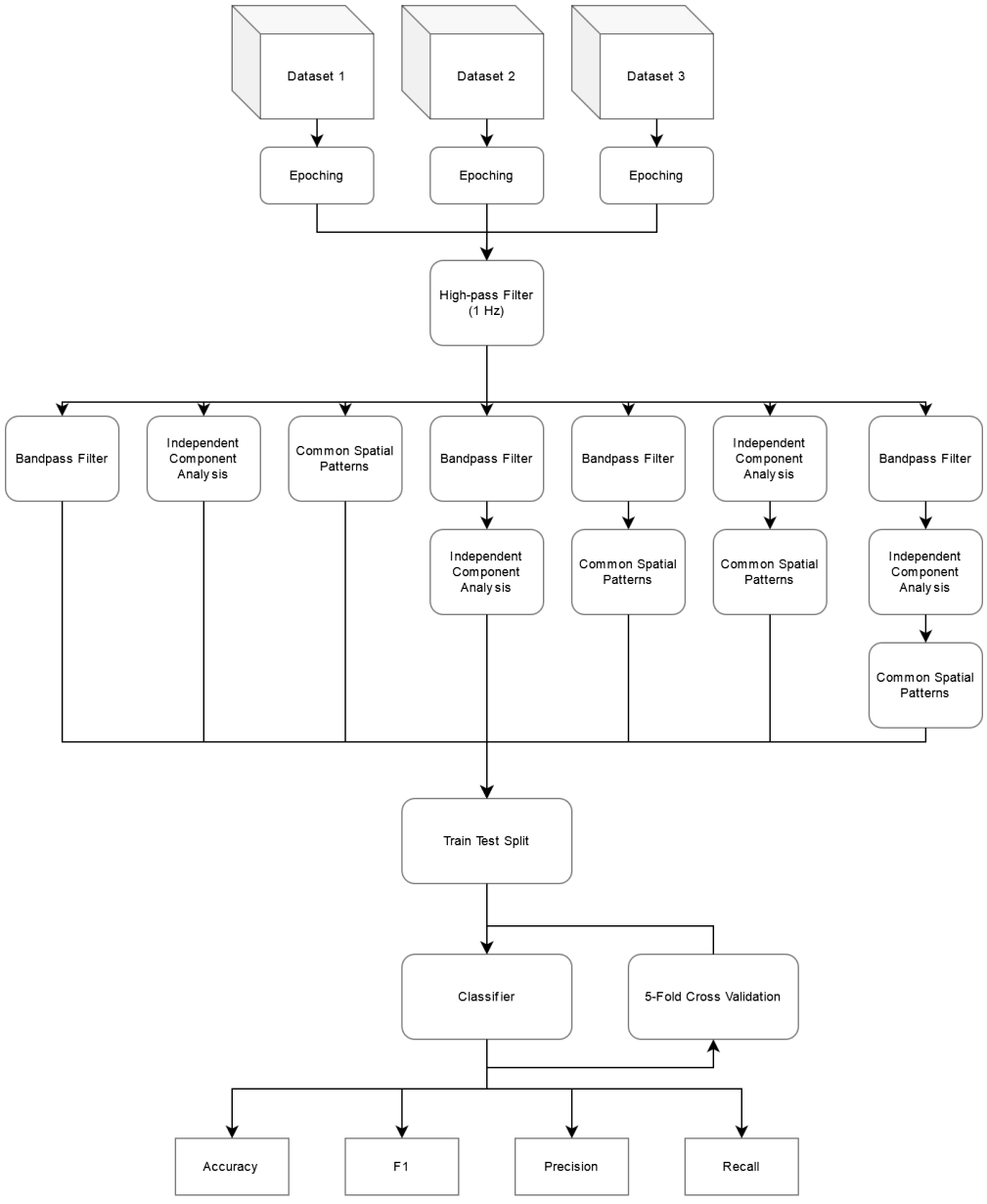

Figure 4.3: Data Processing and Classification Pipeline

4.3 Classification Models

To classify the preprocessed EEG data, seven different models were implemented: Naive Bayes, k-Nearest Neighbors (kNN), linear support vector machine (linear SVM), radial basis function support vector machine (RBF SVM), linear discriminant analysis (LDA), random forest, and a multi-layer perceptron neural network (MLP NN). To control the process a pipeline data structure from SciKit Learn was utilized. Each model utilized 5-fold cross validation in its classification of a trio of data ses to increase generalizability of the model.

Naive Bayes is a probabilistic algorithm that uses Bayes' theorem to classify data. It assumes that the features are independent of each other, which is often not true in practice. However, it is fast and has low memory requirements, making it a popular choice for classification tasks.

k-Nearest Neighbors (kNN) is a non-parametric algorithm that classifies data based on the distance to its nearest neighbors. It works well for small datasets but can be computationally expensive for larger datasets.

Linear SVM is a linear classifier that finds the hyperplane that best separates the classes. It is often used when the data is linearly separable.

RBF SVM is a non-linear classifier that maps the data into a higher-dimensional space using a radial basis function kernel. It is often used when the data is not linearly separable.

Random forest is an ensemble learning method that creates a set of decision trees and combines their predictions to make the final classification. It is often used when dealing with complex, high-dimensional datasets.

MLP NN is a type of neural network that uses multiple layers of nodes to learn complex patterns in the data. It can be used for both classification and regression tasks and has been shown to perform well on a wide range of problems.

The performance of each model was evaluated based on the accuracy, precision, recall, and F1-score metrics.

Overall, these classification models were chosen to compare their performance and de-

termine the most accurate model for classifying the motor tasks based on the preprocessed EEG data.

4.4 Evaluation Metrics

To evaluate the effectiveness of the classification models, we will use several performance metrics including accuracy, F1-score, precision, and recall.

Accuracy: The most commonly used metric that measures the overall correctness of the classification results. It is calculated by dividing the number of correct predictions by the total number of predictions.

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN}$$

Recall: It measures the proportion of true positive predictions among all the actual positive instances in the dataset. High recall indicates that the model is making fewer false negative predictions.

$$Recall = \frac{TP}{TP+FN}$$

Precision: It measures the proportion of true positive predictions among all the positive predictions made by the model. High precision indicates that the model is making fewer false positive predictions.

$$Precision = \frac{TP}{TP+FP}$$

F1 Score: It is a measure of the balance between precision and recall, calculated as the harmonic mean of the two. It provides a single score that considers both precision and recall, which is useful when the classes are imbalanced.

$$F1Score = 2 * \frac{precision * recall}{precision + recall}$$

We will also use confusion matrices to visualize the classification results and to compute other evaluation metrics such as specificity and sensitivity. The evaluation metrics will help us determine the best classification model for the EEG data classification task.

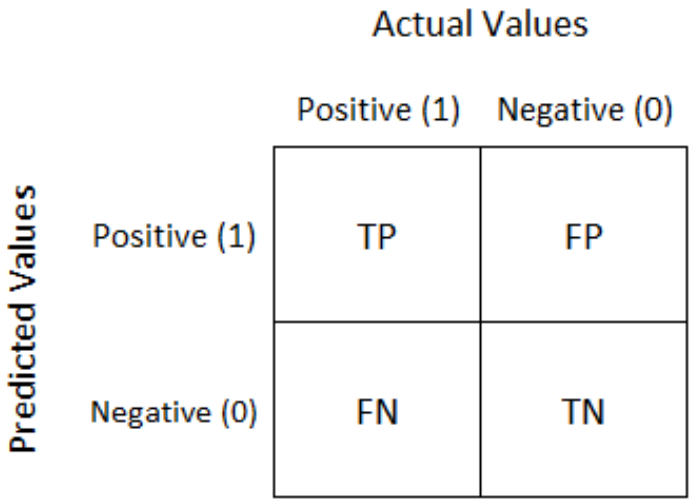

Figure 4.4: Confucian Matrix showing relationship between actual and predicted values in the form of true or false positive and negatives.

4.5 Statistical Analysis

To evaluate the statistical power of the classified data, analysis of variance (ANOVA) and Tukey's Honestly Significant Difference (HSD) were utilized and are defined in Section 3.2.

Figure 4.5 below shows the layout for this analysis after classification. ANOVA was used to evaluate the overall power of the data, while Tukey's HSD was used to evaluate the pairwise power of each combination of the independent variable and its effect on the dependent variable. Effect sizes were reported using partial eta squared (ηp^2). Details for each of these methods can be found in chapter 3, section 3.2. Both the combination of preprocessing techniques and machine learning models was evaluated as well as some specific combinations of both that displayed interesting results.

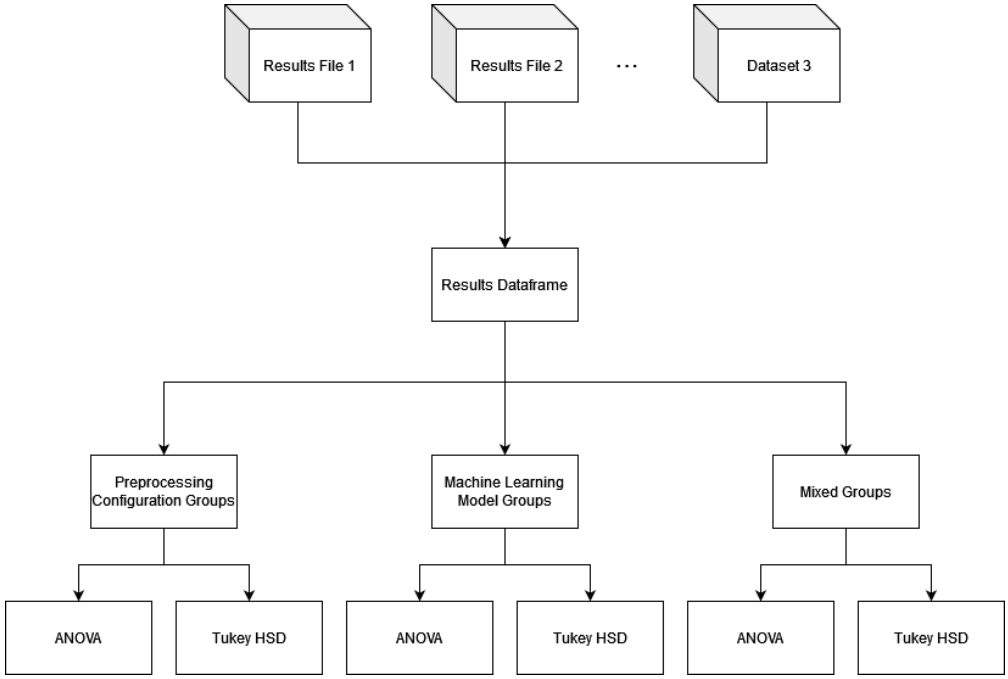

Figure 4.5: Experimental Analysis Layout.

The ANOVA test was used to determine whether there were statistically significant differences between the means of the classification scores across all preprocessing techniques and machine learning models. If a statistically significant difference was found, then Tukey's HSD was performed to determine which specific pairs of techniques and models differed significantly.

Tukey's HSD was used to compare the means of each combination of preprocessing tech-

niques and machine learning models. If the p-value of the Tukey test was below the predetermined alpha level, then it was concluded that there was a significant difference between the means of the compared groups.

The statistical analysis was performed using the SkLearn statistical software package. The significance level for all tests was set at $\alpha = 0.05$.

4.6 Software and Tools

The data analysis and preprocessing were performed using Python 3.8 with the following libraries: NumPy, Pandas, Pingouin, and SciPy. The classification models and pipeline for managing data processing and classification were implemented using the scikit-learn library version 0.24.2.

For data collection, we used the DSI Streamer software from Wearable Sensing to collect EEG data from student volunteers [23]. The software is capable of collecting EEG data from multiple channels simultaneously and transmitting it locally using a TCP/IP Server. Data was then collected and saved in a standardized comma seperated values (csv) file using a python script. This python script was also utilized to time and conduct each data session. All data analysis and preprocessing were conducted in Jupyter Lab version 3.1.10.

Overall, these software and tools provided a robust and efficient framework for data analysis, preprocessing, and classification model implementation, enabling us to achieve the objectives of this research.

Chapter 5

RESULTS

In this chapter, we have discussed in detail the data collected and results based on the methods described in Chapter 4.

5.1 Data Collection Results

Over the course of the experiment the data collection process involved 7 volunteer students who each participated in a data collection session for a motor task using electroencephalography (EEG) equipment. A total of 82 datasets were collected, with approximately 16 datasets per motor type as shown in Table 5.1 below. The data was collected in a controlled environment with consistent procedures and conditions to ensure the validity of the results. Each session did not attempt to control for natural eye blinking through the test. As eye blinks would be present in both positive and negative samples the ML models would treat it as noise. Subjects were asked for any brain-related pathology and that data was recorded. No supplemental data was recorded on physical history, further medical history, or preference for tasks. Further no explicit events were recorded outside of the structure of the data requiring motor tasks to begin and end according to the session's timing.

Overall, the collected datasets provided a substantial amount of data to analyze and evaluate the effectiveness of different preprocessing techniques and machine learning models for motor task classification. The results of the analysis are presented in the following sections.

Table 5.1: Number of Datasets collected per motor task

Motor task	Dataset Count
Jaw clench	12
Eye blinking	21
Eye rolling	15
Neck turning	18
Arm Movement	16

5.2 Raw EEG Plot Analysis

The first stage following data collection was analyzing the motor tasks in a raw EEG format. This was done to visually inspect the data for patterns indicative of the motor tasks. Signals generated by the somatomotor cortex and the muscles are powerful signals and thus were expected to appear in data. Figures 5.1 and 5.2 below are examples of how the brain state reacts over the course of a data collection session. Each task demonstrated patterns that disrupted the resting state of the brain that began and ended with the cues to perform the selected motor task. Given this, it was hypothesized that ML models would successfully be able to distinguish between positive and negative samples if they were properly sliced into epochs. The EEG waveforms also presented differences between channels, with no two channels quite acting alike. Overall, the raw EEG analysis paved the way forward in this work to later analysis and results.

5.3 Task Dependent Analysis

To further analyze each motor task, scalogram plots were utilized. Scalograms are a type of heat-map displaying time, frequency, and power data using the x,y, and hue axes. Comparing the raw EEG plots to scalograms, the x axis (time) remains the same, while the frequencies present at any point of time become the y axis of the scalograms. The amplitude of the signal

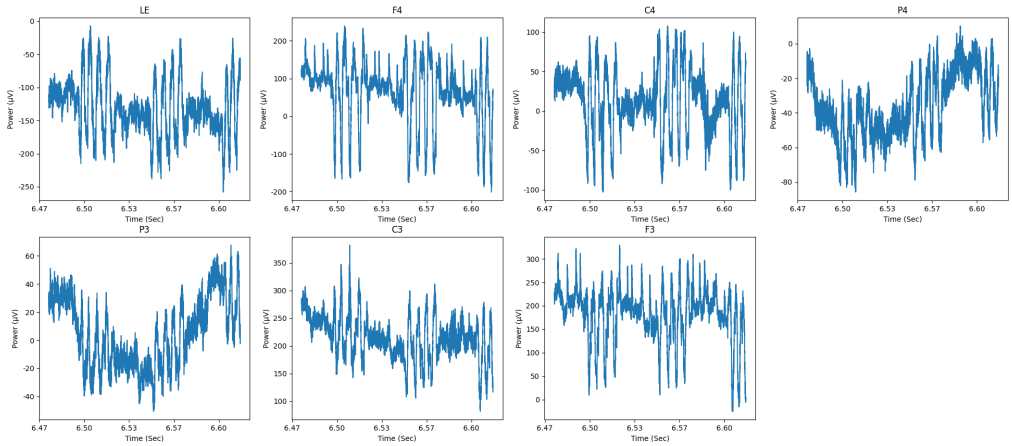

Figure 5.1: Plots of each Channel of Raw EEG Data for sample Eye Roll Data. Periods of greater vertical oscillation are during eye roll "blocks" when subject was asked to repeatedly roll their eyes.

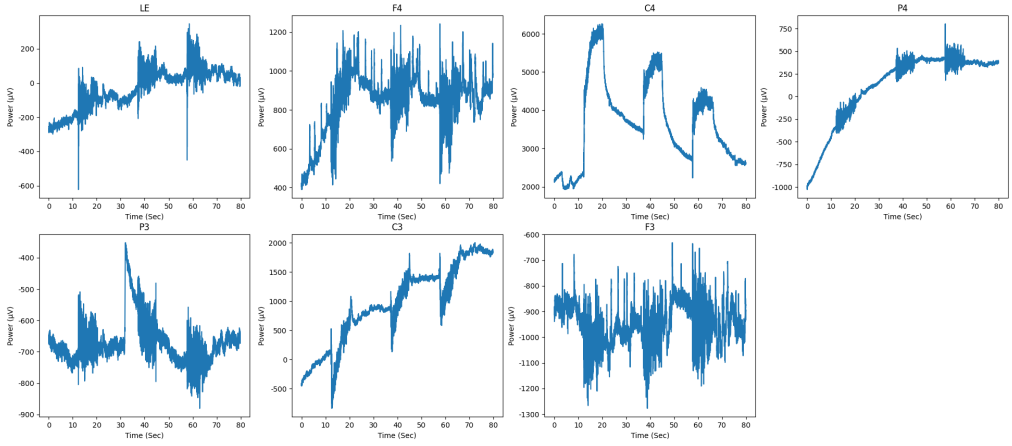

Figure 5.2: Plots of each Channel of Raw EEG Data for sample Jaw Clench Data. Each channel shows different pattern for period where subjects are clenching their jaw. Channel C4 demonstrates the clearest of these patterns, it also demonstrates how the brain state does not return to a resting state immediately.

in the EEG plots become the hue axis of the scalograms. Preliminary scalogram analysis showed that each task displayed a different pattern in time-frequency-amplitude data that was common across all relevant datasets and across participants. It is important to note that scalograms utilize the continuous wavelet transform (CWT). They can represent frequencies higher than the Nyquist Frequency defined in section 3.1. They are still similarly constrained however, and information presented above 150 Hz becomes exponentially unreliable.

Jaw clenching produced a high-amplitude, low-frequency signal, typically around 1-4 Hz. Eye blinking produced a strong signal in the 20-40 Hz range. Eye rolling produced a strong signal in the 50-90 Hz range. Neck turning, similar to Eye rolling, produced a signal in the 50-90 Hz range, but also demonstrated strong, low-frequency transients when the movement began and ended. Below are scalograms demonstrating these different tasks and their signals in collected datasets. Similar to the EEG waveform plots, the scalograms confirm that each channel behaves differently with some motor task signals being more prevalent in certain channels than others as shown in figure 5.4 below. This directly implies that not only can signals originate in specific sections of the brain, but that noise can obfuscate the motor signals and make them less distinguishable.

Arm Movement Motor Task Unreliability Arm movement experiments did not produce consistent results. This was unlike the scalogram data for the jaw clench, eye blink, eye roll, and neck turn tasks. Arm movement results would fluctuate from subject to subject, and occasionally dataset to dataset for the same subject. These subject differences in patterns were apparent in scalogram data as well as classification metrics. Figure 5.5 demonstrates these differences. This was an early indication that the arm movement task would have poor performance in later analysis.

5.4 Anomalous Artifact Findings

During the course of data collection and analysis, several anomalous artifacts become apparent in data. These artifacts will be referred to as the line-noise artifact, spike artifact, and

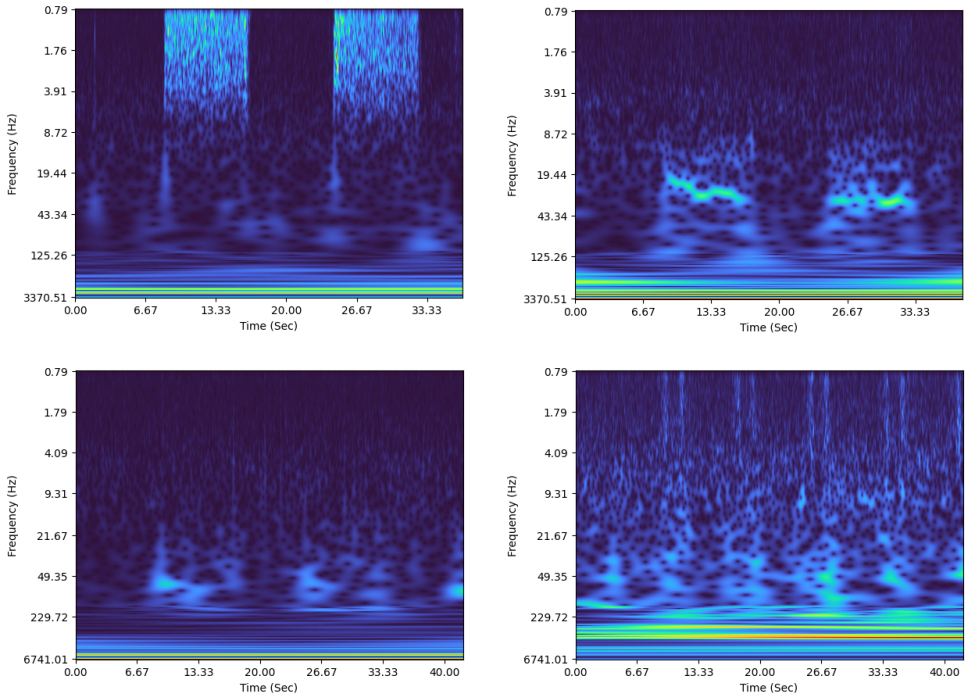

Figure 5.3: Scalograms of Motor Imagery Types: Jaw Clench (top left), Eye Blink (top right), Eye Roll (bottom left), Neck Turn (bottom right). Each motor task displays a different pattern during execution than the others.

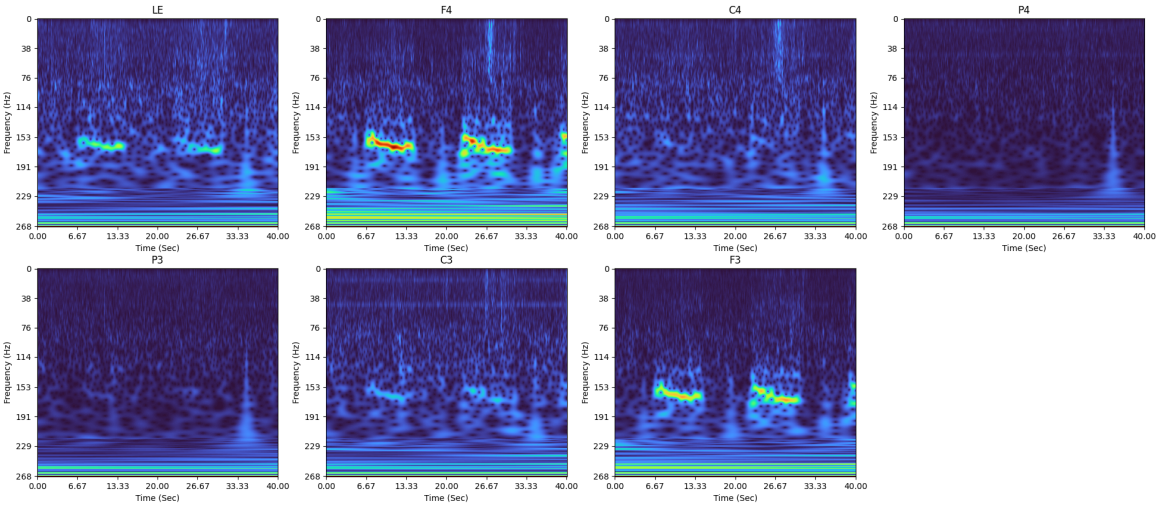

Figure 5.4: Scalograms of each channel of EEG data for the Eye Blink Motor Task. These plots show the differences between channels. Specifically how channel F4 and F3 strongly show the Eye Blink signal, while P4 and P3 do not at all.

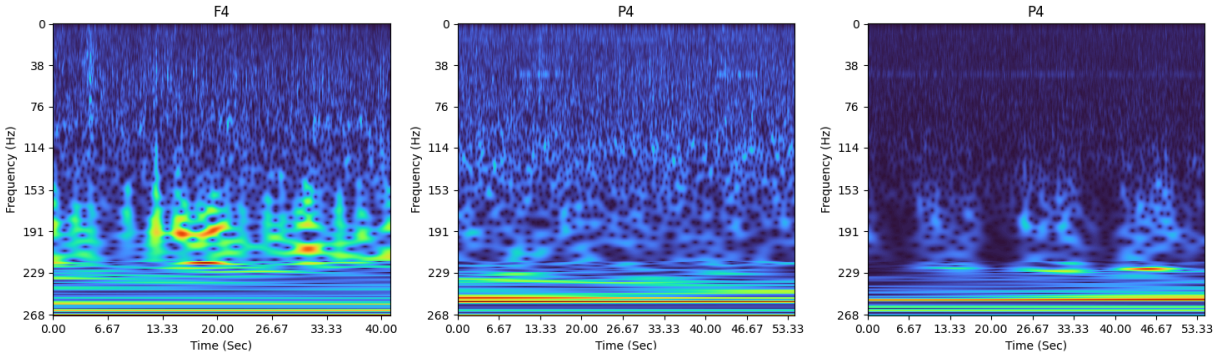

Figure 5.5: Scalograms showing differences between three different subjects for the Arm Movement Motor Task.

cardiac artifact. The artifacts are demonstrated below in figure 5.6. Each artifact appeared in multiple data sets and varied in strength from virtually non-existent, to overwhelming. The line-noise artifact was the most prevalent appearing in approximately 20% of data sets. It often was also the strongest disruption of the three artifacts. The spike artifact appeared in fewer data sets, but was not constant and typically appeared at the beginning or end of a motor task, though this was not always true. The cardiac artifact appeared in specific individuals and not others. For several individuals, the cardiac cycle was so strong that it appeared in raw EEG data with no further analysis, whereas others required ICA for it to be apparent. Overall, these three artifacts appeared in a significant number of data sets and likely impacted classification scores.

Of the two artifacts, only the spike anomaly should significantly impact classification scores. The experimental design as described in section 4.2 in combination with the function of machine learning models in section 3.2 mitigates the impact of the line-noise and heart-beat artifacts. Both of these artifacts appear in all of a specific individuals data sets, this means both positive and negative samples would be impacted equally. Thus, machine learning would learn to treat the artifacts as noise. If an anomaly appears in specific data sets and specific points in time, such as the spike anomaly, this is no longer the case and here we rely upon multiple data sets, cross-fold validation, and the scores of other individuals to mediate the effect of the anomaly. Thus while there are anomalies their impact should be minimized.

5.5 Preprocessing Results

The preprocessing techniques used in this study included frequency filtering (FF), independent component analysis (ICA), and common spatial patterns (CSP). The effectiveness of these techniques was evaluated based on their impact on classification metrics such as accuracy, F1 score, recall, and precision across all utilized machine learning models. Tukey HSD results for averaged motor tasks are listed in Appendix A. Additional box plots for each motor task are provided in Appendix C.

The results of the experiment show that two configurations are particularly effective

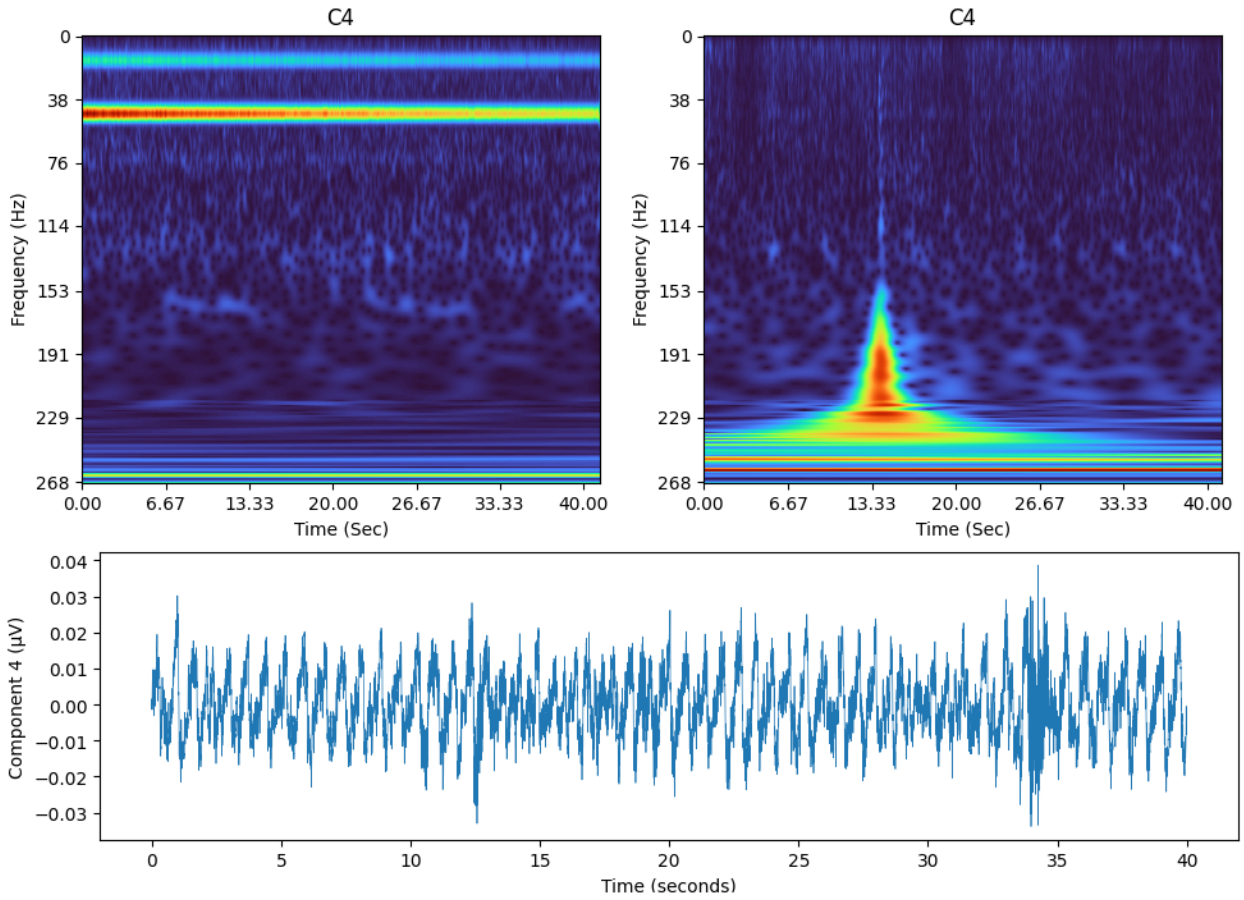

Figure 5.6: Scalograms demonstrating the Line-Noise Artifact (left), Spike Artifact (right) and Cardiac Artifact (bottom).

in improving classification metrics. Specifically, combining ICA and CSP has the greatest impact on classification results, while FF has the least as shown in Figure 5.7 below. As a result, the most effective configurations are either ICA and CSP, or FF, ICA, and CSP. Moreover, the combination of all three consistently achieved the highest results in each category. FF, ICA, and CSP and ICA, CSP had greater variance for F1, precision, and recall than any other configurations. While they did score significantly higher, their results were also over a wider range. For many preprocessing technique configurations, accuracy had a greater value than F1, precision, and recall. The overall range of accuracy scores are grouped together more closely than F1, precision, and recall. Several preprocessing techniques such as FF, ICA; FF, CSP; FF; and ICA overall performed worse than using no preprocessing techniques at all.

These relationships are statistically significant and demonstrate that the configuration of preprocessing techniques has an effect overall on classification metrics as shown in Table 5.2 below. However, preprocessing configuration did not have a significant effect on classification accuracy in the eye roll and neck turn tasks. Further, accuracy was consistently less significant than F1 score, precision, and recall. Preprocessing configuration did not have a significant effect on any metric for the arm movement motor task. FF and CSP appeared as a top performer in the jaw clench task, separating it from the other tasks, as it was the only task where any combination with CSP was most effective, but not CSP alone. This is unusual as the other tasks valued CSP and ICA together as a combination. Each of the stated relationships was validated with pairwise comparisons using Tukey's HSD after establishing that the result was significant overall with ANOVA as defined in section 4.5. Combined, these results show that while many individual results fail to reject the null hypothesis, there are overall greater trends that validate the effectiveness of these three preprocessing techniques on increasing classification metrics and do reject the null hypothesis.

These results highlight how preprocessing techniques can increase the signal-to-noise (SNR) ratio and increase classification scores as a result. The act of increasing the SNR is of significant importance to BCI as noise is the primary obstacle to good classifications

and can often obfuscate a useful signal. While this work focuses on the classification of very obvious signals, its application to brain pathology such as spatial neglect is shown by increasing the SNR. Spatial neglect manifests in an very subtle and fine-grained signal that is often difficult to detect. Utilizing preprocessing to detect such a signal is a must, and the greatest SNR is required. Therefore these results showing that preprocessing techniques yield greater classification scores can be further applied to more subtle signals with confidence.

Table 5.2: Statistical Power and Effect Size Analysis for Preprocessing Techniques Per Metric for Different Motor Imagery Tasks. F1, Precision, and Recall were more significant than Accuracy. Neck Turn experiments yielded insignificant results for Accuracy and F1. Arm Movement experiments were insignificant across all metrics for preprocessing techniques.

Metric	Accuracy		F1		Precision		Recall	
Task	$p-value$	np^2	$p-value$	np^2	$p-value$	np^2	$p-value$	np^2
Average	0.0001	0.4424	0.0000	0.5817	0.0000	0.6120	0.0000	0.5708
Jaw Clench	0.0143	0.4867	0.0001	0.6938	0.0005	0.6299	0.0000	0.7307
Eye Blink	0.0000	0.4799	0.0000	0.5896	0.0000	0.6109	0.0000	0.5824
Eye Roll	0.0543	0.3313	0.0011	0.5035	0.0002	0.5613	0.0035	0.4604
Neck Turn	0.4304	0.1518	0.0244	0.3157	0.0021	0.4113	0.1522	0.2230
Arm Movement	0.8387	0.0958	0.4210	0.1859	0.1899	0.2518	0.6417	0.1390

5.6 Classification Results

Seven Machine Learning (ML) models were chosen to evaluate the performance of the preprocessing techniques. The chosen models were Naive Bayes, kNN, linear SVM, RBF SVM, LDA, random forest, and MLP NN. These models represent a significant range of machine learning models each with strengths and weaknesses. Each task saw some variance in the effectiveness of the models and their performance in each category of metrics. Tukey HSD

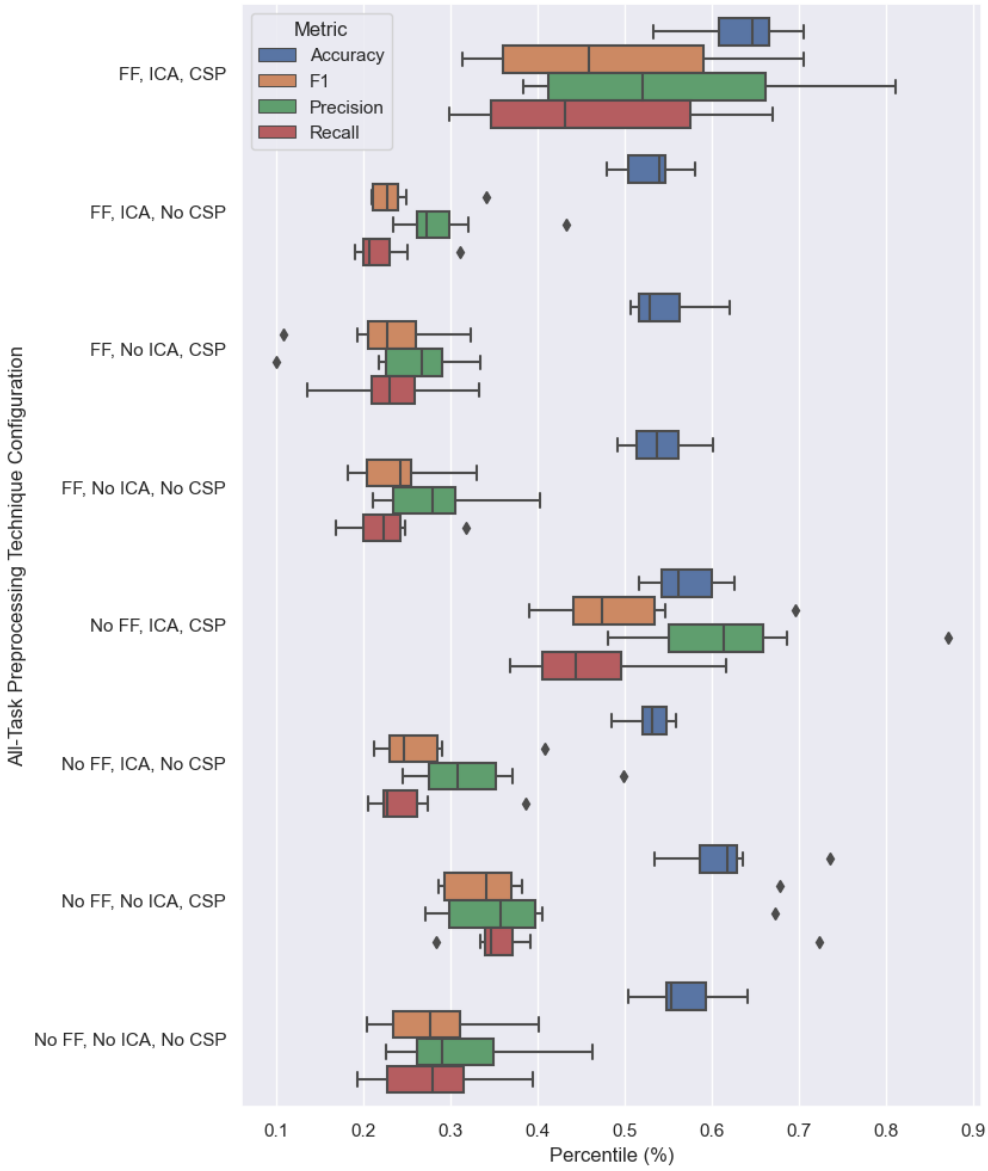

Figure 5.7: Classification Metrics Across Preprocessing Technique Configurations. FF, ICA, CSP; and ICA, CSP performed the greatest overall. This is reflected in their boxes being further to the right than the other configurations. They also had the greatest variance compared to other configurations shown by the width of their boxes and whiskers. Accuracy was more closely grouped than F1, accuracy, and precision. Some configurations performed worse than no preprocessing techniques at all.

results for averaged motor tasks are listed in Appendix B. Additional box plots for each motor task are provided in Appendix D.

Results show that the leading ML models in average across all tasks, and in each task was Naive Bayes and random forest. Tables 5.3 and 5.4 detail the best performing models by motor task and classification metric. Any result with a given value indicates $p < 0.05$ through ANOVA as defined in section 4.5. Multiple best performing models exist as pairwise analysis by Tukey HSD indicates no significant differences between them in the data. The high performance of Naive Bayes may have been attributable to the low sample size of the data. Other models may prefer more data, even orders of magnitude more, to perform sufficiently. Several other models appeared to perform the best in specific circumstances. Linear SVM appeared to perform well in data averaged across all tasks, however it was not a top performing model in any specific task. Similarly, the MLP neural network model performed well when averaged across all task data for the recall metric only. The kNN and LDA models did not perform better than the other models in any task for any metric. ML Models had less effect on classification scores than preprocessing technique, but they still had an impact that is significant.

RBF SVM performed significantly worse than the other models in each motor task. This poor performance may be caused by the relative complexity of RBF SVM. It generates a higher dimensional decision boundary that may not conform well to the data. Further, with no parameter tuning, the chosen settings may be particularly poor for performance. In some cases, the performance of RBF SVM was anomalously low, with scores of 0 appearing.

For ML Model analysis the accuracy metric was most often not significant when analyzing across all preprocessing techniques which is shown in Table 5.2. This reflects the preprocessing technique results in that F1 score, precision, and recall outperformed accuracy. Given a significance level of 0.05 only eye blinking had significant accuracy results using machine learning models as groups. However, this result from eye blinking is only significant because RBF SVM performed so poorly that it was significantly different from every other group. RBF SVM was a significant difference in other metrics as well, such as in the neck turn

motor task. In this task, RBF SVM was the highest performing model in recall, and the worst performing model in precision. Neck turn accuracy and F1 score were functionally identical across ML models. The ML model experiment was less significant overall than the preprocessing technique results.

Table 5.3: Best ML Models by Metric Over All Tasks.

Metric	Jaw Clench
Accuracy	NS
F1	Naive Bayes, Random Forest, and Linear SVM
Precision	Naive Bayes, Random Forest, and Linear SVM
Recall	Naive Bayes, Random Forest, Linear SVM, and MLP NN

Note: "NS" denotes "Not Significant."

Table 5.4: Best ML Models by Metric Per Task

Metric	Jaw Clench	Eye Blink	Eye Roll	Neck Turn	Arm Move
Accuracy	NS	All but RBF SVM	NS	NS	NS
F1	Naive Bayes and Random Forest	Naive Bayes and Random Forest	NS	NS	NS
Precision	Naive Bayes	Naive Bayes	NS	RBF SVM	NS
Recall	Naive Bayes and Random Forest	Naive Bayes and Random Forest	NS	All but RBF SVM	NS

Note: "NS" denotes "Not Significant."

For the purposes of analyzing the best performing ML model the eye roll, neck turn, and arm move motor tasks did not produce results demonstrating a significant difference in the performance of the chosen classification metrics. This is shown in both Table 5.4 and 5.5. These results were above the significance level of 0.05 and thus there is a high likelihood their results may be attributable to random chance.

Analyzing preprocessing technique configuration with respect to subsets of machine learning models reveals nuances that the averages do not capture. The Naive Bayes ML model

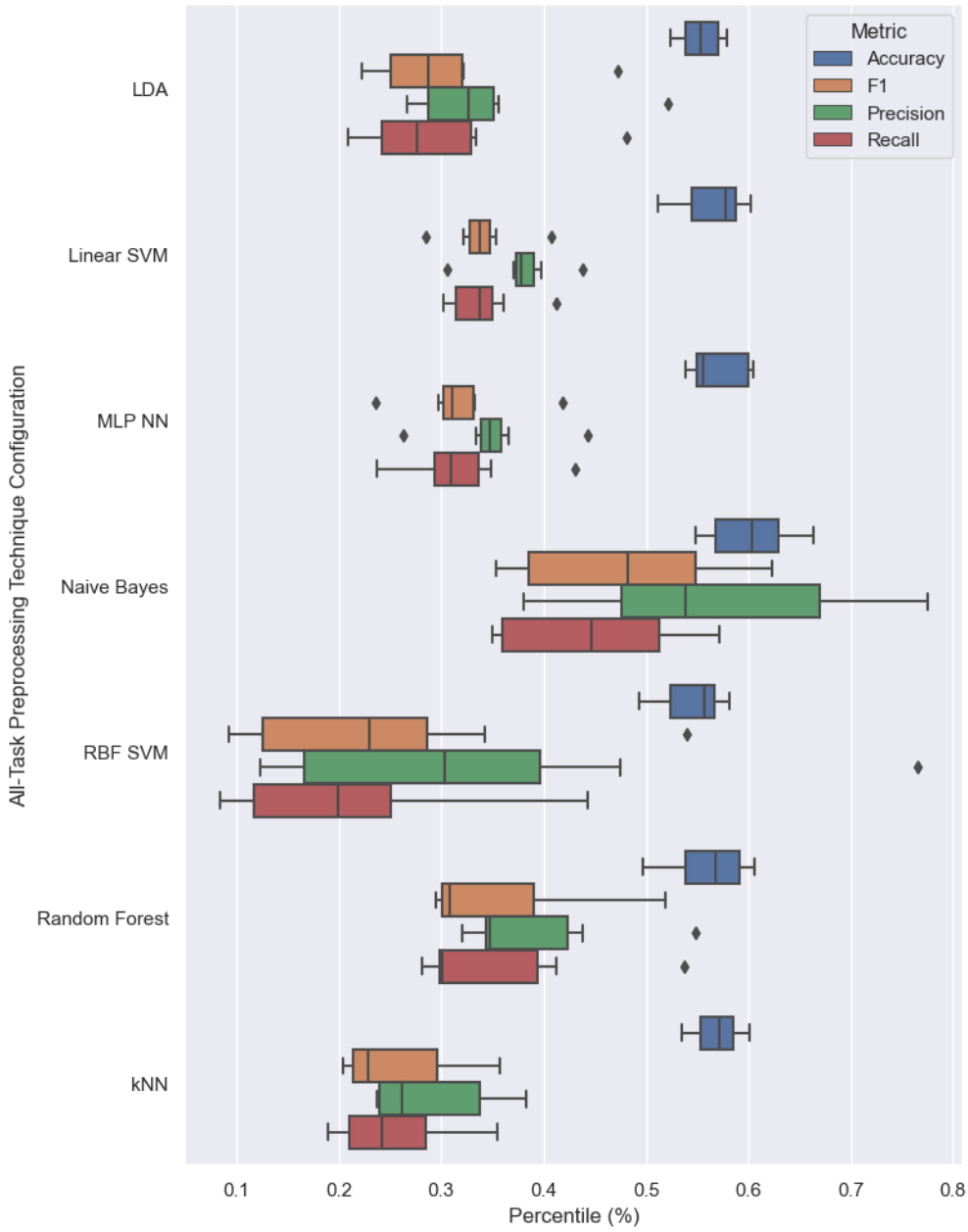

Figure 5.8: Classification Metrics for ML Models. Naive Bayes and random forest were the top performing models represented by their box and whiskers being further to the right. Naive Bayes had a significant amount of variance compared to other models shown by the greater width of its box and whiskers. Most models had less variance than Naive Bayes. Accuracy was more closely grouped and generally was higher than the other metrics.

performs nearly as effectively with no preprocessing techniques, as it does with all of them as demonstrated in Figure 5.9. Simply utilizing CSP alone would also have great benefit to Naive Bayes. Other ML models did not display this same tendency as shown in Figure 5.10. Further, Naive Bayes did not benefit from ICA,CSP in comparison to the other ML models. Among the box-and-whisker plots, only Naive Bayes indicates improvement with no preprocessing configurations at all. Conversely, FF, ICA; FF, CSP; FF; and ICA seem to exhibit worse performance compared to no preprocessing at all for all other ML models, though model by model, this relationship varies in statistical significance by classification metric. In addition to performing worse than other models, RBF SVM fails to reject the null hypothesis when using preprocessing techniques as the independent variable. This likely contributes to its poor performance.These two ML models perform differently than the remaining ML models, and this reveals nuance that is missed when looking at only one experimental question or the other.

Table 5.5: Statistical Power and Effect Size Analysis For ML Model Per Metric for Different Motor Tasks. F1, Precision, and Recall were more significant than Accuracy. Statistical Power was worse than Preprocessing Techniques. Eye Roll, Neck Turn, and Arm Movement were not significant. Effect sizes are provided through partial eta squared (ηp^2).

Metric	Accuracy		F1		Precision		Recall	
Task	$p-value$	np^2	$p-value$	np^2	$p-value$	np^2	$p-value$	np^2
Average	0.0803	0.2263	0.0006	0.4158	0.0023	0.3729	0.0002	0.4451
Jaw Clench	0.1243	0.3544	0.0024	0.5907	0.0040	0.5671	0.0023	0.5915
Eye Blink	0.0083	0.3253	0.0000	0.5176	0.0001	0.4853	0.0000	0.5197
Eye Roll	0.9953	0.0218	0.1034	0.2973	0.1417	0.2751	0.0746	0.3189
Neck Turn	0.6674	0.1043	0.8850	0.0616	0.0049	0.3959	0.0000	0.5911
Arm Movement	0.9210	0.0642	0.5911	0.2565	0.4366	0.1784	0.1027	0.2977

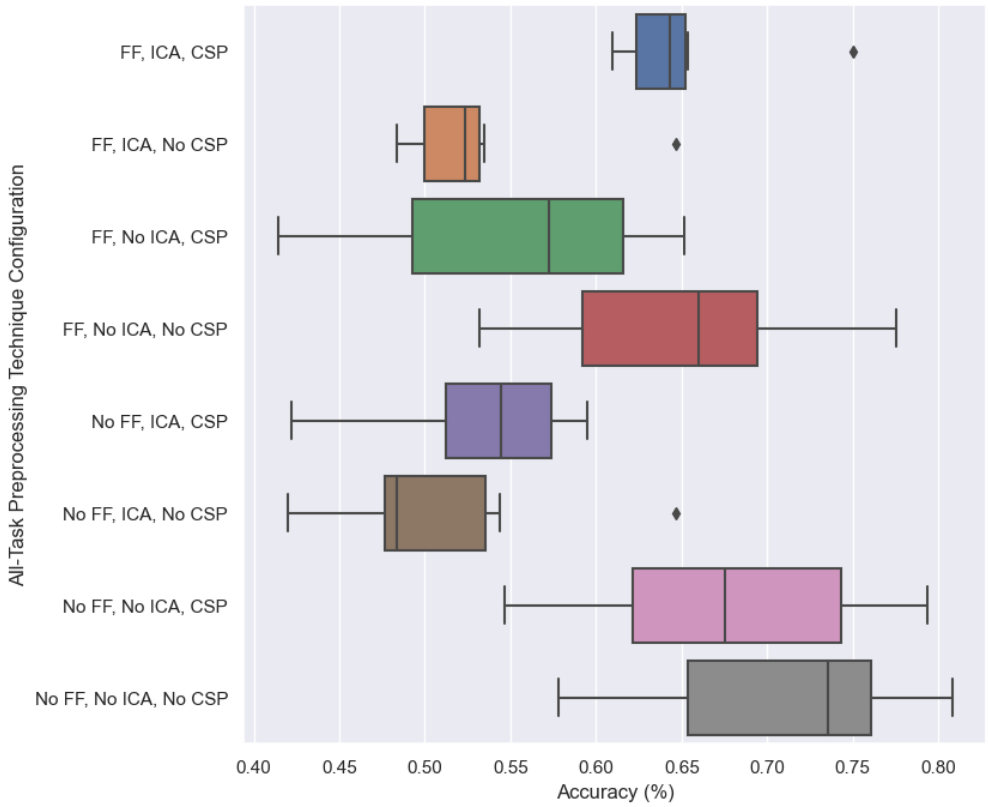

Figure 5.9: Accuracy of Naive Bayes across Preprocessing Techniques. FF, ICA, CSP performed just as well as CSP and no preprocessing techniques. While no preprocessing appears higher, it is not statistically significantly different. Naive Bayes had a significant amount of variance in all configurations.

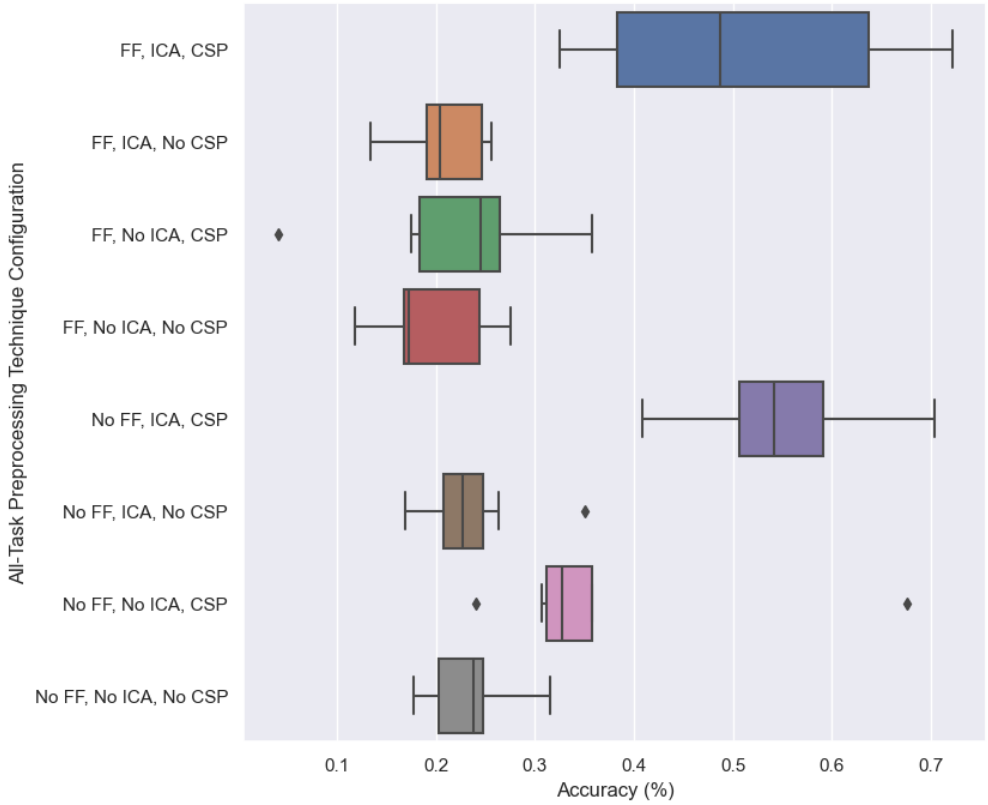

Figure 5.10: Accuracy of ML Models without Naive Bayes or RBF SVM across Preprocessing Techniques. FF, ICA, CSP and ICA, CSP performed the greatest shown by their boxes being further to the right. Similarly to Naive Bayes the higher performers had greater variance. Compared to figure 5.9 this shows how most models had less variance.

5.7 Summary

Effect of Preprocessing Technique Configurations The aim of this experiment was to investigate the effect of preprocessing technique configurations on classification metrics of classifying motor tasks in EEG data. The results indicate that ICA and CSP yielded the greatest increase to metrics, while FF contributed less. FF, ICA, and CSP together were found to be the most effective on average across tasks and for each task individually. However, preprocessing configuration did not have a significant effect on classification accuracy in the eye roll and neck turn motor tasks. Accuracy was also less significant than F1 score, precision, and recall. Similarly, preprocessing configuration did not have a significant effect on any metric for the arm movement motor task. Certain preprocessing techniques performed worse than using no techniques at all. These results showing that preprocessing techniques yield greater classification scores can be further applied to more subtle signals with confidence, such as brain pathology like spatial neglect.

Effect of Machine Learning Model The aim of this experiment was to examine the effect of machine learning models on classification metrics of classifying motor tasks in EEG data. The results show that Naive Bayes and random forest yielded the greatest increase in their metrics. Linear SVM was commonly a top performer in specific metrics for specific tasks, while MLP NN was occasionally a top performer. Neck turn accuracy and F1 score were functionally identical across ML models. RBF SVM was often the worst performing ML model. The effect of machine learning models on classification accuracy was found to be insignificant in most cases. The arm movement and eye roll motor tasks did not yield significant results for ML models. RBF SVM was the worst performing ML model in the eye blink motor task, but the preprocessing configuration did not have a significant effect on any metric for RBF SVM in any task. Naive Bayes strongly preferred either no preprocessing or all preprocessing, with both being statistically alike.

In conclusion, this experiment highlights the importance of choosing appropriate prepro-

cessing techniques and machine learning models for classifying motor tasks in EEG data. The findings of this study can be useful in developing efficient and accurate brain-computer interface systems for various applications.

Chapter 6
CONCLUSION

This section discusses the achievements of this work and our research contributions as well as identified and known limitations in the experiment, and possible areas for future work.

6.1 Conclusion

The overall goal of this work was to contribute to the field of Brain-Computer Interfacing's (BCI) knowledge of motor preprocessing and classification by evaluating preprocessing techniques in multiple niche environments. For this work, those niche environments were the motor tasks of jaw clenching, eye blinking, eye rolling, neck turning, and arm movements. The pipeline behind this project encompasses most of the machine-learning software development life-cycle and its products are the classification scores of three different preprocessing techniques using seven different machine learning models. Its goal was to demonstrate a statistically significant configuration of preprocessing techniques that created a significant difference in classification scores, this goal was achieved.

Designing the Right Machine Learning Pipeline

This work has shown that the design of the motor classification pipeline can greatly influence classification results generated by the chosen machine learning (ML) model at its end. The choice of preprocessing techniques, their parameters, and their order can lead to significant changes in performance. The results in Chapter 5 demonstrate the effectiveness of frequency filtering (FF), independent component analysis (ICA) and common spatial patterns (CSP) on increasing classification metrics for different machine learning models and motor tasks. These

results could be further increased by tailoring the preprocessing techniques to the detection of that particular motor task. Likewise the choice of ML model can impact classification scores by aligning the benefits of each model with the type of motor task to be detected. ML models can also see significant benefit from hyperparameter tuning, potentially increasing classification further. For these reasons the choice of preprocessing techniques and ML model can have significant effect on the outcome of future experiments in motor task brain computer interfaces (BCI).

Summary of Contributions

This work creates a BCI pipeline from data collection to analysis. It creates variations for different motor tasks in formats of blocks or single instances. It applies different variations of this pipeline to achieve different motor classification scores in different environments. We have created a data collection process and data processing pipeline for use in offline motor task classification. Parts of it may be adaptable to a real-time pipeline.

We have demonstrated how data collection experiments could be designed for specific motor tasks. These data experiments were designed to minimize noise during the machine learning training phase. This work also shows how even with noise reduction measures, success of classification is not guaranteed as in the arm movement task. It shows that the motor task classification process is a complex one that requires specific experimental tailoring to achieve success.

We successfully show how certain motor tasks are represented in EEG data and scalogram plots. It shows how they react to individual characteristics such as prevalence of heart-rate, artifacts such as line noise, and temporal artifacts. This work demonstrates how to build a conceptual model of each motor task and create comparisons of them to other motor tasks for later use in machine learning classification.

This work has demonstrated the effectiveness of FF, ICA, and CSP on increasing classification metrics for motor tasks with ML models. We have explored the different capabilities of the preprocessing techniques, and demonstrated that they have specific strengths and

weaknesses, such as their performance when utilized alone versus being combined. We have demonstrated how this combination may yield greater results overall, but more varied results.

We have shown that the choice of ML model can have significant impact on classification metrics. We have explored possible scenarios for certain models to perform exceptionally or poorly. We have shown that overall the preprocessing techniques have a greater statistically relevant impact on classification scores than machine learning model. However, this result should be tempered by the lack of hyperparameter tuning.

The novelty of this work is not only reproducing this effectiveness, but demonstrating it across multiple uncommon motor tasks and in one experimental setting. The combination of these factors is novel in the field of BCI and can contribute to future BCI studies in motor task classification. It is especially relevant to the specific classification of jaw clenching and eye blinking. Though most studies seek to discard these motor tasks, there may be relevant use to instead detect and classify them.

This work is relevant to any study seeking to classify muscle signals or MEPs from the somatomotor cortex. It can be applied in both academic and medical environments on works that need to choose and apply preprocessing techniques. It can aid experimenters in choosing those techniques as well as how to apply them. It also provides avenues forward in the refinement and further development of studies predicated on these techniques with future work below. This work can essentially be used as a stepping-stone to classifying brain-waves, or more subtle signals that require more nuanced preprocessing such as brain pathology in the form of spatial neglect. The detection of spatial neglect in EEG is a subtle signal in comparison to the work done here, but the results showing how preprocessing techniques improve SNR reinforce that they will continue to do so even when applied in narrower circumstances.

We have contributed to the Spatial Neglect Rehabilitation Project at the University of Washington Bothell with the above contributions and the demonstration of the capability of the WearableSensing DSI7 EEG system. Before this work, its capability to detect and classify motor tasks was unknown, it is now without a doubt capable of detecting muscle

signals, and classifying them as such.

Limitations and Opportunities for improvement

Several limitations exist in this work that detract from its validity.

Data Source Data was collected for this experiment exclusively from University of Washington Bothell students. This limits the generalizability of the experiment to that group, and hampers its translation to other environments.

Sample Size The epoching process and the pipeline of classification taking place on trios of datasets, causes a concentration of data that has resulted in a low number of samples being utilized during the classification process. This leads to greater variability in the data and lower classification metrics. Certain types of ML models benefit greatly from larger sample sizes. It can therefore be concluded that the growth of classification scores for particular models, such as the MLP NN, may have been negatively effected overall.

No parameter tuning The experiments do not perform parameter tuning for either preprocessing technique configurations or machine learning models either over all tasks, or for each singular task. Both preprocessing techniques and machine learning models have been demonstrated in other works to benefit significantly from parameter tuning. These limitations primarily effect external validity and the generalizability of these findings to other environments.

Independent component analysis (ICA) does not discard noisy components ICA separates a signal in data into constituent components. This creates components with noise separate from components with a motor signal. The noisy signals constitute a useless source of noise for the ML model, and should be discarded to increase classification metrics. The act of retaining these channels would hamper classification scores as it impacts the model's

decision boundary. It is therefore significant that preprocessing technique configurations utilizing ICA demonstrated greater classification scores despite this limitation.

Common spatial patterns (CSP) is utilized with less than 8 channels CSP attempts to recognize and distinguish spatial patterns in EEG data. The effectiveness of these patterns is dependent on the amount of data present, including from the number of channels. The fewer channels, the lower the effect of the CSP transform on the classification metrics. Other works have demonstrated that the practical lower limit for CSP is 8 channels, the DSI-7 EEG device has 7 channels. Similar to the limitation above for ICA, it is therefore of significance that CSP still resulted in an increase in classification metrics despite this.

Multiple motor task experiments do not reject the null hypothesis Specific metric scores in preprocessing techniques and multiple motor task experiments in ML models did not meet the level of significance. In both preprocessing techniques and ML models all statistical values produced by ANOVA for the arm movement experiment were above the statistical level of significance ($p > 0.05$). As the other data collection experiments successfully generated classifications of motor tasks, this implies either these motor task experiments are malformed, or corrupted in some fashion or that the processing pipeline is not capable of recognizing the motor tasks. This harms both internal and external validity.

Collected data does not screen for anomalies While specific artifacts should have no impact on classification scores, the spike artifact likely does and its impact is unknown. It is arguable that perfectly screening data would reduce its generalizability overall. However as this experiment was conducted on an individual level allowing temporal anomalies to skew results would harm internal validity instead.

The purpose of this experiment is to evaluate the effectiveness of three preprocessing techniques (frequency filtering, independent component analysis, and common spatial patterns) and six machine learning models (Naive Bayes, kNN, linear SVM, RBF SVM, random

forest, and multi-layer perceptron neural network) on classification scores. The evaluation was done only with the selected metrics. It does not attempt to evaluate other preprocessing techniques, nor does it seek to evaluate other machine learning models.

6.2 Future Work

In future experiments, there are plenty of opportunities to investigate new directions in this research. In this section, we discuss some of the possible areas for development.

Data collection The process of collecting data is never complete. This has been born out with the growth of big data and large model machine learning (ML). In the case of motor classification there is a need for both consistency and evaluation of all cases. Therefore a balance needs to be drawn between collection of data from more participants in more environments, and keeping data consistent. Drawing from the resultant data, it is clear that the data collection experiments for neck turn and arm movement need to be redesigned and further experimented upon. Across both research questions, there were no significant results in arm movement. Further development is clearly needed to classify the presence or absence of arm movements from EEG data using these techniques. One such method is to tune the parameters for the existing preprocessing techniques and ML models.

Further work in improving the motor task sessions should be performed as well. This may involve the collection of new metrics such as event flags or classification scores, or encoded information about the participant such as physiological characteristics. This should be done in a standardized manner in accordance with current research in the field of BCI.

Exploring alternate preprocessing techniques and ML Models Preprocessing techniques exist outside of the three being evaluated in this work. Other preprocessing techniques and ML models effectiveness in increasing signal-to-noise ratio in EEG data is worth investigation in comparison to the selected models in this work. Similarly, the selected ML models represent a subset of potential ML models, some of which may be more effective than the

chosen ML models in this work.

Parameter tuning for preprocessing techniques and ML models For the purposes of consistency, parameter tuning was neglected in this experiment. However, the results indicate that each motor task and ML model would benefit from being tailored to the given data. In particular the effectiveness of frequency filtering in increasing classification scores of eye blinking implies that with the correct parameter tuning the technique could perform effectively in the other tasks as well. Independent component analysis (ICA) and common spatial patterns (CSP) may well benefit from parameter tuning to each motor task as well.

Conversion to real-time classification While this experiment benefits from offline analysis, many opportunities in BCI exist in real-time classification of motor tasks. The difficulty of this conversion exists in the definition of the epoching process and how to redefine it for the uncertainty of real-time operation.

Working with motor imagination The classification of motor imagination tasks is of great interest to the field of BCI today. Motor imagination differs greatly from the muscle-based motor classification of this work. It has the potential to allow individuals to control devices without the actual movement of any muscles. It is a different signal that is more subtle than the one investigated in this work.

Expanding beyond single-person ML models There is significant variance between individuals in motor classification. Over the past decade focus has been shifting in BCI from the training and testing of models on single individuals to multiple individuals to try and account for interpersonal variances.

Utilizing explainable ML Many ML models function in a black-box style, where training data is used to create a model that is nearly inexplicable in its decisions. Explainable ML seeks to generate connections between the training data and the decisions being made by

the model at the end. Work in this avenue could help to account for the power of anomalies in skewing data classifications, as well as authenticate future findings as legitimate. It may also provide insights into which channels and which portion of data have relevant features to classification, potentially creating avenues for future feature extraction and analysis.

Filtering for particular brainwaves Instead of prioritizing muscle signals, using particular brainwaves, such as beta and mu waves, could allow for more targeted classification. This would assist in filtering out unwanted anomalies. One method for targeting brainwaves could be the use of a band-pass filter set to the frequency of a particular wave. This could have significant benefits as EEG does not have enough spatial bandwidth to produce detailed information about muscle activation and strength of use. This would also directly lead to the classification of event-related potentials (ERP) and event-related synchronization/desynchronization (ERS/ERD).

Classifying multiple muscle tasks simultaneously Expanding beyond the classification of a single task and utilizing ensemble classifiers to predict the presence of multiple muscle tasks simultaneously would be an enormous but beneficial task. The presence of multiple muscle tasks simultaneously would be a confounding factor for the decision boundaries of the ML models which currently may treat each muscle task similarly. In such a scenario, the decision boundaries would require significant refining. Utilizing parameter tuning would also likely be required in such a use-case, as well as further preprocessing and filtering. Filtering for specific brainwaves may also provide useful insights to this avenue. Overall it could be of significant value to the field of BCI and present significant challenges.

Application to real-world patients There is an avenue for development in collecting data in real-world circumstances and training models either for each individual or across them. This work is done on single student volunteers which limits its generalizability. While this allows us to accurately evaluate the performance of these preprocessing techniques, it is

not a real-world application. Therefore this work can be applied to a real-world application to filter for noise such as event stimuli (sounds, sights) and motor stimuli (movements) with confidence that the selected preprocessing techniques would perform well. This could be true in either an academic or medical environment where real-patients are involved. If further work is performed in parameter tuning or any other avenue of development, it would only strengthen these findings.

BIBLIOGRAPHY

[1] Nyquist frequency.

[2] M. Abramowitz and I. A. Stegun. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. Dover, New York, 9th printing edition, 1972.

[3] Sarwar Ayesha and Emmady Prabhu. Spatial neglect. 5 2022.

[4] Xiaoping Bai, Xiangzhou Wang, Shuhua Zheng, and Mingxin Yu. The offline feature extraction of four-class motor imagery eeg based on ica and wavelet-csp. In *Proceedings of the 33rd Chinese Control Conference*, pages 7189–7194, 2014.

[5] Nima Bigdely-Shamlo, Jonathan Touryan, Alejandro Ojeda, Christian Kothe, Tim Mullen, and Kay Robbins. Automated eeg mega-analysis i: Spectral and amplitude characteristics across studies. *NeuroImage*, 207:116361, 2020.

[6] Luca de Raad. DSI flash lights api.m.

[7] Francesco Ferracuti, Sabrina Iarlori, Zahra Mansour, Andrea Monteriù, and Camillo Porcaro. Comparing between different sets of preprocessing, classifiers, and channels selection techniques to optimise motor imagery pattern classification system from eeg pattern recognition. *Brain sciences*, 12(1):57–, 2021.

[8] Daeun Gwon, Kyungho Won, Minseok Song, Chang S Nam, Sung Chan Jun, and Minkyu Ahn. Review of public motor imagery and execution datasets in brain-computer interfaces. *Frontiers in human neuroscience*, 17:1134869–1134869, 2023.

[9] Brahim Hamadicharef. Brain-computer interface (bci) literature - a bibliometric study.

In *10th International Conference on Information Science, Signal Processing and their Applications (ISSPA 2010)*, pages 626–629, 2010.

[10] Etnier Jennifer L and Gapin Jennifer I. *Encyclopedia of Sport and Exercise Psychology*. Gale eBooks, Thousand Oaks, CA, 2014.

[11] Li Korina and Malhotra Paresh A. Spatial neglect. *Practical Neurology*, 10 2015.

[12] Ajitesh KumarI. Level of significance & hypothesis testing, 2021.

[13] Blaser Larry. *The Gale Encyclopedia of Science*. Gale, Farmington Hills, MI, 5 edition, 2014.

[14] Clifford Maswanganyi, Chunling Tu, Pius Owolawi, and Shengzhi Du. Overview of artifacts detection and elimination methods for bci using eeg. In *2018 IEEE 3rd International Conference on Image, Vision and Computing (ICIVC)*, pages 832–836, 2018.

[15] Media Title. homunculus. Image, N/A. Accessed May 21, 2023.

[16] Remigiusz J. Rak, Marcin Kołodziej, and Andrzej Majkowski. Brain-computer interface as measurement and control system the review paper. *Metrology and Measurement Systems*, (No 3):427–444, 2012.

[17] John T.E. Richardson. Eta squared and partial eta squared as measures of effect size in educational research. *Educational Research Review*, 6(2):135–147, 2011.

[18] Martin Justinus Rosenfelder, Myra Spiliopoulou, Burkhard Hoppenstedt, Rüdiger Pryss, Patrick Fissler, Mario della Piedra Walter, Iris-Tatjana Kolassa, and Andreas Bender. Stability of mental motor-imagery classification in eeg depends on the choice of classifier model and experiment design, but not on signal preprocessing. *Frontiers in computational neuroscience*, 17, 2023.

[19] Yannick Roy, Hubert Banville, Isabela Albuquerque, Alexandre Gramfort, Tiago H Falk, and Jocelyn Faubert. Deep learning-based electroencephalography analysis: a systematic review. *Journal of neural engineering*, 16(5):051001–051001, 2019.

[20] William R. Shadish, Thomas D. Cook, and Donald T. Campbell. *Experimental and Quasi-Experimental Designs for Generalized Causal Inference*. Houghton Mifflin Company, 2002.

[21] The McGill Physiology Virtual Lab. Biomedical signals acquisition, EEG, introduction, 2022.

[22] Raphael Vallat. Pingouin.Anova. Pingouin 0.5.3 Documentation, Accessed 21 May 2023.

[23] Wearable Sensing. DSI-Streamer.

[24] Piotr Wierzgała, Dariusz Zapała, Grzegorz M. Wojcik, and Jolanta Masiak. Most popular signal processing methods in motor-imagery bci: A review and meta-analysis. *Frontiers in Neuroinformatics*, 12, 2018.

[25] Andrea Zimmermann-Schlatter, Corina Schuster, Milo A Puhan, Ewa Siekierka, and Johann Steurer. Efficacy of motor imagery in post-stroke rehabilitation: a systematic review. *Journal of neuroengineering and rehabilitation*, 5(1):1–10, 2008.

Appendix A

PREPROCESSING TECHNIQUE CONFIGURATION COMBINED MOTOR TASK TUKEY HSD RESULTS

Table A.1: All Task Accuracy Tukey HSD Comparison. ANOVA:[$p = 0.0001, \eta p^2 = 0.4423$]

A	B	mean(A)	mean(B)	p-tukey	eta-square
FF, ICA, CSP	FF, ICA, No CSP	0.6330	0.5290	0.0016	0.5423
FF, ICA, CSP	FF, No ICA, CSP	0.6330	0.5455	0.0131	0.4267
FF, ICA, CSP	FF, No ICA, No CSP	0.6330	0.5404	0.0071	0.4672
FF, ICA, CSP	No FF, ICA, CSP	0.6330	0.5696	0.1600	0.2886
FF, ICA, CSP	No FF, ICA, No CSP	0.6330	0.5302	0.0019	0.5669
FF, ICA, CSP	No FF, No ICA, CSP	0.6330	0.6170	0.9973	0.0175
FF, ICA, CSP	No FF, No ICA, No CSP	0.6330	0.5689	0.1516	0.2783
FF, ICA, No CSP	FF, No ICA, CSP	0.5290	0.5455	0.9968	0.0443
FF, ICA, No CSP	FF, No ICA, No CSP	0.5290	0.5404	0.9997	0.0238
FF, ICA, No CSP	No FF, ICA, CSP	0.5290	0.5696	0.6865	0.2287
FF, ICA, No CSP	No FF, ICA, No CSP	0.5290	0.5302	1.0000	0.0004
FF, ICA, No CSP	No FF, No ICA, CSP	0.5290	0.6170	0.0124	0.4324
FF, ICA, No CSP	No FF, No ICA, No CSP	0.5290	0.5689	0.7027	0.2028
FF, No ICA, CSP	FF, No ICA, No CSP	0.5455	0.5404	1.0000	0.0039
FF, No ICA, CSP	No FF, ICA, CSP	0.5455	0.5696	0.9711	0.0793
FF, No ICA, CSP	No FF, ICA, No CSP	0.5455	0.5302	0.9980	0.0463
FF, No ICA, CSP	No FF, No ICA, CSP	0.5455	0.6170	0.0763	0.3109
FF, No ICA, CSP	No FF, No ICA, No CSP	0.5455	0.5689	0.9750	0.0687
FF, No ICA, No CSP	No FF, ICA, CSP	0.5404	0.5696	0.9218	0.1204
FF, No ICA, No CSP	No FF, ICA, No CSP	0.5404	0.5302	0.9999	0.0237
FF, No ICA, No CSP	No FF, No ICA, CSP	0.5404	0.6170	0.0453	0.3515
FF, No ICA, No CSP	No FF, No ICA, No CSP	0.5404	0.5689	0.9297	0.1050
No FF, ICA, CSP	No FF, ICA, No CSP	0.5696	0.5302	0.7175	0.2564
No FF, ICA, CSP	No FF, No ICA, CSP	0.5696	0.6170	0.5008	0.1702
No FF, ICA, CSP	No FF, No ICA, No CSP	0.5696	0.5689	1.0000	0.0001
No FF, ICA, No CSP	No FF, No ICA, CSP	0.5302	0.6170	0.0143	0.4527
No FF, ICA, No CSP	No FF, No ICA, No CSP	0.5302	0.5689	0.7332	0.2234
No FF, No ICA, CSP	No FF, No ICA, No CSP	0.6170	0.5689	0.4841	0.1649

Table A.2: All Task F1 Score Tukey HSD Comparison. ANOVA:[$p = 0.0000, \eta p^2 = 0.5817$]

A	B	mean(A)	mean(B)	p-tukey	eta-square
FF, ICA, CSP	FF, ICA, No CSP	0.4830	0.2402	0.0005	0.5186
FF, ICA, CSP	FF, No ICA, CSP	0.4830	0.2278	0.0002	0.5215
FF, ICA, CSP	FF, No ICA, No CSP	0.4830	0.2397	0.0005	0.5169
FF, ICA, CSP	No FF, ICA, CSP	0.4830	0.5015	1.0000	0.0048
FF, ICA, CSP	No FF, ICA, No CSP	0.4830	0.2718	0.0033	0.4297
FF, ICA, CSP	No FF, No ICA, CSP	0.4830	0.3763	0.4374	0.1142
FF, ICA, CSP	No FF, No ICA, No CSP	0.4830	0.2822	0.0062	0.4006
FF, ICA, No CSP	FF, No ICA, CSP	0.2402	0.2278	1.0000	0.0111
FF, ICA, No CSP	FF, No ICA, No CSP	0.2402	0.2397	1.0000	0.0000
FF, ICA, No CSP	No FF, ICA, CSP	0.2402	0.5015	0.0001	0.7325
FF, ICA, No CSP	No FF, ICA, No CSP	0.2402	0.2718	0.9985	0.0703
FF, ICA, No CSP	No FF, No ICA, CSP	0.2402	0.3763	0.1597	0.3046
FF, ICA, No CSP	No FF, No ICA, No CSP	0.2402	0.2822	0.9910	0.1098
FF, No ICA, CSP	FF, No ICA, No CSP	0.2278	0.2397	1.0000	0.0099
FF, No ICA, CSP	No FF, ICA, CSP	0.2278	0.5015	0.0001	0.7142
FF, No ICA, CSP	No FF, ICA, No CSP	0.2278	0.2718	0.9881	0.0960
FF, No ICA, CSP	No FF, No ICA, CSP	0.2278	0.3763	0.0940	0.3179
FF, No ICA, CSP	No FF, No ICA, No CSP	0.2278	0.2822	0.9610	0.1327
FF, No ICA, No CSP	No FF, ICA, CSP	0.2397	0.5015	0.0001	0.7286
FF, No ICA, No CSP	No FF, ICA, No CSP	0.2397	0.2718	0.9983	0.0694
FF, No ICA, No CSP	No FF, No ICA, CSP	0.2397	0.3763	0.1567	0.3032
FF, No ICA, No CSP	No FF, No ICA, No CSP	0.2397	0.2822	0.9904	0.1080
No FF, ICA, CSP	No FF, ICA, No CSP	0.5015	0.2718	0.0011	0.6421
No FF, ICA, CSP	No FF, No ICA, CSP	0.5015	0.3763	0.2426	0.2113
No FF, ICA, CSP	No FF, No ICA, No CSP	0.5015	0.2822	0.0021	0.6117
No FF, ICA, No CSP	No FF, No ICA, CSP	0.2718	0.3763	0.4645	0.1893
No FF, ICA, No CSP	No FF, No ICA, No CSP	0.2718	0.2822	1.0000	0.0057
No FF, No ICA, CSP	No FF, No ICA, No CSP	0.3763	0.2822	0.5969	0.1560

Table A.3: All Task Precision Tukey HSD Comparison. ANOVA:[$p = 0.0000, \eta p^2 = 0.6119$]

A	B	mean(A)	mean(B)	p-tukey	eta-square
FF, ICA, CSP	FF, ICA, No CSP	0.5519	0.2948	0.0016	0.4696
FF, ICA, CSP	FF, No ICA, CSP	0.5519	0.2480	0.0001	0.5436
FF, ICA, CSP	FF, No ICA, No CSP	0.5519	0.2825	0.0008	0.4942
FF, ICA, CSP	No FF, ICA, CSP	0.5519	0.6262	0.9082	0.0532
FF, ICA, CSP	No FF, ICA, No CSP	0.5519	0.3299	0.0097	0.3795
FF, ICA, CSP	No FF, No ICA, CSP	0.5519	0.3851	0.1102	0.2119
FF, ICA, CSP	No FF, No ICA, No CSP	0.5519	0.3148	0.0046	0.4109
FF, ICA, No CSP	FF, No ICA, CSP	0.2948	0.2480	0.9926	0.0969
FF, ICA, No CSP	FF, No ICA, No CSP	0.2948	0.2825	1.0000	0.0088
FF, ICA, No CSP	No FF, ICA, CSP	0.2948	0.6262	0.0000	0.7281
FF, ICA, No CSP	No FF, ICA, No CSP	0.2948	0.3299	0.9988	0.0499
FF, ICA, No CSP	No FF, No ICA, CSP	0.2948	0.3851	0.7855	0.1495
FF, ICA, No CSP	No FF, No ICA, No CSP	0.2948	0.3148	1.0000	0.0167
FF, No ICA, CSP	FF, No ICA, No CSP	0.2480	0.2825	0.9989	0.0560
FF, No ICA, CSP	No FF, ICA, CSP	0.2480	0.6262	0.0000	0.7651
FF, No ICA, CSP	No FF, ICA, No CSP	0.2480	0.3299	0.8566	0.2030
FF, No ICA, CSP	No FF, No ICA, CSP	0.2480	0.3851	0.2998	0.2761
FF, No ICA, CSP	No FF, No ICA, No CSP	0.2480	0.3148	0.9457	0.1446
FF, No ICA, No CSP	No FF, ICA, CSP	0.2825	0.6262	0.0000	0.7440
FF, No ICA, No CSP	No FF, ICA, No CSP	0.2825	0.3299	0.9920	0.0888
FF, No ICA, No CSP	No FF, No ICA, CSP	0.2825	0.3851	0.6599	0.1862
FF, No ICA, No CSP	No FF, No ICA, No CSP	0.2825	0.3148	0.9993	0.0432
No FF, ICA, CSP	No FF, ICA, No CSP	0.6262	0.3299	0.0002	0.6515
No FF, ICA, CSP	No FF, No ICA, CSP	0.6262	0.3851	0.0037	0.4542
No FF, ICA, CSP	No FF, No ICA, No CSP	0.6262	0.3148	0.0001	0.6736
No FF, ICA, No CSP	No FF, No ICA, CSP	0.3299	0.3851	0.9807	0.0550
No FF, ICA, No CSP	No FF, No ICA, No CSP	0.3299	0.3148	1.0000	0.0077
No FF, No ICA, CSP	No FF, No ICA, No CSP	0.3851	0.3148	0.9297	0.0862

Table A.4: All Task Recall Tukey HSD Comparison. ANOVA:[$p = 0.0000, \eta p^2 = 0.5708$]

A	B	mean(A)	mean(B)	p-tukey	eta-square
FF, ICA, CSP	FF, ICA, No CSP	0.4635	0.2245	0.0003	0.5451
FF, ICA, CSP	FF, No ICA, CSP	0.4635	0.2339	0.0006	0.5034
FF, ICA, CSP	FF, No ICA, No CSP	0.4635	0.2282	0.0004	0.5327
FF, ICA, CSP	No FF, ICA, CSP	0.4635	0.4622	1.0000	0.0000
FF, ICA, CSP	No FF, ICA, No CSP	0.4635	0.2556	0.0026	0.4557
FF, ICA, CSP	No FF, No ICA, CSP	0.4635	0.3970	0.8749	0.0481
FF, ICA, CSP	No FF, No ICA, No CSP	0.4635	0.2790	0.0106	0.3857
FF, ICA, No CSP	FF, No ICA, CSP	0.2245	0.2339	1.0000	0.0075
FF, ICA, No CSP	FF, No ICA, No CSP	0.2245	0.2282	1.0000	0.0017
FF, ICA, No CSP	No FF, ICA, CSP	0.2245	0.4622	0.0004	0.7562
FF, ICA, No CSP	No FF, ICA, No CSP	0.2245	0.2556	0.9982	0.0795
FF, ICA, No CSP	No FF, No ICA, CSP	0.2245	0.3970	0.0209	0.3874
FF, ICA, No CSP	No FF, No ICA, No CSP	0.2245	0.2790	0.9522	0.1767
FF, No ICA, CSP	FF, No ICA, No CSP	0.2339	0.2282	1.0000	0.0026
FF, No ICA, CSP	No FF, ICA, CSP	0.2339	0.4622	0.0007	0.6980
FF, No ICA, CSP	No FF, ICA, No CSP	0.2339	0.2556	0.9998	0.0295
FF, No ICA, CSP	No FF, No ICA, CSP	0.2339	0.3970	0.0348	0.3411
FF, No ICA, CSP	No FF, No ICA, No CSP	0.2339	0.2790	0.9830	0.1007
FF, No ICA, No CSP	No FF, ICA, CSP	0.2282	0.4622	0.0005	0.7413
FF, No ICA, No CSP	No FF, ICA, No CSP	0.2282	0.2556	0.9992	0.0585
FF, No ICA, No CSP	No FF, No ICA, CSP	0.2282	0.3970	0.0256	0.3727
FF, No ICA, No CSP	No FF, No ICA, No CSP	0.2282	0.2790	0.9671	0.1490
No FF, ICA, CSP	No FF, ICA, No CSP	0.4622	0.2556	0.0028	0.6583
No FF, ICA, CSP	No FF, No ICA, CSP	0.4622	0.3970	0.8852	0.0684
No FF, ICA, CSP	No FF, No ICA, No CSP	0.4622	0.2790	0.0114	0.5755
No FF, ICA, No CSP	No FF, No ICA, CSP	0.2556	0.3970	0.1022	0.2815
No FF, ICA, No CSP	No FF, No ICA, No CSP	0.2556	0.2790	0.9997	0.0298
No FF, No ICA, CSP	No FF, No ICA, No CSP	0.3970	0.2790	0.2675	0.2062

Appendix B

MACHINE LEARNING MODEL COMBINED MOTOR TASK TUKEY HSD RESULTS

Table B.1: All Task Accuracy Tukey HSD Comparison. ANOVA:[$p = 0.0803, \eta p^2 = 0.2263$]

A	B	mean(A)	mean(B)	p-tukey	eta-square
LDA	Linear SVM	0.5536	0.5651	0.9943	0.0400
LDA	MLP NN	0.5536	0.5712	0.9518	0.1046
LDA	Naive Bayes	0.5536	0.6012	0.1227	0.3255
LDA	RBF SVM	0.5536	0.5446	0.9985	0.0243
LDA	Random Forest	0.5536	0.5618	0.9991	0.0163
LDA	kNN	0.5536	0.5694	0.9707	0.1051
Linear SVM	MLP NN	0.5651	0.5712	0.9999	0.0092
Linear SVM	Naive Bayes	0.5651	0.6012	0.4013	0.1793
Linear SVM	RBF SVM	0.5651	0.5446	0.9021	0.0853
Linear SVM	Random Forest	0.5651	0.5618	1.0000	0.0020
Linear SVM	kNN	0.5651	0.5694	1.0000	0.0054
MLP NN	Naive Bayes	0.5712	0.6012	0.6162	0.1432
MLP NN	RBF SVM	0.5712	0.5446	0.7371	0.1509
MLP NN	Random Forest	0.5712	0.5618	0.9982	0.0178
MLP NN	kNN	0.5712	0.5694	1.0000	0.0010
Naive Bayes	RBF SVM	0.6012	0.5446	0.0371	0.3470
Naive Bayes	Random Forest	0.6012	0.5618	0.2994	0.1844
Naive Bayes	kNN	0.6012	0.5694	0.5531	0.1709
RBF SVM	Random Forest	0.5446	0.5618	0.9554	0.0521
RBF SVM	kNN	0.5446	0.5694	0.7921	0.1510
Random Forest	kNN	0.5618	0.5694	0.9994	0.0133

Table B.2: All Task F1 Score Tukey HSD Comparison. ANOVA:[$p = 0.0006, \eta p^2 = 0.4157$]

A	B	mean(A)	mean(B)	p-tukey	eta-square
LDA	Linear SVM	0.3033	0.3400	0.9884	0.0750
LDA	MLP NN	0.3033	0.3191	0.9999	0.0125
LDA	Naive Bayes	0.3033	0.4753	0.0171	0.4372
LDA	RBF SVM	0.3033	0.2410	0.8611	0.0576
LDA	Random Forest	0.3033	0.3577	0.9208	0.0971
LDA	kNN	0.3033	0.2584	0.9677	0.0867
Linear SVM	MLP NN	0.3400	0.3191	0.9995	0.0476
Linear SVM	Naive Bayes	0.3400	0.4753	0.1075	0.4043
Linear SVM	RBF SVM	0.3400	0.2410	0.4178	0.1576
Linear SVM	Random Forest	0.3400	0.3577	0.9998	0.0188
Linear SVM	kNN	0.3400	0.2584	0.6413	0.3961
MLP NN	Naive Bayes	0.3191	0.4753	0.0396	0.4469
MLP NN	RBF SVM	0.3191	0.2410	0.6867	0.0989
MLP NN	Random Forest	0.3191	0.3577	0.9848	0.0705
MLP NN	kNN	0.3191	0.2584	0.8742	0.2164
Naive Bayes	RBF SVM	0.4753	0.2410	0.0004	0.4265
Naive Bayes	Random Forest	0.4753	0.3577	0.2236	0.2669
Naive Bayes	kNN	0.4753	0.2584	0.0012	0.5980
RBF SVM	Random Forest	0.2410	0.3577	0.2311	0.1771
RBF SVM	kNN	0.2410	0.2584	0.9998	0.0052
Random Forest	kNN	0.3577	0.2584	0.4122	0.3186

Table B.3: All Task Precision Tukey HSD Comparison. ANOVA:[$p = 0.0022, \eta p^2 = 0.3729$]

A	B	mean(A)	mean(B)	p-tukey	eta-square
LDA	Linear SVM	0.3417	0.3786	0.9968	0.0705
LDA	MLP NN	0.3417	0.3496	1.0000	0.0031
LDA	Naive Bayes	0.3417	0.5698	0.0120	0.4635
LDA	RBF SVM	0.3417	0.3309	1.0000	0.0010
LDA	Random Forest	0.3417	0.3930	0.9815	0.0868
LDA	kNN	0.3417	0.2906	0.9821	0.1040
Linear SVM	MLP NN	0.3786	0.3496	0.9992	0.0877
Linear SVM	Naive Bayes	0.3786	0.5698	0.0553	0.4303
Linear SVM	RBF SVM	0.3786	0.3309	0.9874	0.0210
Linear SVM	Random Forest	0.3786	0.3930	1.0000	0.0129
Linear SVM	kNN	0.3786	0.2906	0.7966	0.4217
MLP NN	Naive Bayes	0.3496	0.5698	0.0170	0.4872
MLP NN	RBF SVM	0.3496	0.3309	0.9999	0.0032
MLP NN	Random Forest	0.3496	0.3930	0.9923	0.0923
MLP NN	kNN	0.3496	0.2906	0.9633	0.2084
Naive Bayes	RBF SVM	0.5698	0.3309	0.0074	0.2781
Naive Bayes	Random Forest	0.5698	0.3930	0.0944	0.3493
Naive Bayes	kNN	0.5698	0.2906	0.0011	0.5956
RBF SVM	Random Forest	0.3309	0.3930	0.9531	0.0323
RBF SVM	kNN	0.3309	0.2906	0.9949	0.0145
Random Forest	kNN	0.3930	0.2906	0.6614	0.3395

Table B.4: All Task Precision Tukey HSD Comparison. ANOVA:[$p = 0.0002, \eta p^2 = 0.4450$]

A	B	mean(A)	mean(B)	p-tukey	eta-square
LDA	Linear SVM	0.3014	0.3400	0.9776	0.0710
LDA	MLP NN	0.3014	0.3197	0.9996	0.0139
LDA	Naive Bayes	0.3014	0.4448	0.0421	0.3792
LDA	RBF SVM	0.3014	0.2086	0.4001	0.1506
LDA	Random Forest	0.3014	0.3578	0.8716	0.0859
LDA	kNN	0.3014	0.2536	0.9376	0.0864
Linear SVM	MLP NN	0.3400	0.3197	0.9993	0.0388
Linear SVM	Naive Bayes	0.3400	0.4448	0.2616	0.3559
Linear SVM	RBF SVM	0.3400	0.2086	0.0790	0.3315
Linear SVM	Random Forest	0.3400	0.3578	0.9997	0.0156
Linear SVM	kNN	0.3400	0.2536	0.4873	0.4192
MLP NN	Naive Bayes	0.3197	0.4448	0.1083	0.3914
MLP NN	RBF SVM	0.3197	0.2086	0.2020	0.2395
MLP NN	Random Forest	0.3197	0.3578	0.9790	0.0558
MLP NN	kNN	0.3197	0.2536	0.7660	0.2285
Naive Bayes	RBF SVM	0.4448	0.2086	0.0001	0.5325
Naive Bayes	Random Forest	0.4448	0.3578	0.4793	0.1807
Naive Bayes	kNN	0.4448	0.2536	0.0023	0.5987
RBF SVM	Random Forest	0.2086	0.3578	0.0303	0.3117
RBF SVM	kNN	0.2086	0.2536	0.9522	0.0490
Random Forest	kNN	0.3578	0.2536	0.2675	0.3052

Appendix C

PREPROCESSING TECHNIQUE INDIVIDUAL MOTOR TASK BOX PLOTS

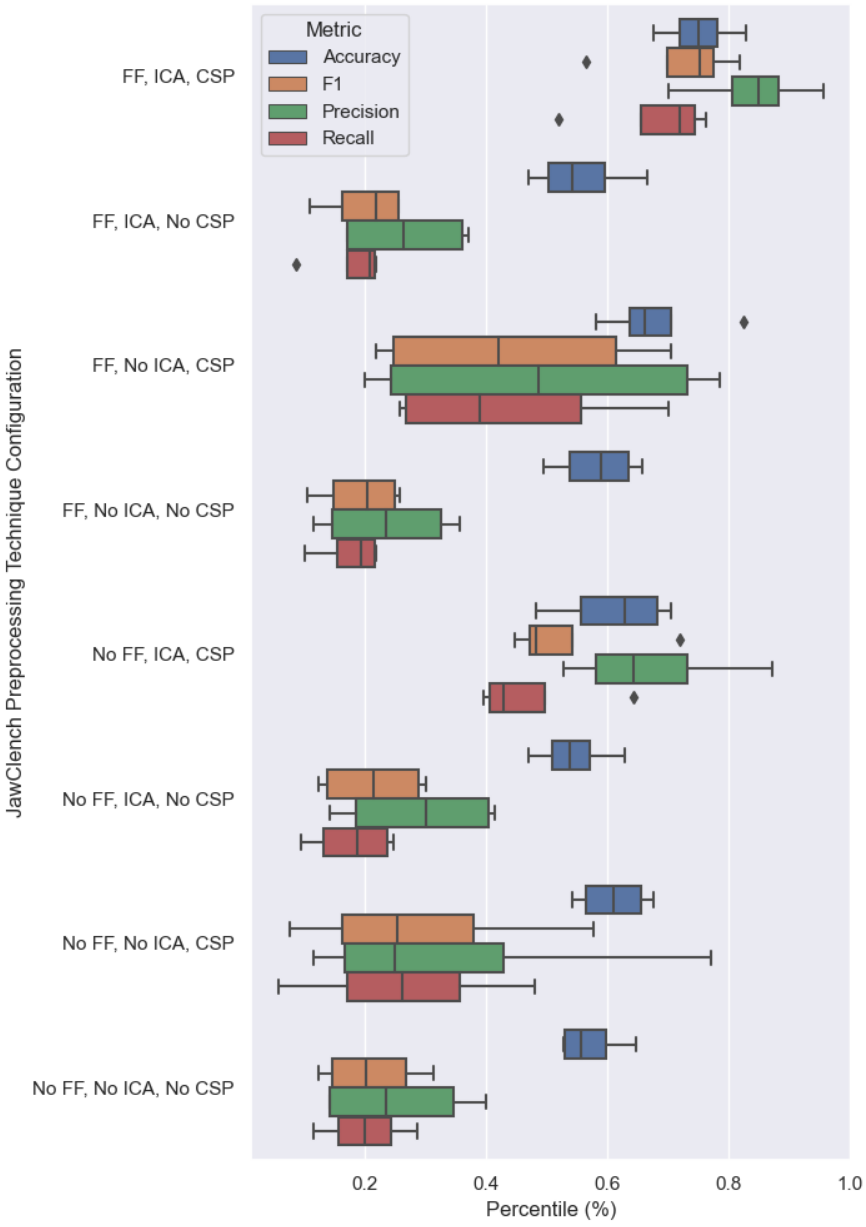

Figure C.1: Jaw Clench Preprocessing Technique Configuration Metric Box Plot

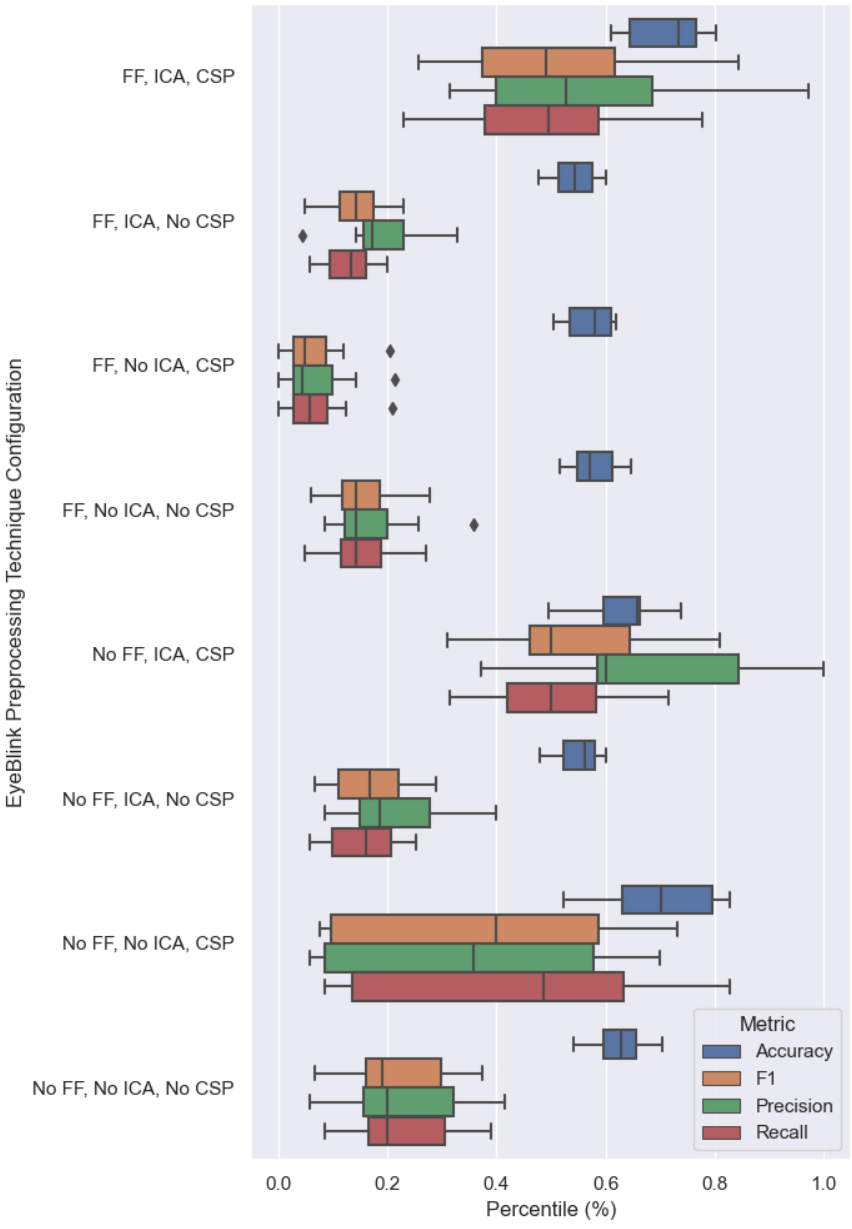

Figure C.2: Eye Blink Preprocessing Technique Configuration Metric Box Plot

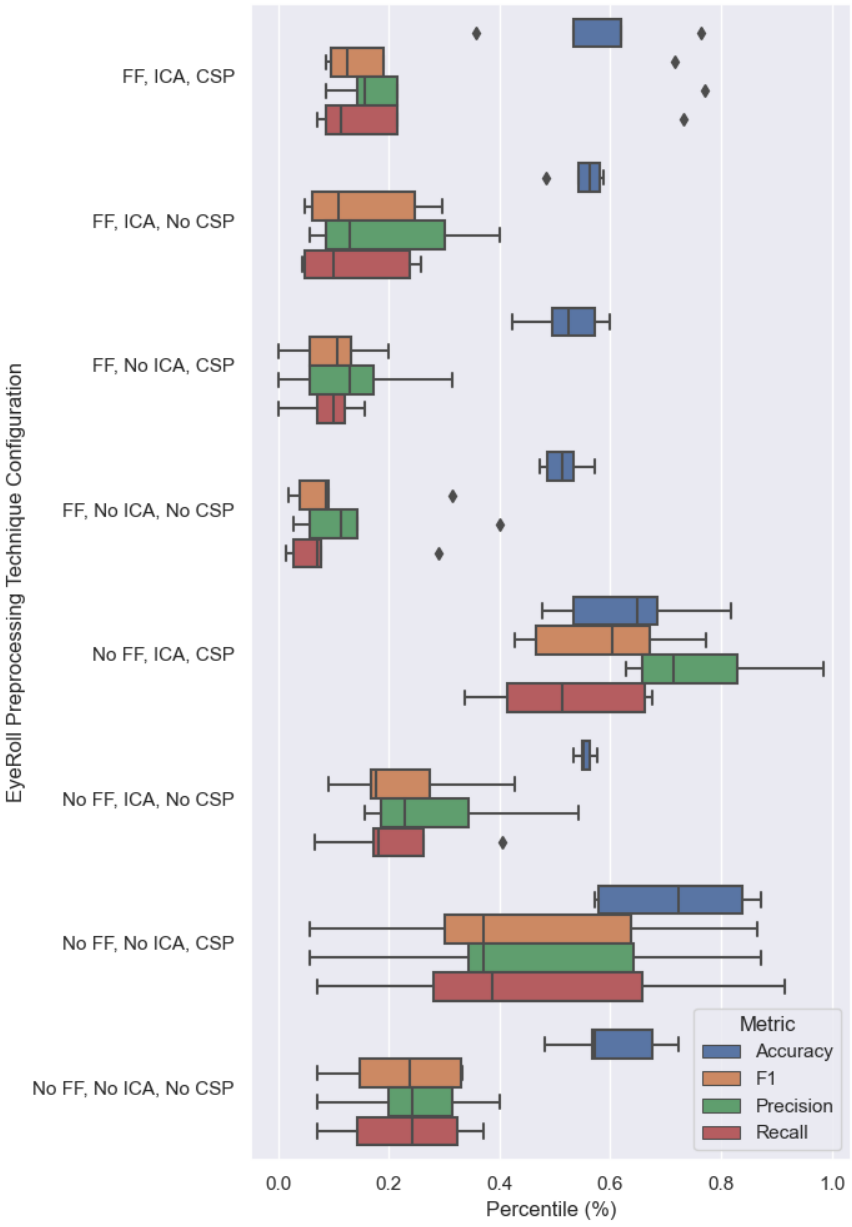

Figure C.3: Eye Roll Preprocessing Technique Configuration Metric Box Plot

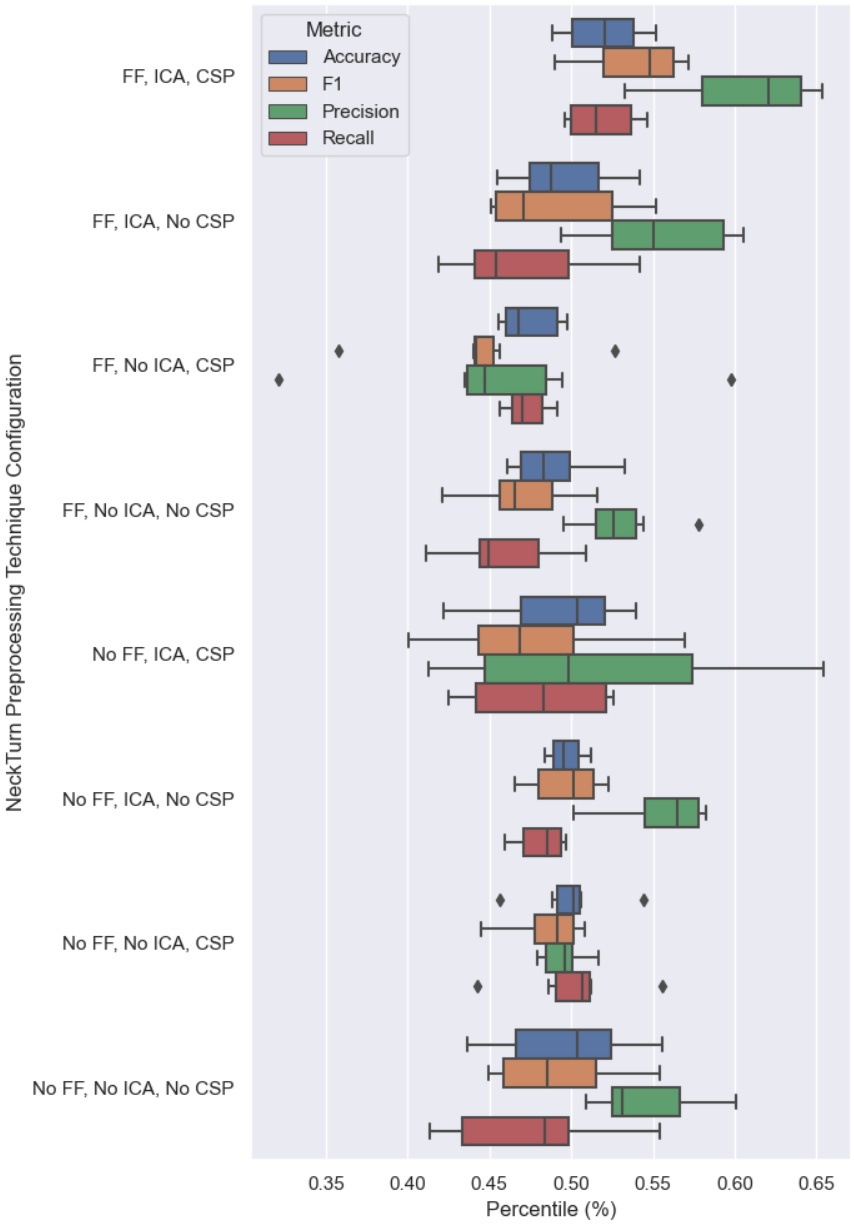

Figure C.4: Neck Turn Preprocessing Technique Configuration Metric Box Plot

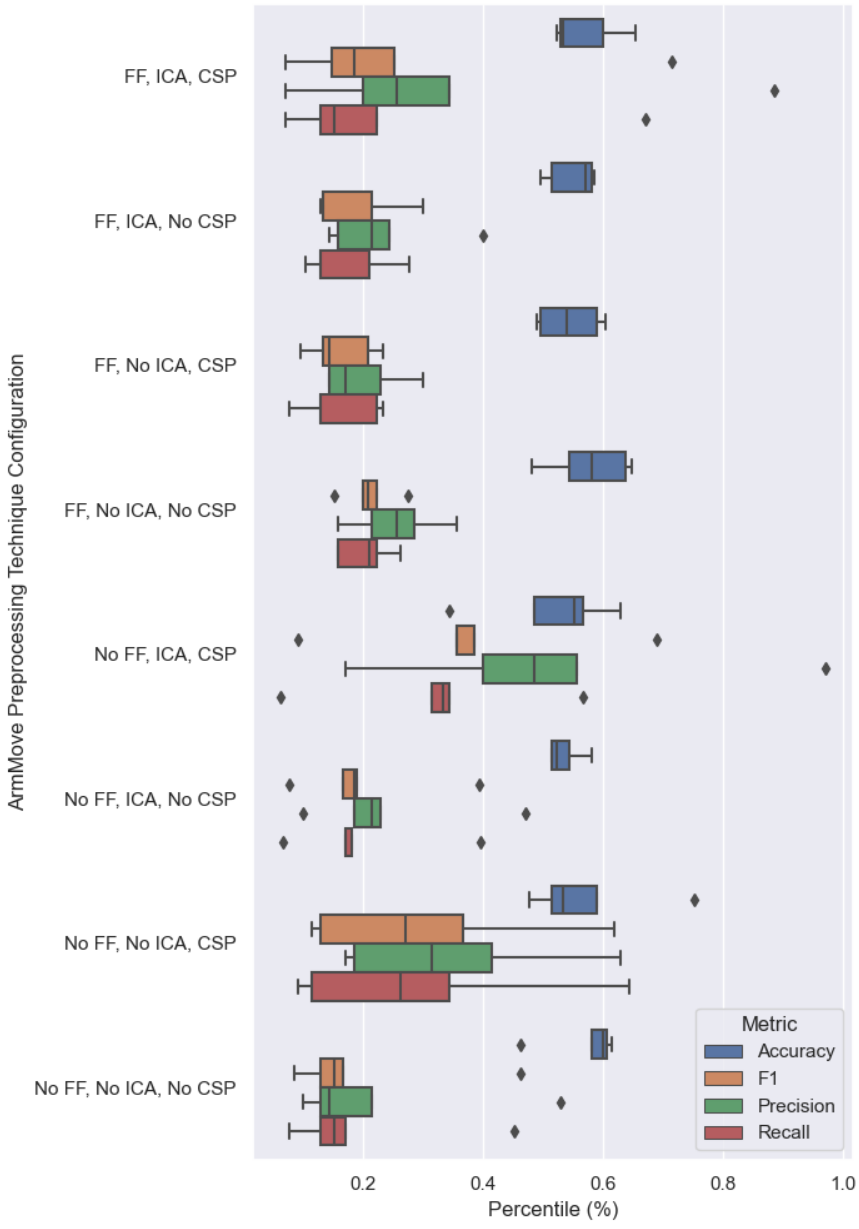

Figure C.5: Arm Movement Preprocessing Technique Configuration Metric Box Plot

Appendix D

MACHINE LEARNING MODEL INDIVIDUAL MOTOR TASK BOX PLOTS

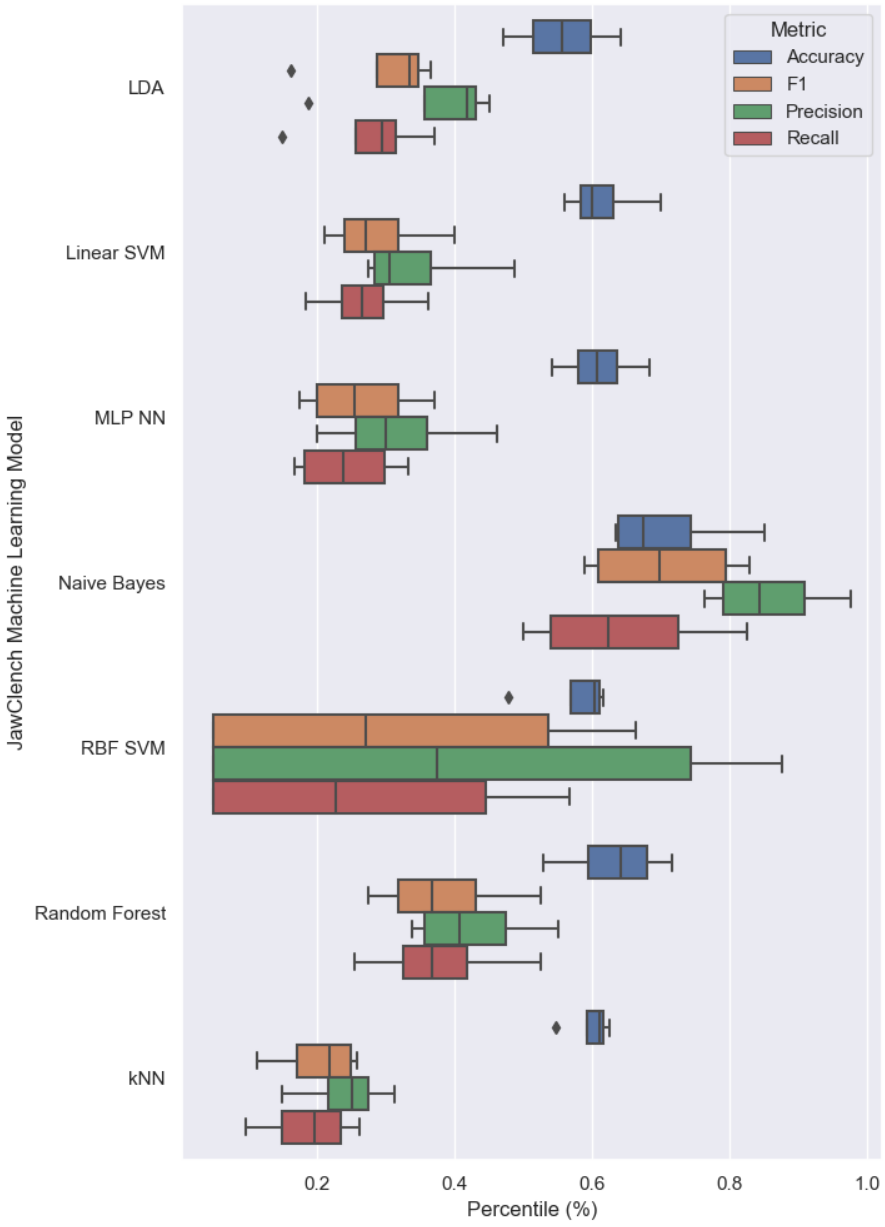

Figure D.1: Jaw Clench Machine Learning Model Metric Box Plot

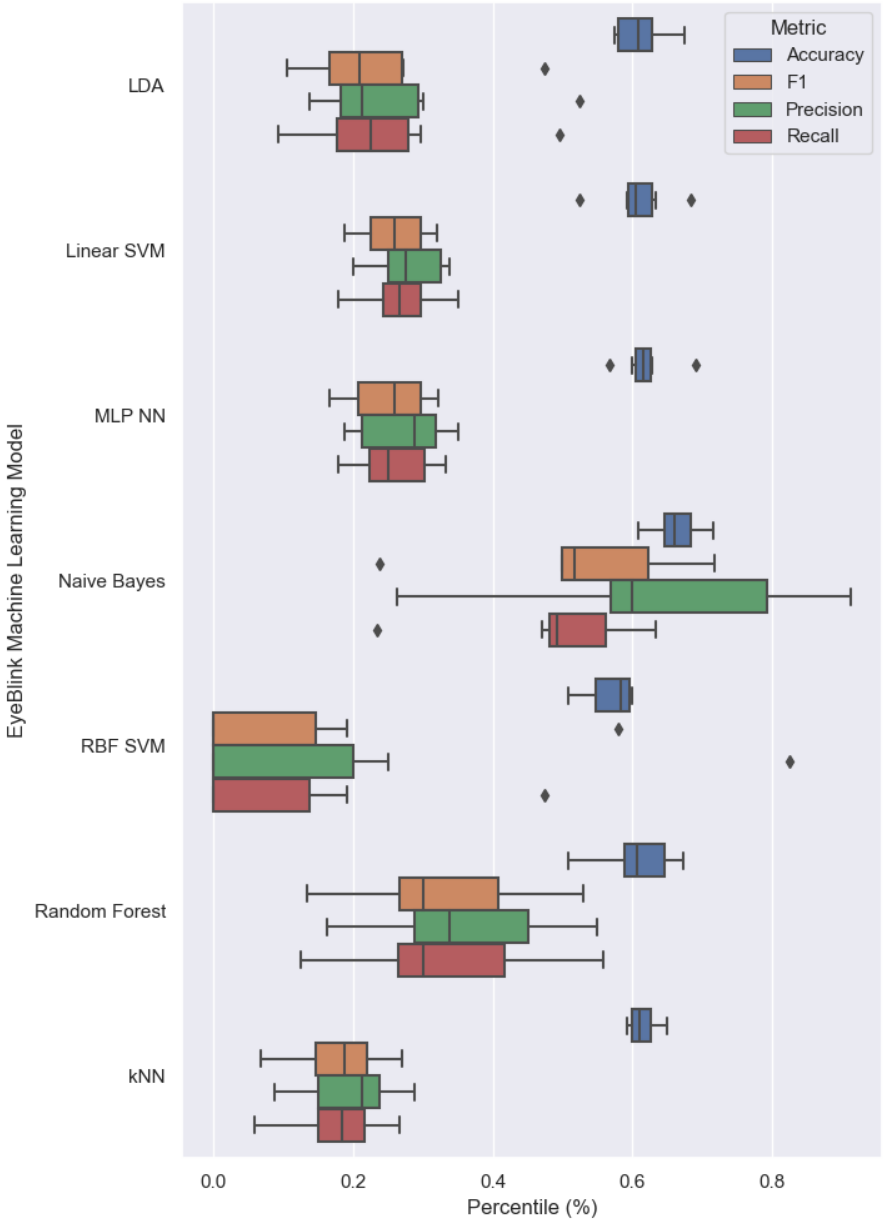

Figure D.2: Eye Blink Machine Learning Model Metric Box Plot

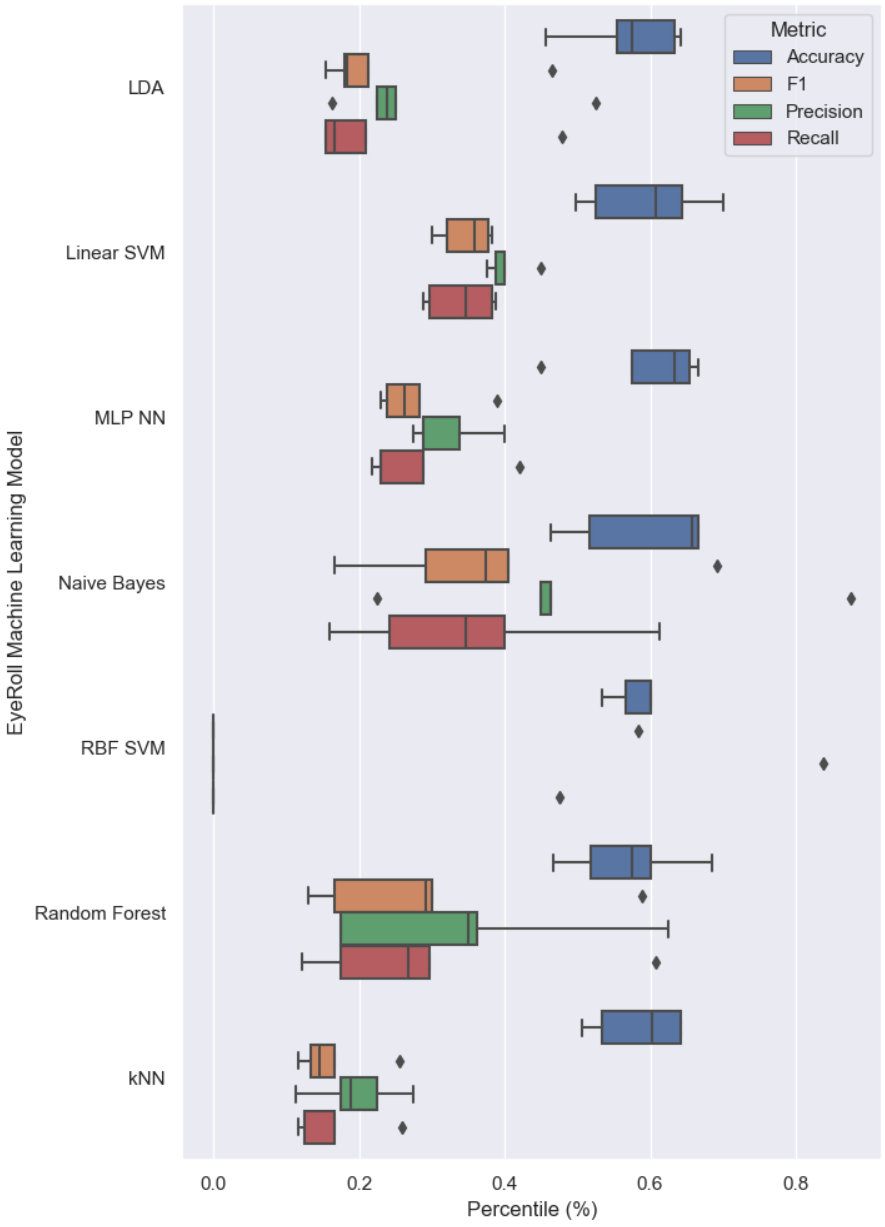

Figure D.3: Eye Roll Machine Learning Model Metric Box Plot

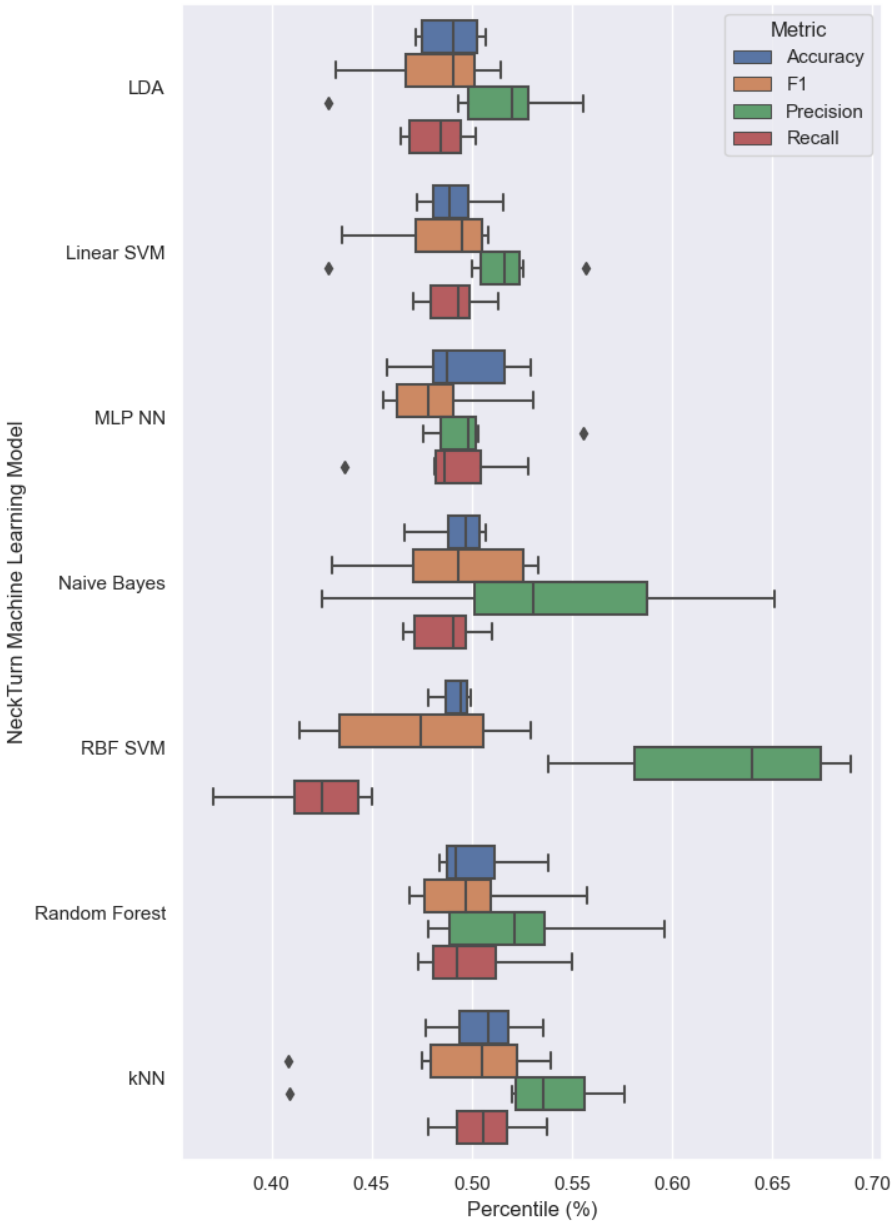

Figure D.4: Neck Turn Machine Learning Model Metric Box Plot

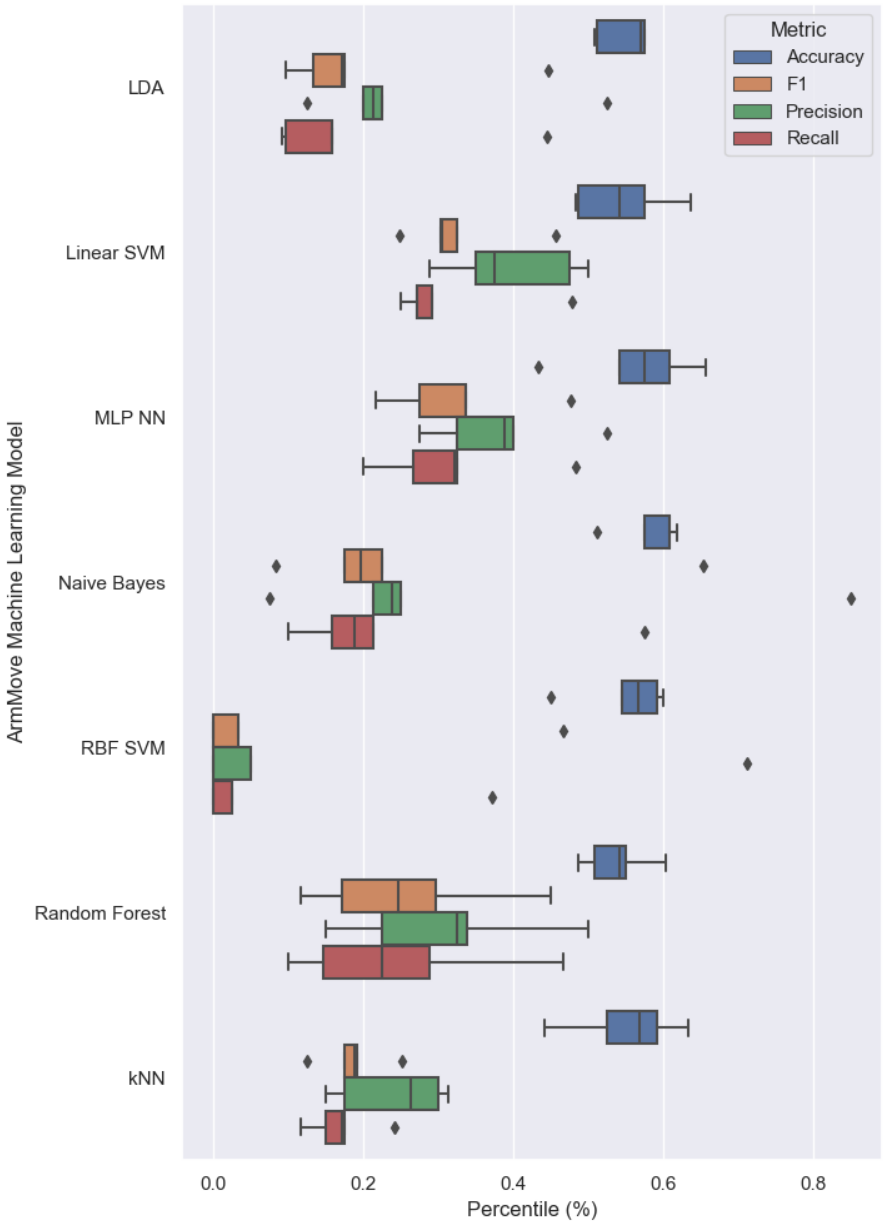

Figure D.5: Arm Movement Machine Learning Model Metric Box Plot

Appendix E

DSI STREAMER

E.1 Panels

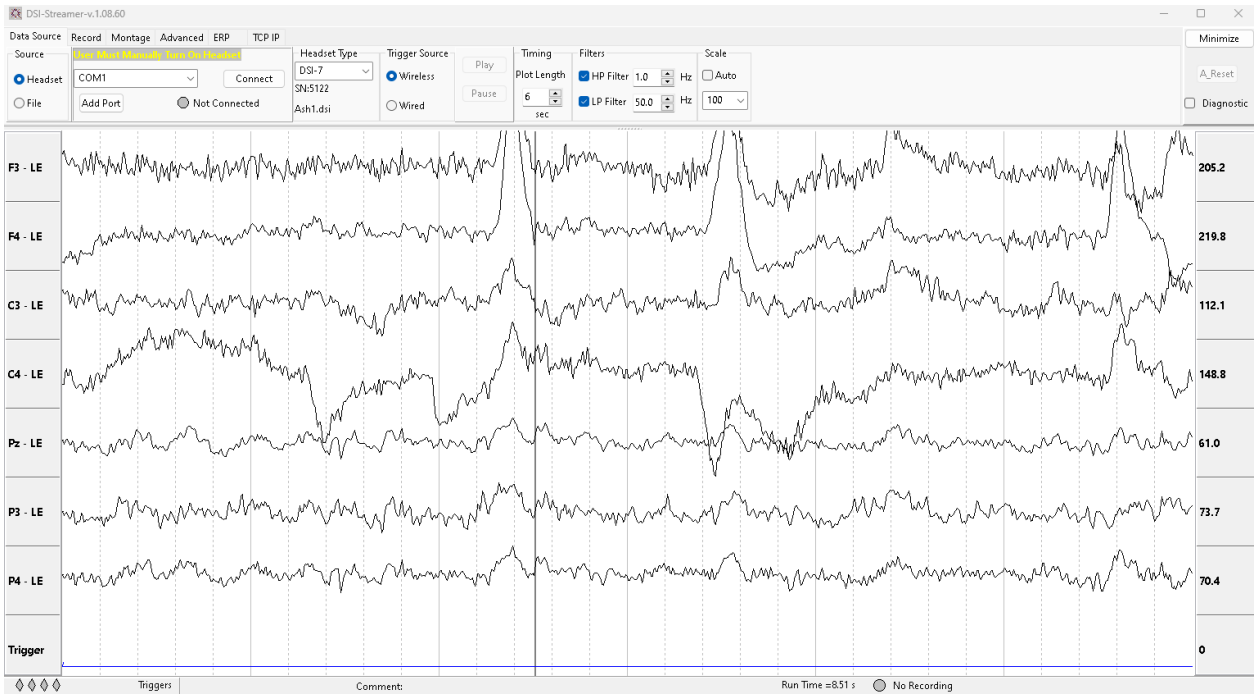

Figure E.1: Data Source Panel in Streamer. Used to connect to DSI7 EEG Headset.

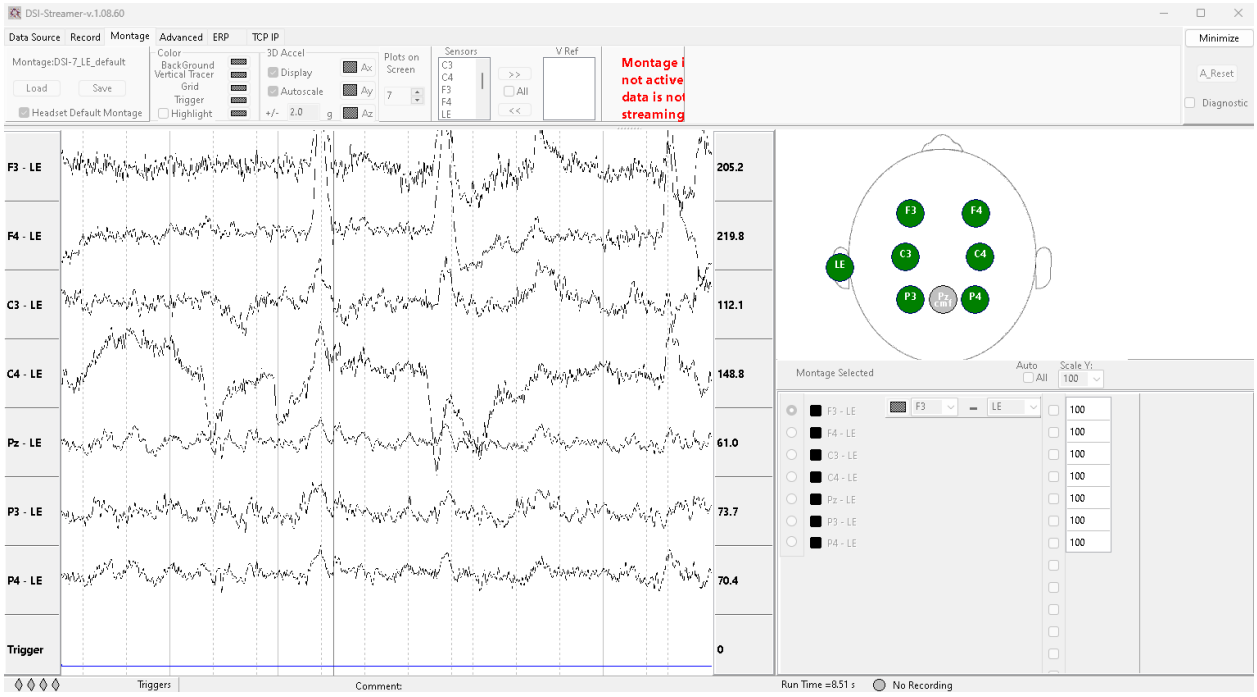

Figure E.2: Montage Panel in DSI Streamer. Used to calibrate each electrode on the DSI7 EEG Headset.

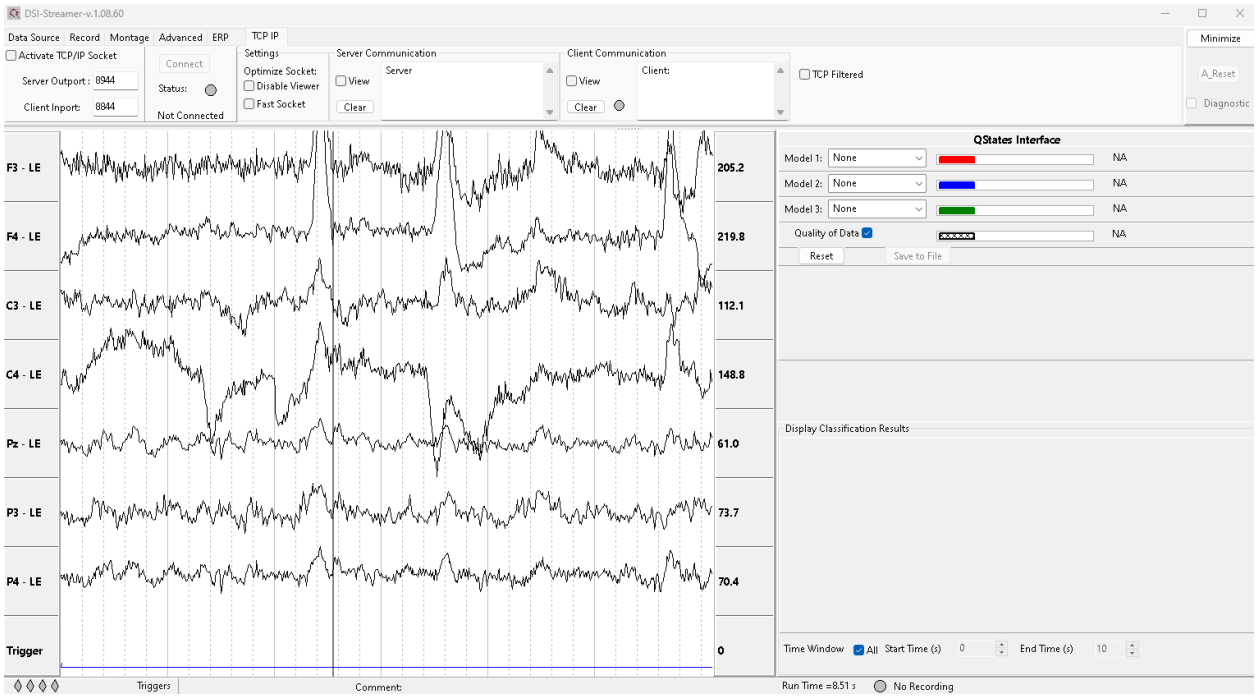

Figure E.3: TCP/IP Panel in DSI Streamer. Used to create the TCP socket that the data collection python program retrieves data from.

E.2 Known Bugs

E.2.1 TCP/IP Program Slowdown/Speedup

During data collection a critical bug was discovered in DSI Streamer. This bug involved the order in which data collection and the TCP/IP Socket was setup. It is considered critical to this work because it negatively impacted data collection and resulted in loss of data.

This bug was recreated on three different systems and is recreatable at time of writing. This bug causes the program run at approximately 71% normal speed. With data experiments set to 54 seconds each, this resulted in only 41 seconds being recorded until the bug was discovered. This impacted almost 50 datasets resulting in the loss of about 13 minutes of collected data. Interestingly, the opposite problem can occur as well. If the TCP/IP socket is turned on first, then disabled while data is being collected, it causes the program to begin running exceedingly fast. At least 3x its normal operation speed. When replaying a 54 second file under this condition, it results in the file being replayed in approximately 11 seconds.

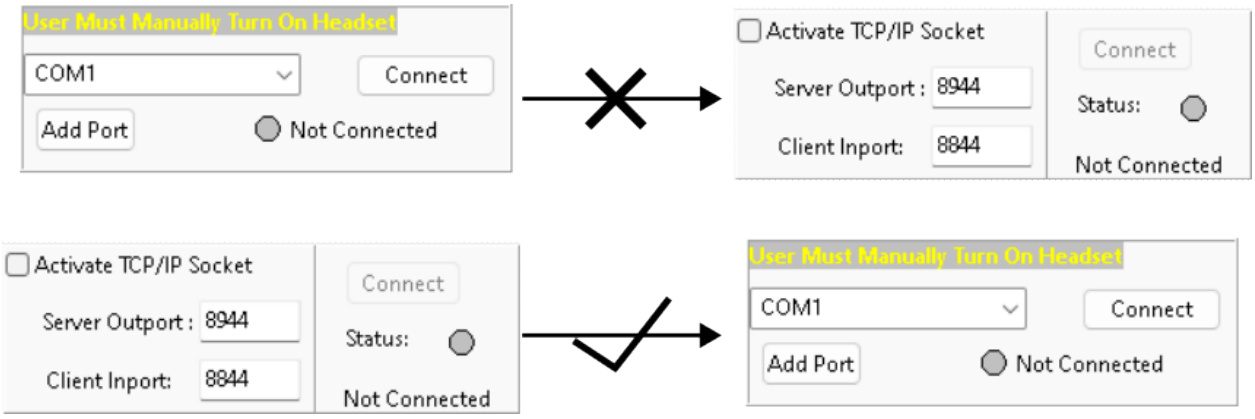

Figure E.4: How to avoid the TCP/IP Slowdown Bug. Do not disable TCP/IP during Data collection either.

Cause In DSI Streamer, if the headset is connected to or a file is replayed before the TCP/IP socket is opened, it causes the DSI Streamer program to immediately slow down. This slow down causes not only the timer to slow down, it also decreases the speed at which data is transmitted over the TCP/IP socket.

Remedy Open the TCP/IP socket before connecting to the headset or replaying a file. Alternatively, if the bug has already been initiated, disabling the TCP/IP socket also fixes the bug.