EEG-Based Estimation of Human Reaction Time Corresponding to Change of Visual Event.

by

Mohammad Samin Nur Chowdhury

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved October 2019 by the
Graduate Supervisory Committee:

Daniel W. Bliss, Chair
Antonia Papandreou-Suppappola
Gene Brewer

ARIZONA STATE UNIVERSITY

December 2019

ABSTRACT

The human brain controls a person's actions and reactions. In this study, the main objective is to quantify reaction time towards a change of visual event and figuring out the inherent relationship between response time and corresponding brain activities. Furthermore, which parts of the human brain are responsible for the reaction time is also of interest. As electroencephalogram (EEG) signals are proportional to the change of brain functionalities with time, EEG signals from different locations of the brain are used as indicators of brain activities. As the different channels are from different parts of our brain, identifying most relevant channels can provide the idea of responsible brain locations. In this study, response time is estimated using EEG signal features from time, frequency and time-frequency domain. Regression-based estimation using the full data-set results in RMSE (Root Mean Square Error) of 99.5 milliseconds and a correlation value of 0.57. However, the addition of non-EEG features with the existing features gives RMSE of 101.7 ms and a correlation value of 0.58. Using the same analysis with a custom data-set provides RMSE of 135.7 milliseconds and a correlation value of 0.69. Classification-based estimation provides 79% & 72% of accuracy for binary and 3-class classification respectively. Classification of extremes (high-low) results in 95% of accuracy. Combining recursive feature elimination, tree-based feature importance, and mutual feature information method, important channels, and features are isolated based on the best result. As human response time is not solely dependent on brain activities, it requires additional information about the subject to improve the reaction time estimation.

ACKNOWLEDGMENTS

My heartfelt gratitude to Dr. Daniel W. Bliss for his guidance, encouragement, and patience with my snail-paced progress. I can't thank Dr. Antonia Papandreou-Suppappola and Dr. Gene Brewer enough for their precious time and valuable insights on my thesis. I would also like to thank Dr. Arindam Dutta for his consistent assistance throughout the process.

TABLE OF CONTENTS

CHAPTER Page

CHAPTER Page

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

Figure Page

Figure Page

Chapter 1

INTRODUCTION

It is known that different parts of the brain are responsible for different human behaviors and a person's response to a particular event directly depends on how the brain is functioning. A lot of previous psychological experiments on human behavior have been conducted to establish a relationship between human action and brain activities. However, as the brain is one of the largest and most complicated parts of the human body with uncountable functions going on continuously, there is a long way to go to unfold all the mysteries in this field. This study involves a specific form of human reaction and electroencephalogram (EEG) signals from different parts of the brain are used here as the representatives of brain states.

This study is based on an experiment conducted to observe human responses based on a change of visual event. A computer screen was used where a plus sign (+) was shown at the center and after variable time duration, the sign was changed into a cross. This action was repeated several times for each subject and every single repetition is considered as a trial. The number of repetitions is also variable. The task of every subject was to press the space bar as soon as the sign was changed from plus (+) to cross. The EEG signal used for response estimation is considered from the beginning of the action to the time of sign-change. On the other hand, from the time of sign-change to the time of key-press is the response time of the person. Figure-1.1 shows how the experiment is working.

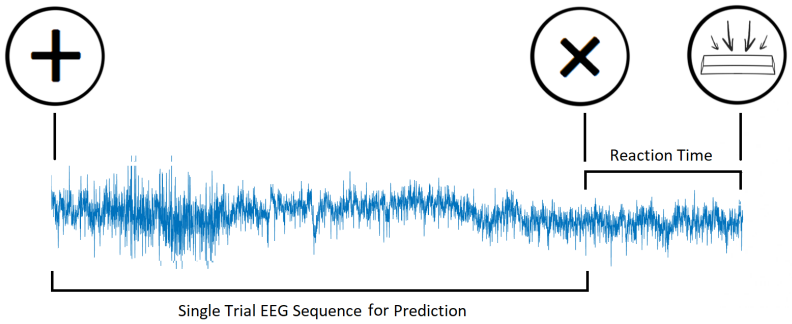


Figure 1.1: Experiment Demonstration

The goal of this study is to observe how fast a person reacts to this change and how it depends on different psychological parameters using EEG signals. A feature extraction-based approach is taken here, where features from time, frequency and time-frequency domain are extracted from different EEG channels to understand the brain activities during the relevant time frames. Further, different machine learning algorithms are taken into account to establish the relation between EEG signal and reaction time. Based on the relationship, channels (brain locations) and features that are highly related to the response are separated.

1.1 Organization

The whole study is divided into 11 chapters. The basic idea of this study is already expressed in chapter 1. Chapter 2 covers the background information relevant to the study. It includes the basics of EEG signal, human response time and prior works related to this experiment. The structure and organization of the data and the necessary pre-processing is discussed in chapter 3. Before working with the data, we need to observe the data first. Visualization of the EEG data in time, frequency and time-frequency domain are covered in chapter 4. This chapter also includes individual subject analysis. Chapter 5 contains information about feature extraction in all the

stated domains. In chapter 6 a regression-based approach is shown to predict the response time using extracted features and evaluate the performance. Using the same features, we now consider the problem in hand as a classification problem and chapter 7 discusses different algorithms and their performances for classification. Both binary and 3-class classification are considered here with an additional high-low response-based classification. Chapter 8 demonstrates an attempt to figure out which EEG channels and features are of interest in our study. Possible future extensions to this study are provided in chapter 9. Chapter 10 brings an end to our study and chapter 11 shows all the reference works that were taken into account.

Chapter 2

BACKGROUND

2.1 EEG Signal

Electroencephalography (EEG) is a method of electrophysiological monitoring to record the electrical activity of the brain. Although invasive electrodes are sometimes used, as in electrocorticography, it is usually noninvasive, with the electrodes placed along the scalp. Voltage fluctuations resulting from ionic current within the brain neurons are measured by EEG. Clinically, EEG refers to the recording of the brain's spontaneous electrical activity over a certain period. The recording is from multiple electrodes placed on the scalp. Diagnostic applications are focused either on event-related potentials or the spectral content of EEG. The first one investigates potential fluctuations time-locked to an event, such as 'button press' or 'stimulus onset'. The latter analyses the type of neural oscillations known as "brain waves" that can be observed in the frequency domain representation of EEG signals.

Though the spatial resolution is limited, EEG is a valuable tool for research and diagnosis. It is one of the very few mobile techniques available that offers millisecond-range temporal resolution which is not possible with CT, PET or MRI. In scalp based conventional EEG, the recording is obtained by placing electrodes on the scalp with a conductive gel, usually after preparing the scalp area by light abrasion to reduce impedance due to dead skin cells. Most of the systems usually use electrodes, each of which is connected to a separate wire. In some systems, electrodes are embedded into caps or nets; this is particularly useful when high-density arrays of electrodes are

needed.

Each electrode is connected to one input of a differential amplifier (one amplifier per pair of electrodes) and a common reference electrode is connected to the other input of each differential amplifier. These amplifiers amplify the voltage between the active electrode and the reference (typically 1,000100,000 times). Most EEG systems these days are digital, and the amplified signal is digitized via an analog-to-digital converter (ADC), after being passed through an anti-aliasing filter. Analog-to-digital sampling can go up to 20 kHz for some research applications but typically it is 256512 Hz in clinical scalp EEG. Opensource hardware such as OpenBCI can capture EEG signal and this signal can be processed by freely available EEG software such as EEGLAB or the Neurophysiological Biomarker Toolbox.

Typically, an adult human EEG signal is around 10 V to 100 V in amplitude when measured from the scalp and is about 1020 mV when measured from subdural electrodes. Since an EEG voltage signal represents a difference between the voltages at two electrodes, the display of the EEG for the reading of encephalography may be set up in one of several ways. The representation of the EEG channels is referred to as a montage.

1. Sequential montage
2. Referential montage
3. Average reference montage
4. Laplacian montage

2.1.1 Channels

An EEG channel is an electrode capturing brainwave activity. EEG systems can have as few as a single channel to as many as 256 channels typically. Electrode placement on the head maintains a formal standard known as the 10/20 system or International 10/20 system. This system uses the distance from the bridge of the nose to the lowest point of the skull from the back of the head as a reference distance for a given person's head size. Then the electrodes are separated from each other either by 10% or 20% of the reference distance. When an greater number of channels is required, the 10/20 system is modified where the electrodes are separated by 10% of the reference distance (10/10). Further resolution of %5 separation (10/5) distances adds even more electrodes to the network cluster. Figure-2.1 shows sample EEG channel locations.

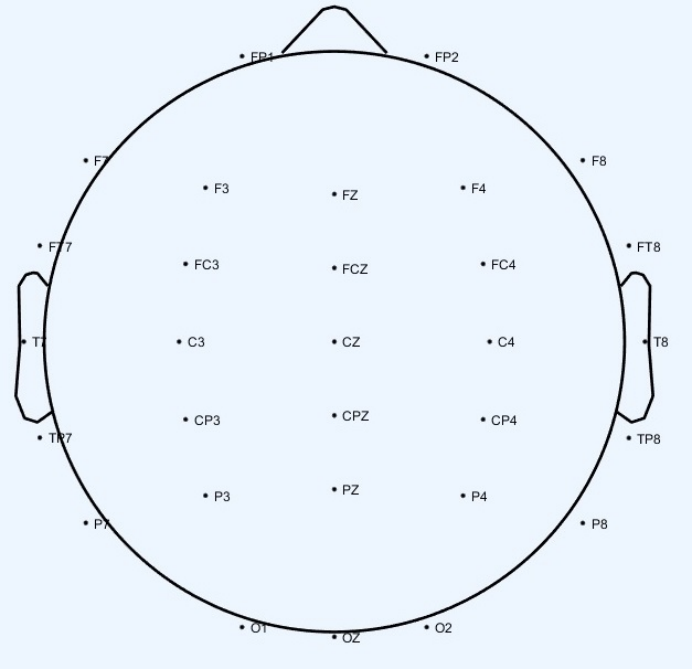


Figure 2.1: EEG Channel Locations

2.2 Human Reaction Time

Reaction time or response time usually refers to the amount of time that takes place between when a person perceives something to when that person responds to it. It is the combined ability to detect, process, and respond to a stimulus. Reaction time is a combination of the following factors:

1. Perception: Seeing, hearing, or feeling a stimulus with certainty is very important for good reaction time.

2. Processing: To have good reaction time, it's necessary to be highly focused and understand the information quickly.

3. Response: Motor agility is necessary to be able to act fast and have a good response time.

If any part of these processes is altered, reaction time will be affected consequently. Reaction time includes a motor component, unlike processing speed. Therefore, having good reaction time is associated with having good reflexes.

Reaction time can vary depending on a variety of factors:

1. The complexity of the stimulus: The more complex the stimulus, the more information that has to be processed, the longer this process will take.

2. Familiarity and expectations: If a stimulus to respond is known and was responded before, the reaction time will be lower. The less information that one needs to process, the quicker the reaction time will be.

3. The current state of the organism: Some factors that may negatively affect the detection of the stimulus are attention, fatigue, high temperature, old age, or

even eating too much food or substances like alcohol or other drugs. All of these factors may negatively affect the detection of the stimulus, processing it, and responding to it.

4. Stimulated sensory modality: Reaction time is shorter when the stimulus that triggers the response is auditory than if it is visual because auditory stimuli require less processing. Each sensory modality has a different reaction time.

Aside from other factors, the type of stimulus that we process also affects reaction time:

1. Simple: There is one single response to a single stimulus. For example, pressing a specific key on the computer when a word appears.

2. Choice: There are different responses to different stimuli. For example, pressing the right arrow key if a word appears in Spanish, and pressing the left arrow key if the word appears in another language.

3. Selection: There are different stimuli, but you only have to respond to one. For example, press the space bar only when the word appears in English. If it appears in Spanish, you don't do anything.

Good reaction time allows a person to be agile and efficient when it comes to responding to stimuli and situations like having a conversation, driving, playing sports, etc. Luckily, reaction time can be trained and improved. For this study, we are dealing with a simple visual event.

2.3 Prior Works

Some works tried to analyze brain signals corresponding to visual events and evaluate the relationships. EEG frequency and reaction time-based sequential analysis

was performed in one of the prior studies to establish a relationship between speed of response and background frequency (Morrell, 1966). Another study demonstrates an experiment to monitor EEG signal and reaction time during a normoxic saturation dive. Three of the Navy divers were exposed to a normoxic breathing mixture at a certain pressure for seven consecutive days. Visual evoked responses (VERs), electroencephalograms (EEGs) and simple reaction time (RT) were measured for all. The outcomes of the experiment indicate changes in the amplitude of both alpha and theta with a decrease in alpha frequency (Mckay *et al.*, 1977). One of the studies tried to apply a method for temporally extracting stimulus- and response-locked components of brain activity from human scalp electroencephalography (EEG) during an auditory simple reaction time task (Takeda *et al.*, 2008). A power spectrum based analysis was performed using ICA, PCA to estimate visual attention from the EEG signal (Ahirwal and Londhe, 2012). Twenty-six pre-elite table tennis players in Taiwan were recruited to perform a cued reaction time task to observe the relation between reaction time and EEG signal (Cheng *et al.*, 2010).

However, our study tries to predict the reaction time of a person using a single trial EEG signal. A few of the works tried to predict response time based on EEG (Luo and Sajda, 2006; Wu *et al.*, 2017). Most of these approaches only considered the regression-based analysis and didn't use a large enough data-set for reliable outcomes. Also, most of the studies were subject-specific and didn't produce a generalized model. In general, not much work has been done yet in this exact field.

Chapter 3

DATA HANDLING AND PREPROCESSING

3.1 Data Structure

- As the artifacts created by eye blinks cause a problem in reading and processing the EEG data, no-blinks data-set was used in this study where the artifacts are already taken care of. Figure-3.1 shows how the no-blinks data-set is represented in EEGLAB.

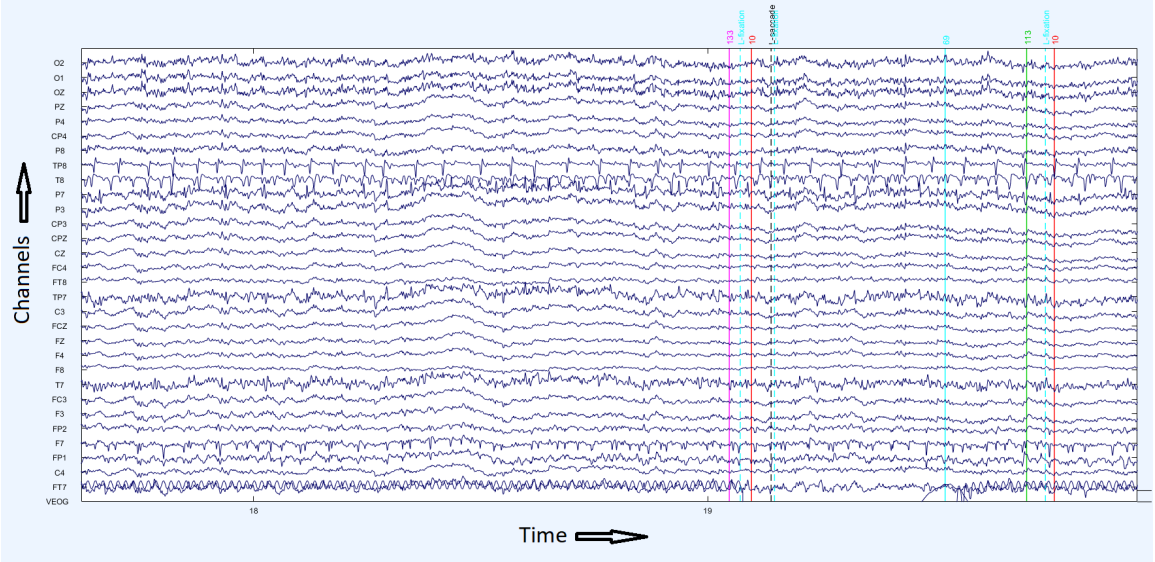


Figure 3.1: No Blinks EEG Data for All the Channels

- The experiment was conducted on 48 subjects.
- For each subject, the duration of the experiment was around 30 minutes.
- EEG signal was captured at a sampling rate of 1000 samples/second.
- For each subject, the number of data points in any channel is around 30*60*1000=1,800,000.

- EEG signals were obtained from 30 effective channels.

3.2 Normalization

Normalization is a standard procedure while dealing with EEG signals. It enables the post-processing to be balanced for all the channels. The data in hand is normalized as follows:

1. Mean- The mean value of all the data points in a channel was subtracted from each of the points.

2. Variance- All the data points of a particular channel were divided by the standard deviation (square root of variance) of that channel's data points. So, we have zero mean and unit variance data for all the channels.

3.3 Important Data Isolation

For each subject, the same experiment has been conducted multiple times and there are several events. The events are:

1. staring point

2. beginning of the wait interval

3. change of sign

4. response of the subject

5. endpoint

- For the purpose of analysis, we need to isolate data for every single trial in a certain way.

- Between the starting point and endpoint, the other three events repeat several times depending on how many times the experiment has been repeated.
- For each experiment:
    1. beginning of wait interval to change of sign = observation time of the subject
    2. change of sign to the response of the subject= response time of the subject
- EEG data for each channel is separated for all the subjects and all the trials based on both observation time and response time.
- So, now for each channel, we have:
  Number of observation sequences or response sequences = Number of trials for each subject

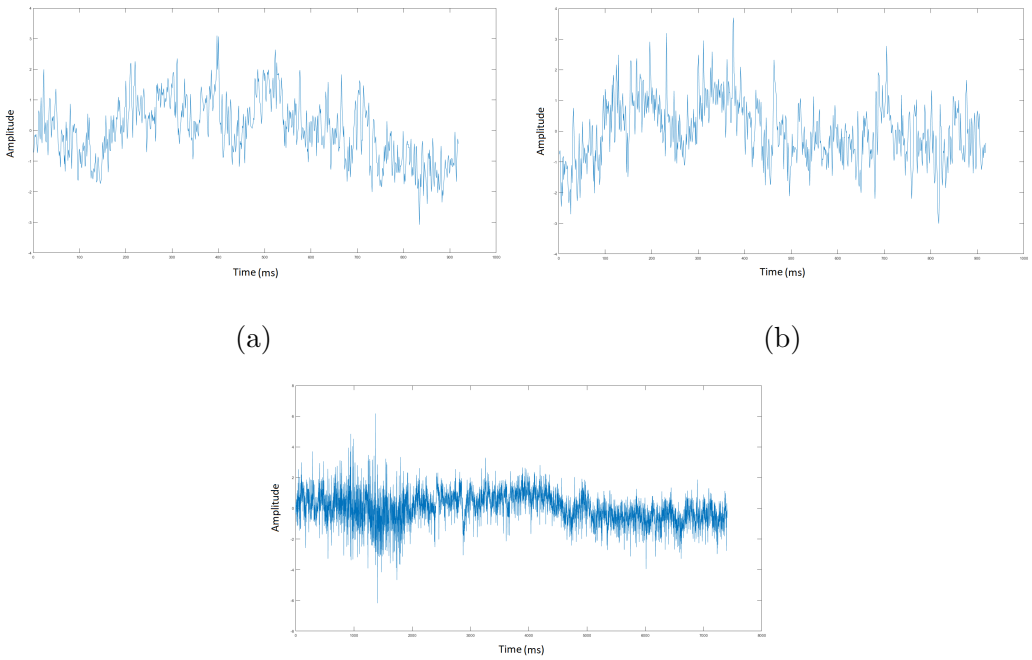
The pre-processing is done and now from the Observation sequences we can extract features and use them to establish desired relations with the response sequences.

Chapter 4

VISUALIZATION OF DATA IN DIFFERENT DOMAINS

4.1 Time Domain Visualization

The observation sequences in hand can be visualized in the time domain. The time domain representations of the sequences vary in length as the observation period changes in each trial. However, the range of this period is from 919 ms to 8908 ms. Figure-4.1 demonstrates sample time domain representations.



(a) (b)

(c)

Figure 4.1: Time Domain Representation of EEG Signal for (a) Subject-1, Channel-1, Trial-1, (b) Subject-28, Channel-6, Trial-5, (c) Subject-45, Channel-22, Trial-10.

4.2 Frequency Domain Visualization

Taking the Fourier transform of the time domain sequence gives us the frequency domain observation. Frequency domain representation is very important as the different frequency band properties can be of good use while categorizing the sequences to establish a relationship with the response time.

As real-time data is discrete in both time and frequency domain, discrete Fourier transform (DFT) gives us the desired spectral representation. Equation (4.1) shows the basic computation of DFT.

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{\frac{-j2\pi kn}{N}} \tag{4.1}$$

Where, $X[k]$ and $x[n]$ are frequency and time domain sequences respectively. k is used to denote the frequency domain ordinal, and n is used to represent the time-domain ordinal. N is the length of the sequence to be transformed. However, to reduce the computational complexity, fast Fourier transform (FFT) was used.

For any frequency domain analysis, the frequencies of less than half the sampling frequency are of interest. In our case, as the sampling frequency is 1000 Hz, we are taking all the frequency bins less than 500 Hz into account. However, it is seen that all the bins over 150 Hz are almost zero. So, we are using frequency domain representation up to 150 Hz. Figure-4.2 demonstrates sample frequency domain representations.

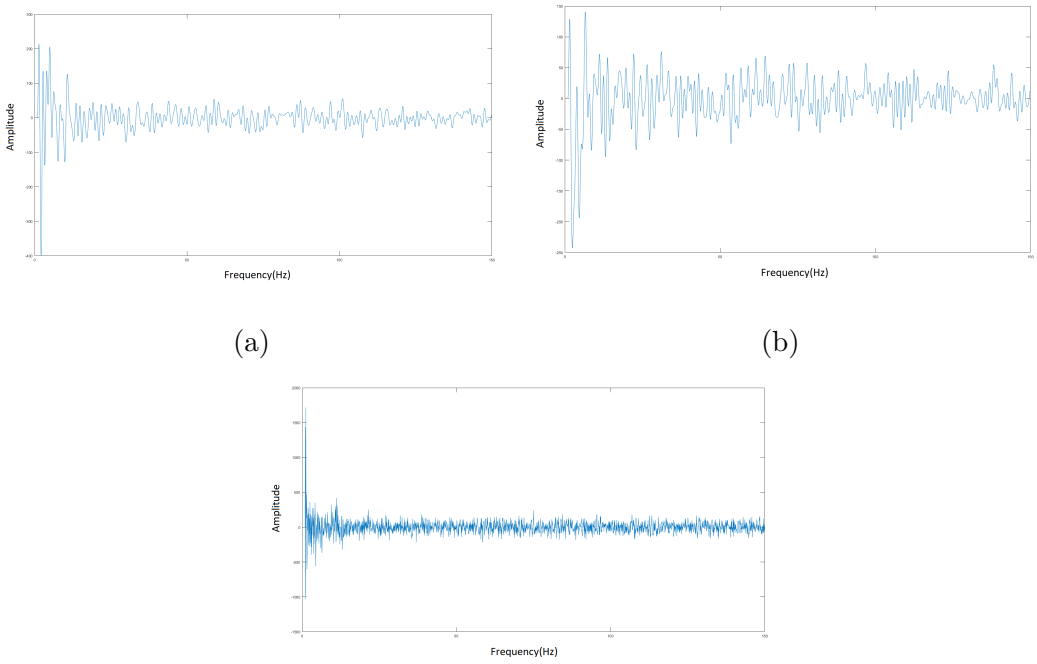


Figure 4.2: Frequency Domain Representation of EEG Signal for (a) Subject-1, Channel-1, Trial-1, (b) Subject-28, Channel-6, Trial-5, (c) Subject-45, Channel-22, Trial-10.

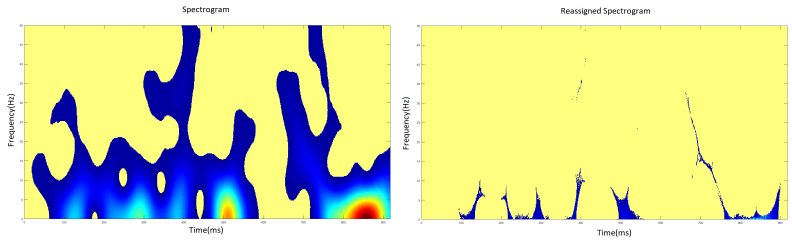
4.3 Time-frequency Domain Visualization

Time-frequency representation gives an overview of how a signal's frequencies are changing over time. There are lots of popular time-frequency representations available. For this study, a reassigned version of the spectrogram is used. Time-frequency domain representation using spectrogram is calculated using equation (4.2) (Sanders *et al.*, 2014).

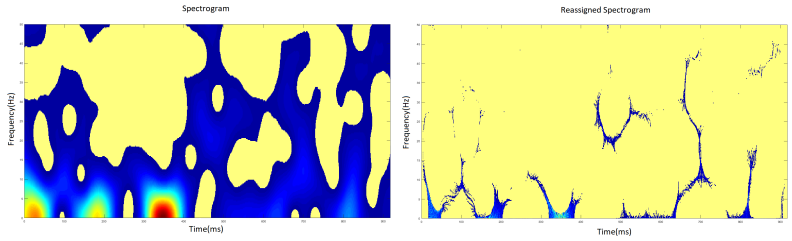
$$S_x(m,\omega) = \left| \sum_{n=-\infty}^{\infty} x[n]h[n-m]e^{-j\omega n} \right|^2 \tag{4.2}$$

Where, $x[n]$ is the signal and $h[n]$ is the window function. m here indicates the time and ω is the angular frequency. Time-frequency domain representation can provide

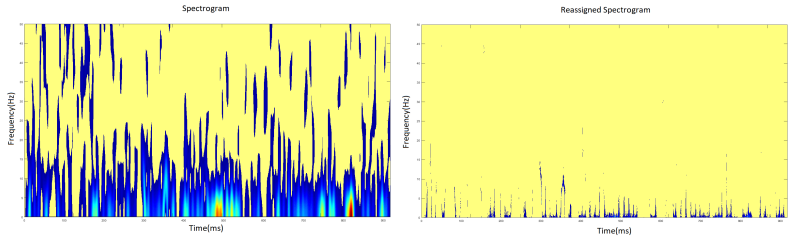
useful information about frequency variation over time. Figure-4.3 demonstrates sample time-frequency representations of the data.



(a)



(b)



(c)

Figure 4.3: Time-Frequency Domain Representation of EEG Signal for (a) Subject-1, Channel-1, Trial-1, (b) Subject-28, Channel-6, Trial-5, (c) Subject-45, Channel-22, Trial-10.

4.4 Individual Subject's Response Analysis

Looking into individual subject's reaction times over the period of 30 minutes, it is evident that the majority of the participants (75%) have shown a gradual increase of response time with successive trials. Performing a moving average operation on an individual's response times shows an upward trend. Sorting the response times of a subject from low to high, the overall variation of response times can be seen. The average value of skewness (defined as $(\mu - \nu)/\sigma$ where, μ is mean, ν is median and σ is standard deviation) for all the subjects' reaction time is 1.83. Sample plots of reaction time after moving average operation and sorting operation are shown in figure-4.4 and 4.5 respectively.

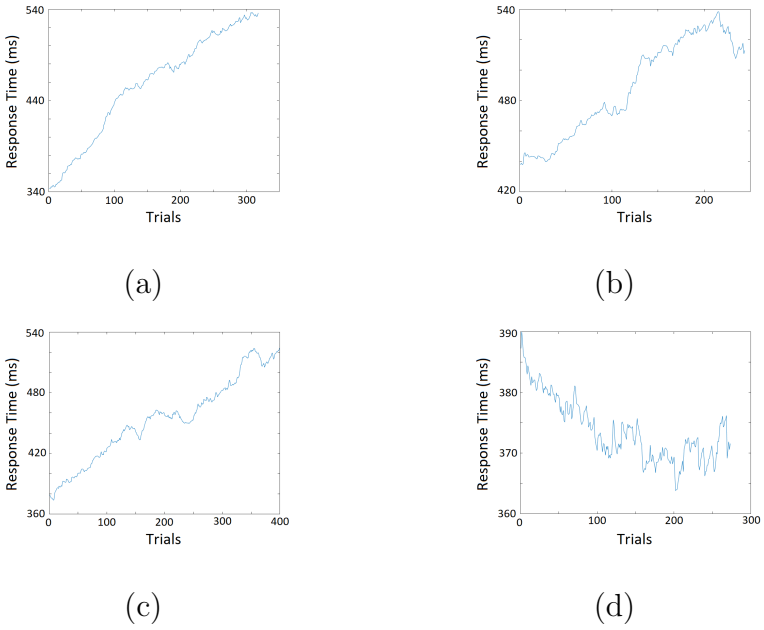


Figure 4.4: Sample Observation of Individual Response Times After Moving Average Operation Where, (a), (b) & (c) Shows Upward Trend, While (d) Shows Downward Trend.

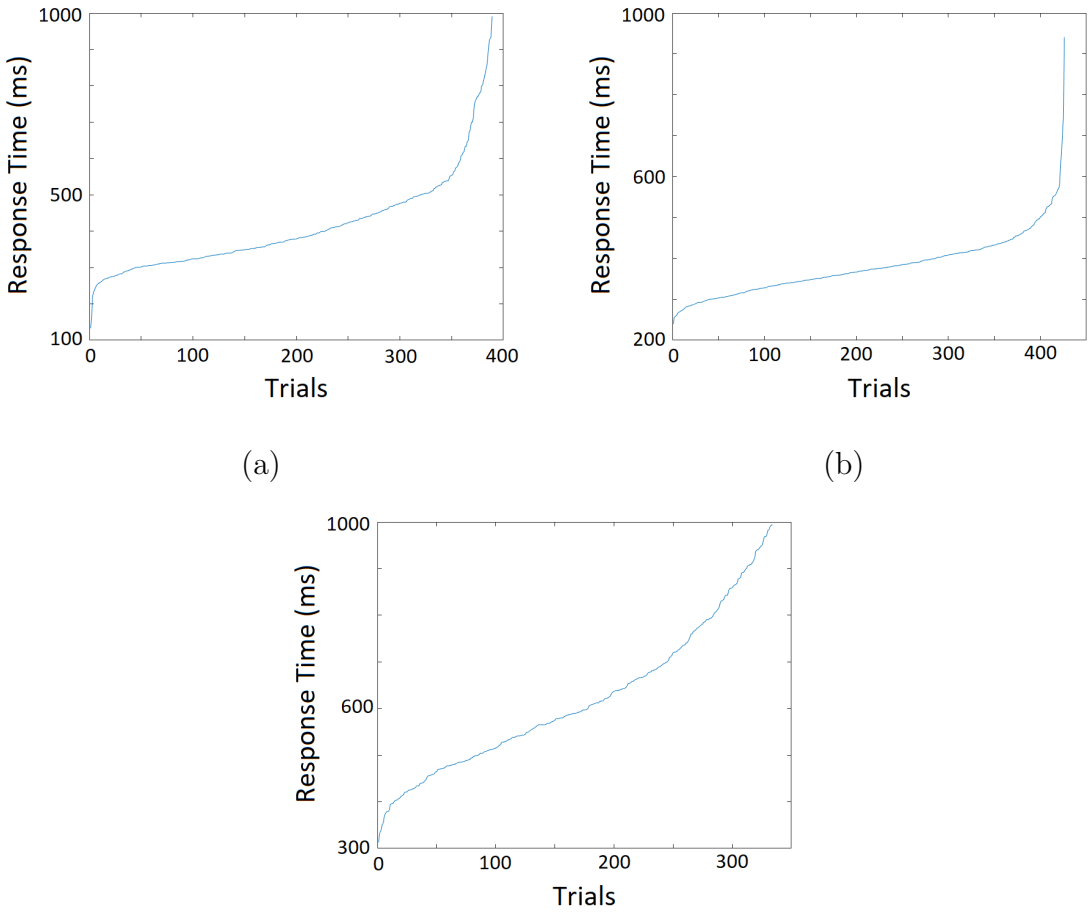


Figure 4.5: Sample Observation of Individual Response Times After Sorting Operation.

Chapter 5

FEATURE EXTRACTION

Features are extracted from the multi-channel EEG signal recorded during the observation period of each trial. Several features from time, frequency and time-frequency domain are extracted from all 30 channels (Aboalayon *et al.*, 2016).

5.1 Time Domain Features

From the time domain representation of the data, features like Hjorth parameters (Charbonnier *et al.*, 2011; Gudmundsson *et al.*, 2005) and average power can be of great use. Activity, mobility, and complexity are Hjorth parameters.

5.1.1 Activity

The activity parameter represents the signal power, the variance of a time function. This can indicate the surface of the power spectrum in the frequency domain. This is represented by equation (5.1):

$$Activity = var\big(y(t)\big) \tag{5.1}$$

Where y(t) represents the signal and var indicates variance.

5.1.2 Mobility

The mobility parameter represents the mean frequency or the proportion of the standard deviation of the power spectrum. This is defined as the square root of the

variance of the first derivative of the signal y(t) divided by variance of the signal y(t). Equation (5.2) shows it:

$$Mobility = \sqrt{\frac{var\left(\frac{dy(t)}{dt}\right)}{var\big(y(t)\big)}} \tag{5.2}$$

5.1.3 Complexity

The Complexity parameter represents the change in frequency. The parameter compares the signal's similarity to a pure sine wave, where the value converges to 1 if the signal is more similar. Equation (5.3) indicates the calculation:

$$Complexity = \frac{Mobility\left(\frac{dy(t)}{dt}\right)}{Mobility\big(y(t)\big)} \tag{5.3}$$

5.1.4 EEG Average Power

The average power of the sequence is calculated by taking the time sequence and computing the l-2 norm followed by an averaging operation.

5.2 Frequency Domain Features

5.2.1 Spectral Entropy

The spectral entropy is a measure of the regularity of the signal, a pure sine wave has entropy zero and uncorrelated white noise has entropy one. Spectral entropy (SEN) is the Shannon entropy properly normalized and applied to the power spectrum

density of the EEG signal (Radha *et al.*, 2014). SEN can be calculated using equation (5.4).

$$SEN = \frac{-\sum P_k \log P_k}{\log N} \tag{5.4}$$

where, P_k are the spectral powers of the normalized frequencies, such that $\sum P_k = 1$ and N is the number of frequencies (bin).

5.2.2 Percentage of Energy in Each Frequency Band

The observation sequence data was separated into delta(1-4Hz), theta(4-8Hz), alpha(8-12Hz), beta(12-35Hz) and gamma(35-150Hz) frequency bands (Aboalayon and Faezipour, 2014). Sum of power for each band divided by the total energy gives us the following percentages of energy in each frequency band:

1. Delta Energy/Total Energy
2. Theta Energy/Total Energy
3. Alpha Energy/Total Energy
4. Beta Energy/Total Energy
5. Gamma Energy/Total Energy

5.2.3 Power Ratio of the Frequency Bands

The ratio of the average power of one frequency band with respect to the consecutive higher frequency band is calculated, i.e., the power ratios of delta/theta, theta/alpha, alpha/beta, and beta/gamma are evaluated. They are as following:

1. Average Delta Power/Average Theta Power
2. Average Theta Power/Average Alpha Power

3. Average Alpha Power/Average Beta Power

4. Average Beta Power/Average Gamma Power

5.2.4 Power Ratio and Percentage Energy of the Frequency Bands for Multiple Smaller Windows

The ratio of the average power of one frequency band with respect to the consecutive higher frequency band as well as percentage energy is calculated for multiple 0.1-seconds smaller windows and the maximum, minimum and median values among all the 0.1-seconds windows are taken into account.

5.3 Time-frequency Domain Features

Renyi information and ridges can be useful in our study (Sanders *et al.*, 2014).

5.3.1 Renyi Information

Renyi information is a measurement of the complexity of a signal. It indicates the number of basic sinusoids involved with the particular signal in hand (Fraiwan *et al.*, 2012). Renyi information can be calculated using equation (5.5).

$$R_x^\alpha = \frac{1}{1-\alpha} \log_2 \left[\sum_{m=-\infty}^{\infty} \sum_{\omega=-\infty}^{\infty} S_x^\alpha(m,\omega) \right] \tag{5.5}$$

Where, R_x^α is the Renyi information of order α. $\alpha = 3$ was used for our calculation.

We used a moving non-overlapping window of 0.4 seconds duration to see the change of Renyi information with time. So, we have the following features:

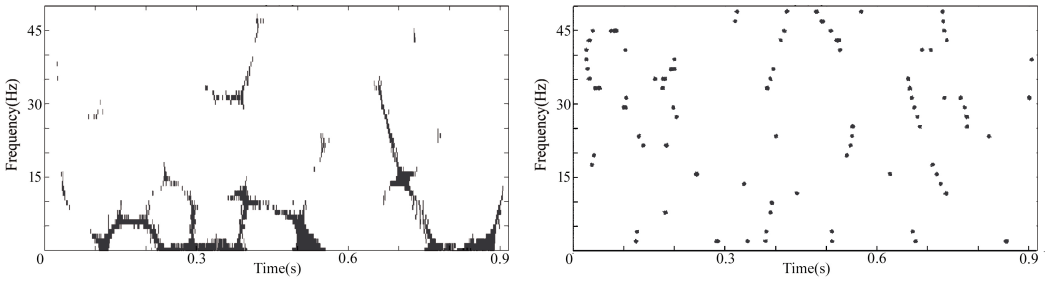
1. Maximum Renyi Information Among Windows

2. Minimum Renyi Information Among Windows

3. Median Renyi Information Among Windows

5.3.2 Ridges in Frequency Bands

Ridges give an idea of the time-frequency structure of the data by providing significant points of the representation. Figure-5.1 shows a sample time-frequency representation and the corresponding ridge plot.



(a) (b)

Figure 5.1: (a) Sample Reassigned Version of Spectrogram & (b) Corresponding Ridge Plot.

Using the same 0.4-seconds window, the number of ridges for different bands are calculated. The following features are taken from the calculations:

1. Average Total Number of Ridges Over All Windows
2. Maximum Number of Ridges for Delta and Theta Bands Among All Windows
3. Minimum Number of Ridges for Delta and Theta Bands Among All Windows
4. Median Number of Ridges for Delta and Theta Bands Among All Windows
5. Maximum Number of Ridges for Alpha and Beta Bands Among All Windows
6. Minimum Number of Ridges for Alpha and Beta Bands Among All Windows

7. Median Number of Ridges for Alpha and Beta Bands Among All Windows
8. Maximum Number of Ridges for Gamma Band Among All Windows
9. Minimum Number of Ridges for Gamma Band Among All Windows
10. Median Number of Ridges for Gamma Band Among All Windows

5.4 Last Moment Observation Effect

To see if the last moment observation has any impact on the outcome, all the above features were extracted again from the last 0.8 seconds of the observation period instead of the whole observation.

5.5 Non-EEG Features

All the features stated in the previous sections are extracted from a single trial EEG signal. Other than these, 3 additional features are added just to monitor the performance of the regression-based analysis. These features are discussed in this section.

5.5.1 Mean Waiting Time

The experiment is designed in such a way that the mean waiting time for the subjects can be classified into 3 categories. These categories consist of mean waiting times around 3500ms, 5000ms, and 6000ms respectively. Figure-5.2 shows the mean waiting time for all 48 subjects. Every single category has 16 subjects. These 3 categories are labeled as 1,2 and 3. Afterward, these labels are normalized and used as a subject-specific feature.

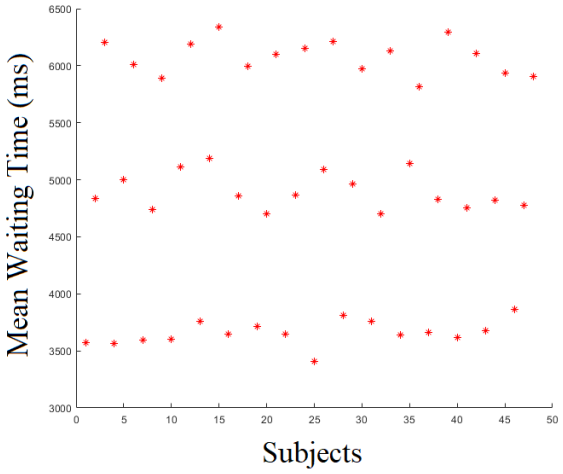


Figure 5.2: Mean Waiting Times for each Subject.

5.5.2 Subject Number

EEG signal varies from subject to subject. So, it is important to see if the subject number can contribute to our analysis by providing information about the subject identity.

5.5.3 Trial Number

Trial numbers can be a useful feature to see if the performance of a subject has any direct relation with the passage of time.

Chapter 6

REGRESSION APPROACH

Regression is done using extracted features to estimate the response time. Several algorithms are used to get results and compare them. The algorithms and results are discussed in this section.

6.1 Regression Algorithms

6.1.1 Linear Regression

Linear Regression is a statistical tool used to model the relationship between a dependent variable and some "explanatory" variables. Let the domain set $\mathcal{X}$ be a subset of $\mathbb{R}^d$ for some d, and let the label set $\mathcal{Y}$ belong to $\mathbb{R}$. Then the linear function h: $\mathbb{R}^d \longrightarrow \mathbb{R}$ best approximates the relationship between the variables. The hypothesis class of linear regression predictors can be denoted as

$$\mathcal{H}_{reg} = L_d = \{x \mapsto \langle \mathbf{w}, \mathbf{x} \rangle + b : \mathbf{w} \in \mathbb{R}^d, b \in \mathbb{R}\} \tag{6.1}$$

The definition of loss in classification is given by $\ell(h, (\mathbf{x}, y))$. This indicates whether h($\mathbf{x}$) correctly predicts y or not. For regression, the squared loss function is defined, which is given by:

$$\ell(h, (\mathbf{x}, y)) = (h(\mathbf{x}) - y)^2 \tag{6.2}$$

6.1.2 Ridge Regression

Ridge Regression is a technique for analyzing multiple regression data that suffer from multicollinearity. When multicollinearity occurs, least squares estimates are

unbiased, but their variances are large so they may be far from the true value. By adding a degree of bias to the regression estimates, ridge regression reduces the standard errors. It is hoped that the net effect will be to give estimates that are more reliable.

6.1.3 Support Vector Regression

SVMs are binary classifiers and their primary goal is to locate a separating hyperplane in the space between the two classes by mapping the data into a higher-dimensional space.

Let there be a dataset constituting l patterns where each l happens to be a pair of the type $(x_i, y_i) \forall i \in [1, ..., l], x_i \in \mathbb{R}^m$, and $y_i = \pm 1$. A standard binary SVM is formulated by minimizing a Convex Constrained Quadratic Programming (CCQP) objective function as follows:

$$\min_{\alpha} \tfrac{1}{2} \alpha^{\mathbf{T}} \mathbf{Q} \alpha - \mathbf{r}^{\mathbf{T}} \alpha \tag{6.3}$$

$$0 \leq \alpha_i \leq C \forall i \in [1, \ldots, l], \tag{6.4}$$

$$\mathbf{y}^{\mathbf{T}} \alpha = \mathbf{0}, \tag{6.5}$$

where C is the regularization parameter or penalty term which controls how well the SVM would perform on unseen test data, $r_i = 1 \forall i$ and Q is the symmetric positive semidefinite $l \times l$ kernel matrix where $q_{i,j} = y_i y_j K(x_i, x_j)$.

The solution of the CCQP function affords us with the $\alpha_i \forall i \in [1, \ldots, l]$ values which are plugged into the Feed-Forward Phase (FFP) of the SVM as follows in order

to do class prediction:

$$f(x) = \sum_{i=1}^{l} y_i \alpha_i K(x_i, x) + b \tag{6.6}$$

where b is the bias term and is generally computed based on the support vectors that lie in the margins.

6.1.4 Extra Tree Regression

The Extra-Tree method (standing for extremely randomized trees) was proposed with the main objective of further randomizing tree building in the context of numerical input features, where the choice of the optimal cut-point is responsible for a large proportion of the variance of the induced tree.

With respect to random forests, the method drops the idea of using bootstrap copies of the learning sample, and instead of trying to find an optimal cut-point for each one of the K randomly chosen features at each node, it selects a cut-point at random.

This idea is rather productive in the context of many problems characterized by a large number of numerical features varying more or less continuously: it leads often to increased accuracy thanks to its smoothing and at the same time significantly reduces computational burdens linked to the determination of optimal cut-points in standard trees and random forests.

6.1.5 Random Forest Regression

Random Forest operates by adding multiple decision trees and using the Bootstrap Aggregation Method. It is a highly efficient ML algorithm for predictive analysis. The random forest model is very suitable at handling tabular data with numerical features. Unlike linear models, random forests are able to capture non-linear interaction between the features and the target. However, the tree-based models are not designed to work with very sparse features. When dealing with sparse input data, the sparse features need to either be pre-processed to generate numerical statistics or in such cases, a linear model should be incorporated.

6.2 Results

To measure regression-based performance, correlation coefficient and root mean square error (RMSE) are computed using different algorithms. 3 different regression-based approaches are taken here:

1. At first, regression-based analysis is performed using the complete data-set.
2. Then the effect of additional non-EEG features is also observed.
3. Finally, as most of the reaction times are lying in the same region which is from 300 ms to 500 ms, a custom data-set consists of equivalent number of sample data points in all the regions is used to perform regression-based analysis.

6.2.1 Regression Using Full Data

Table-6.1 shows the outcomes. Actual and predicted reaction times and their comparisons are also provided in this segment.

Table 6.1: Regression-Based Analysis Results Using Full Data

Algorithm	Correlation Coeff.	RMSE (ms)
Linear Regression	0.39	116.6
Ridge Regression	0.4	114.8
Support Vector Regression	0.40	112.5
Extra Tree Regression	0.56	99.6
Random Forest Regression	**0.57**	**99.5**

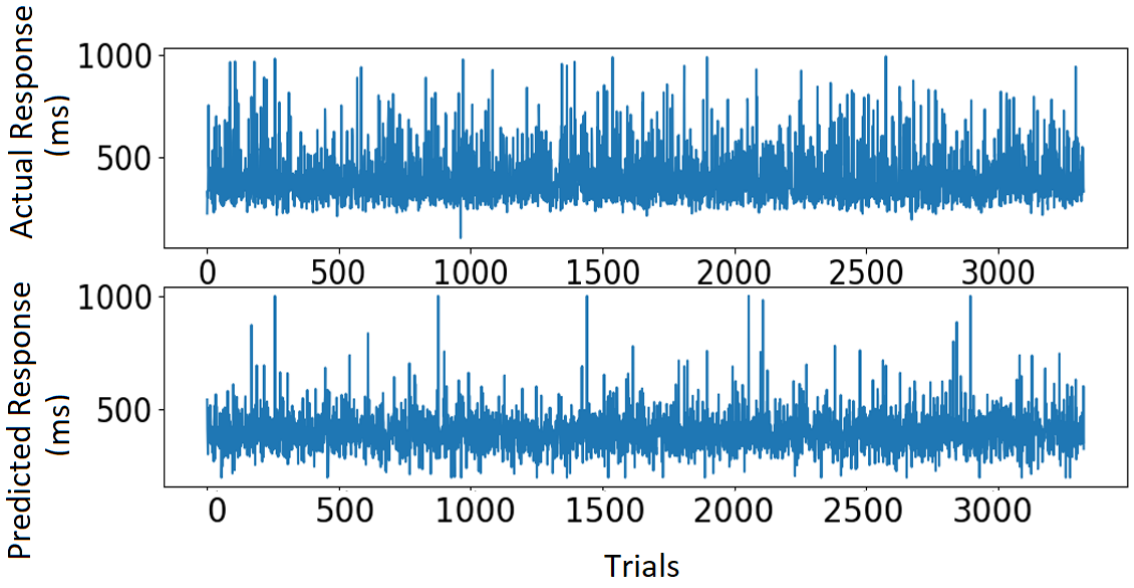


Figure 6.1: Actual and Predicted Reaction Time Using Linear Regression for Full Data.

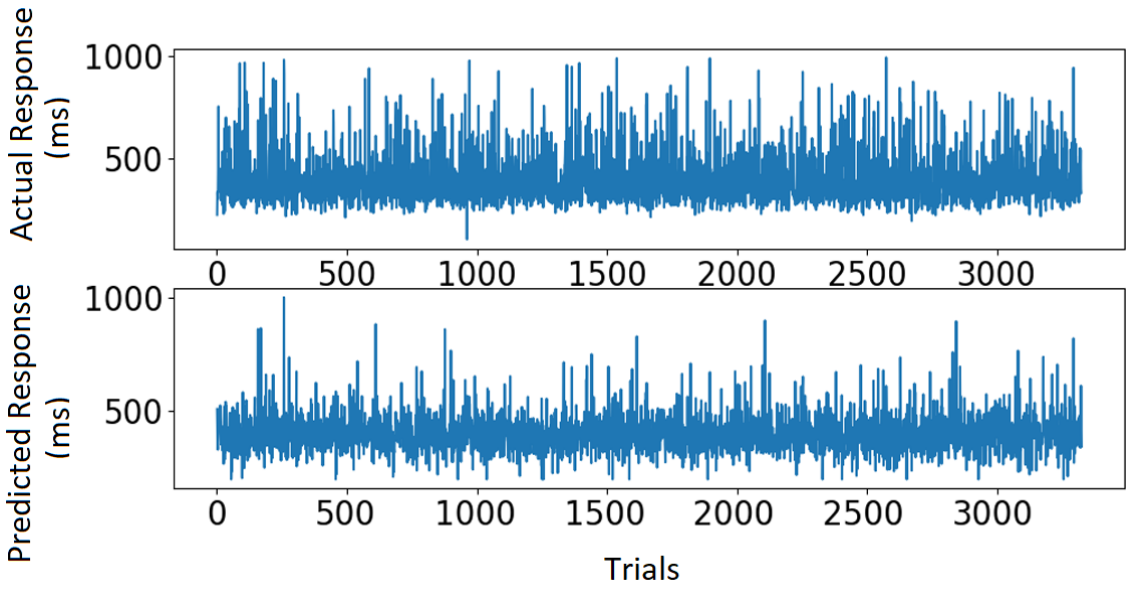


Figure 6.2: Actual and Predicted Reaction Time Using Ridge Regression for Full Data.

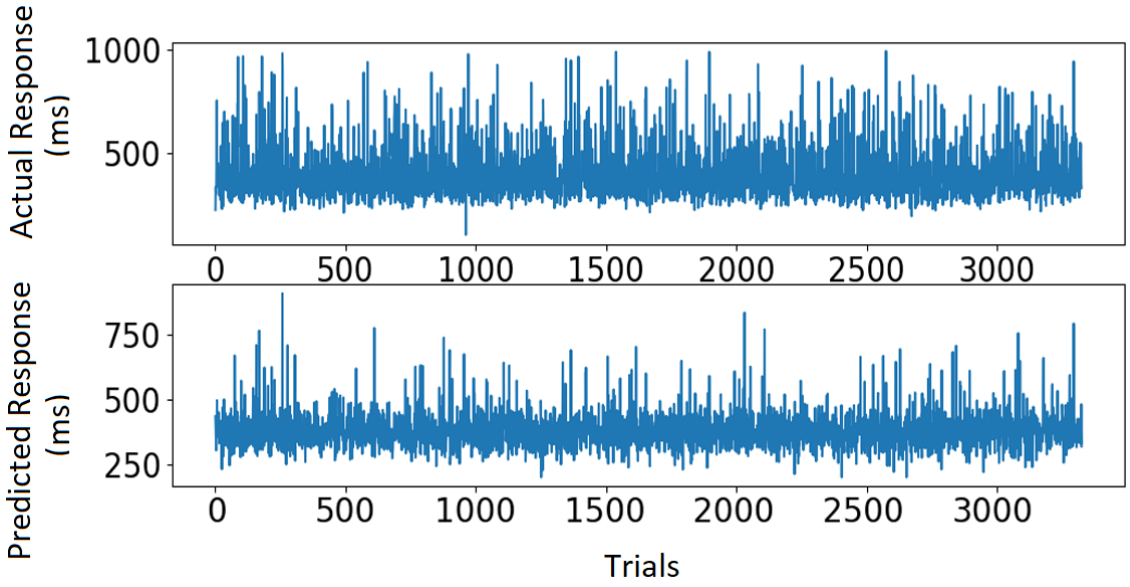


Figure 6.3: Actual and Predicted Reaction Time Using Support Vector Regression for Full Data.

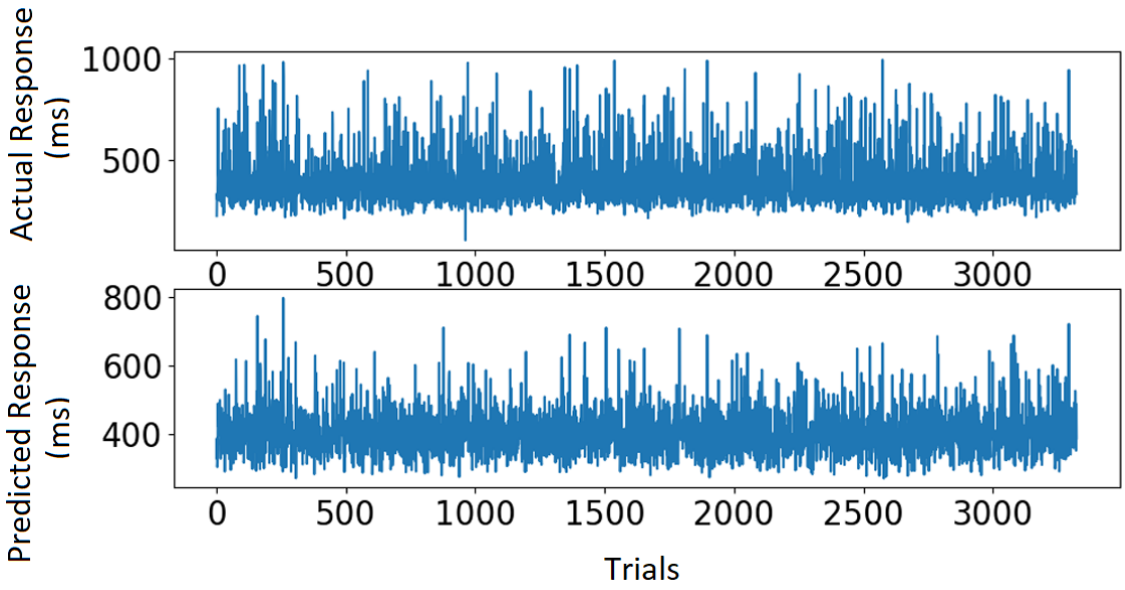


Figure 6.4: Actual and Predicted Reaction Time Using Extra Tree Regression for Full Data.

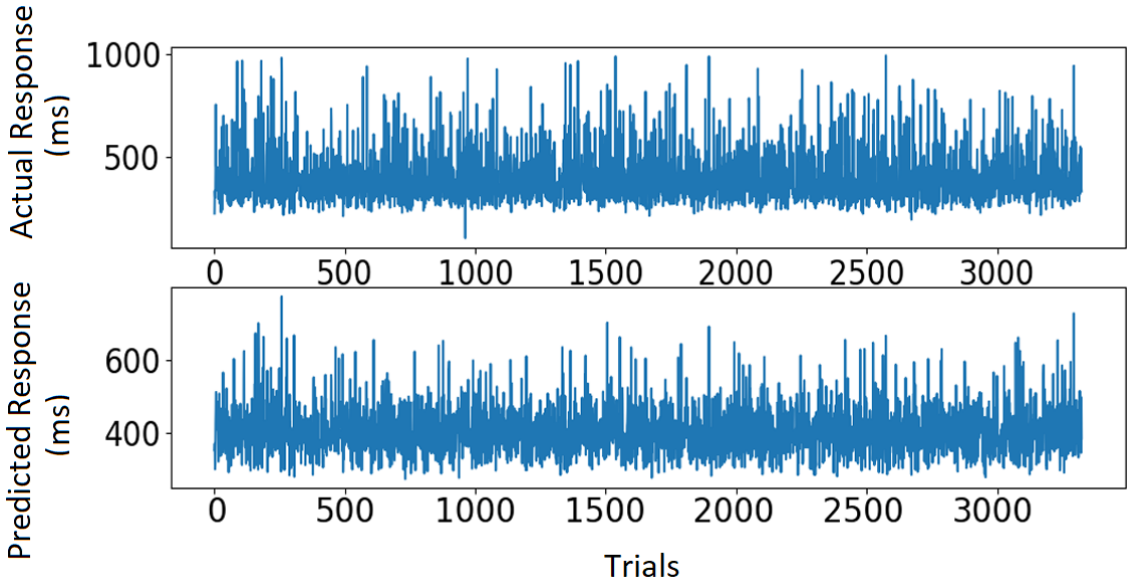
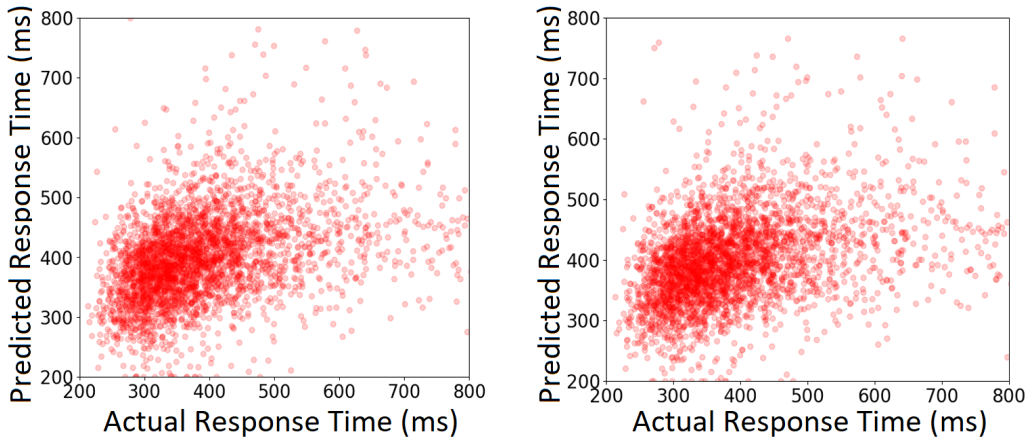
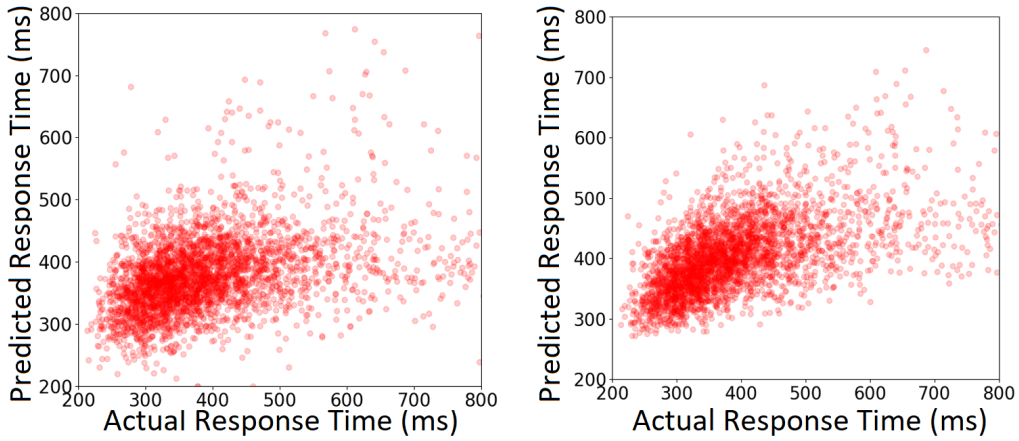


Figure 6.5: Actual and Predicted Reaction Time Using Random Forest Regression for Full Data.



(a) (b)

Figure 6.6: Actual vs Predicted Reaction Time for Full Data Using (a) Linear Regression & (b) Ridge Regression.



(a) (b)

Figure 6.7: Actual vs Predicted Reaction Time for Full Data Using (a) Support Vector Regression & (b) Extra Tree Regression.

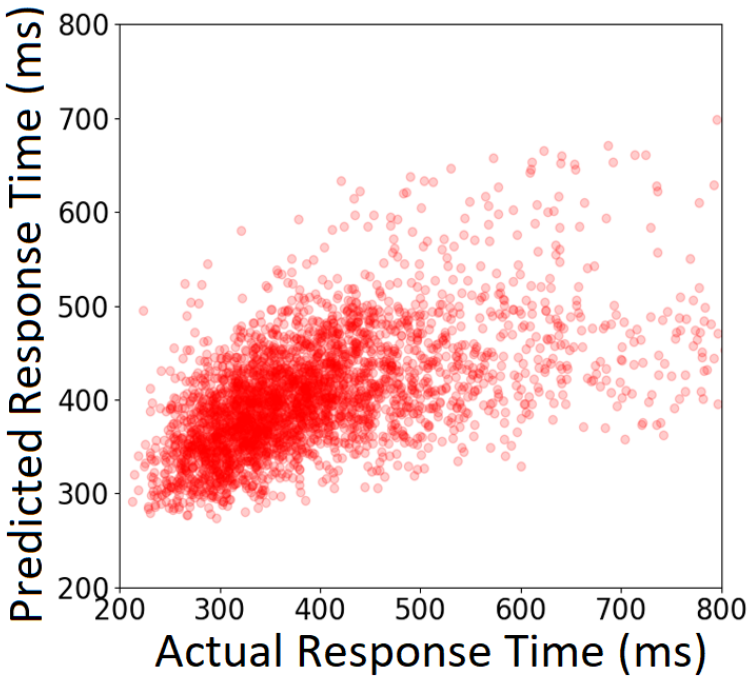


Figure 6.8: Actual vs Predicted Reaction Time for Full Data Using Random Forest Algorithm.

6.2.2 Regression Using Full Data and Additional Non-EEG Features

Table-6.2 shows the outcomes. Actual and predicted reaction times and their comparisons are also provided in this segment.

Table 6.2: Regression-Based Analysis Results Using Full Data and Additional Non-EEG Features

Algorithm	Correlation Coeff.	RMSE (ms)
Linear Regression	0.42	117.8
Ridge Regression	0.42	116.5
Support Vector Regression	0.43	114.9
Extra Tree Regression	0.56	102.8
Random Forest Regression	**0.58**	**101.7**

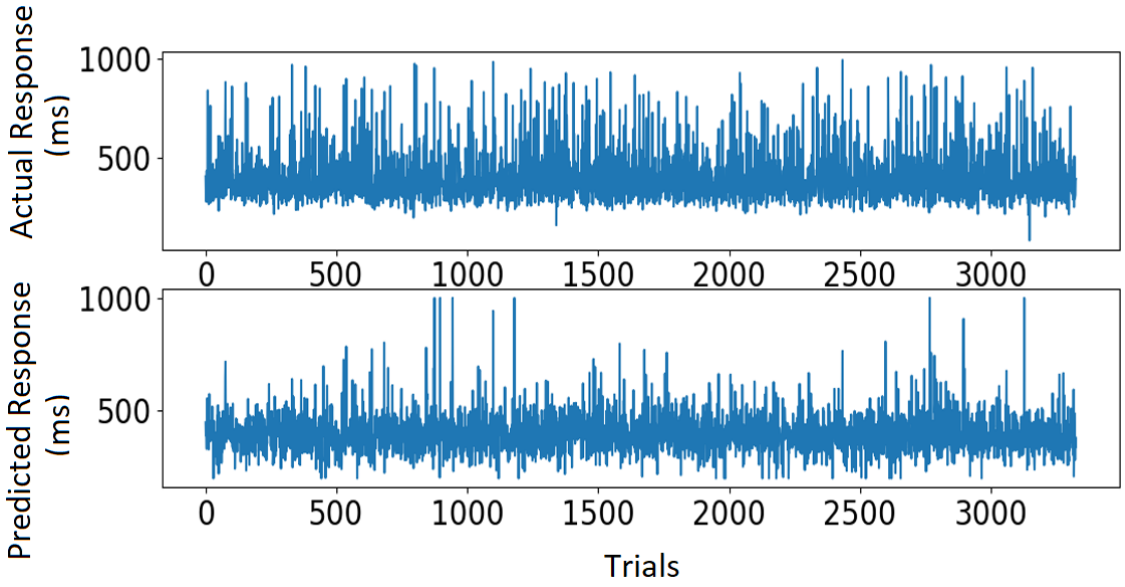


Figure 6.9: Actual and Predicted Reaction Time Using Linear Regression for Full Data and Additional Non-EEG Features.

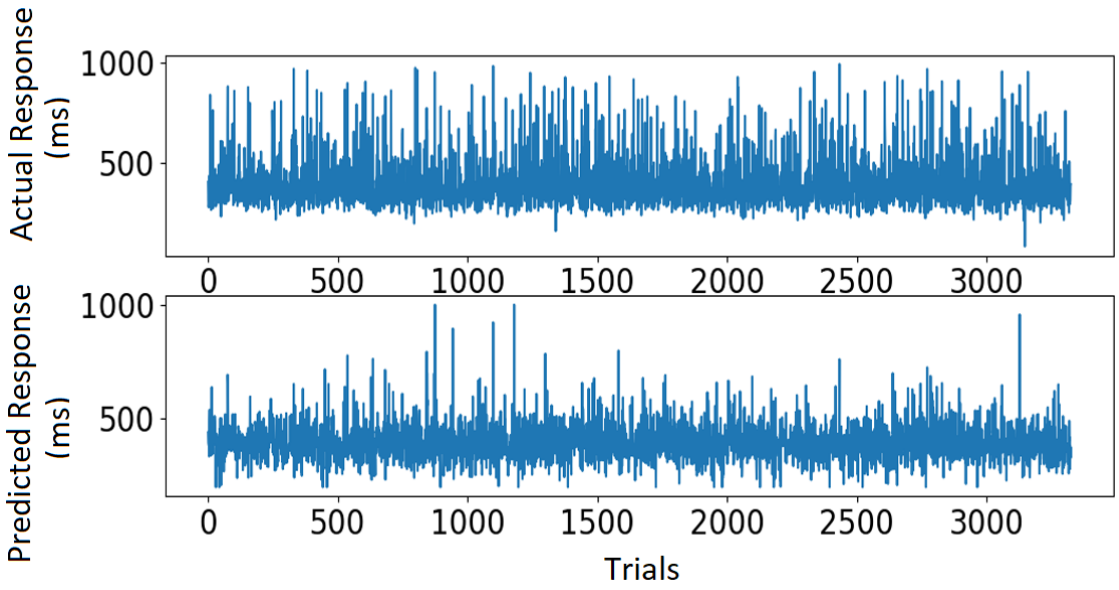


Figure 6.10: Actual and Predicted Reaction Time Using Ridge Regression for Full Data and Additional Non-EEG Features.

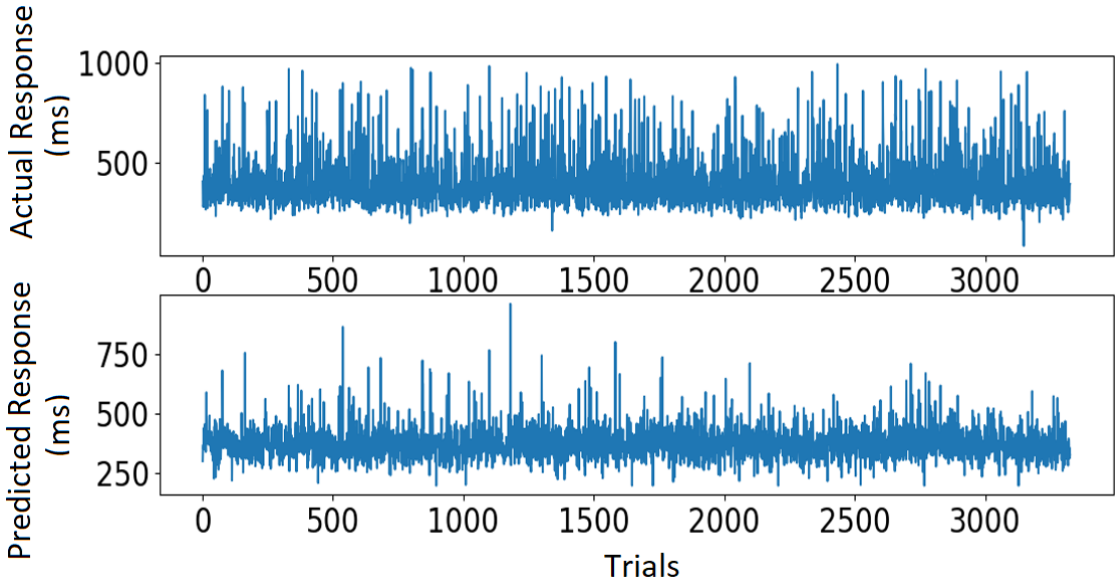


Figure 6.11: Actual and Predicted Reaction Time Using Support Vector Regression for Full Data and Additional Non-EEG Features.

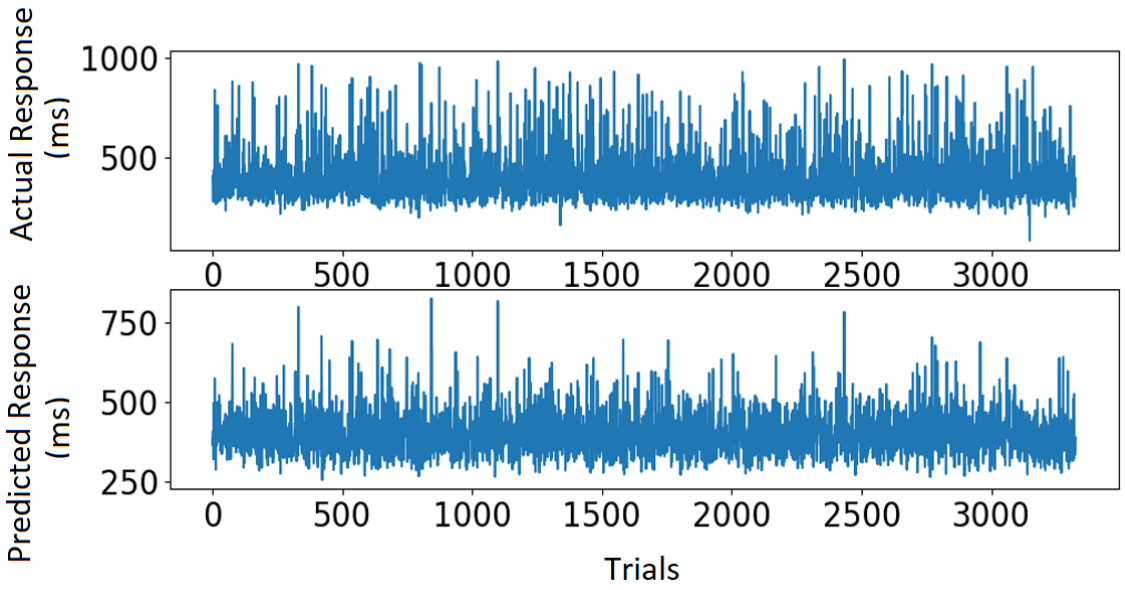


Figure 6.12: Actual and Predicted Reaction Time Using Extra Tree Regression for Full Data and Additional Non-EEG Features.

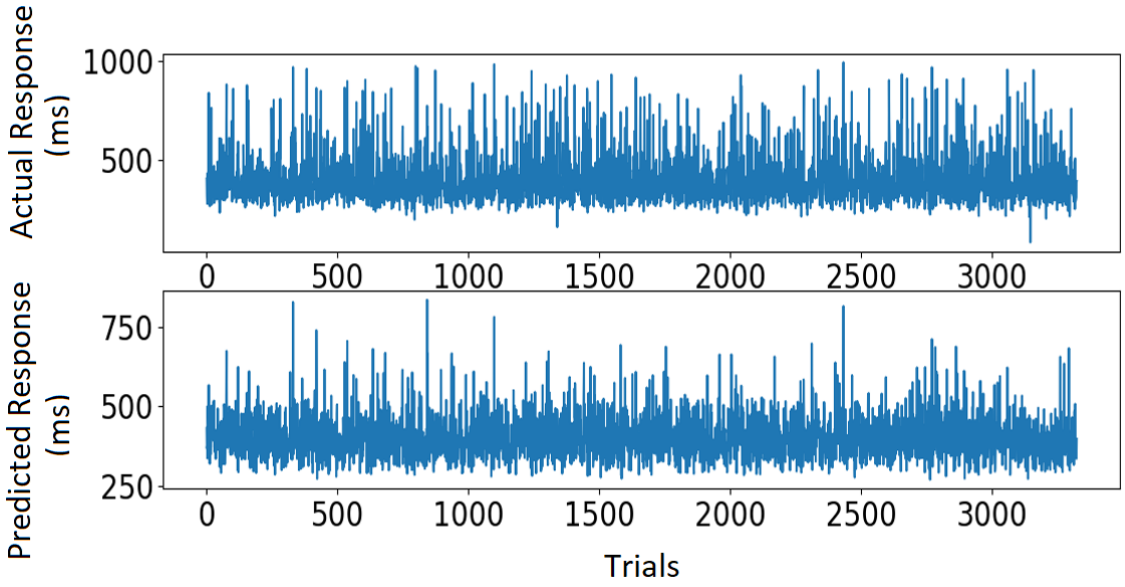
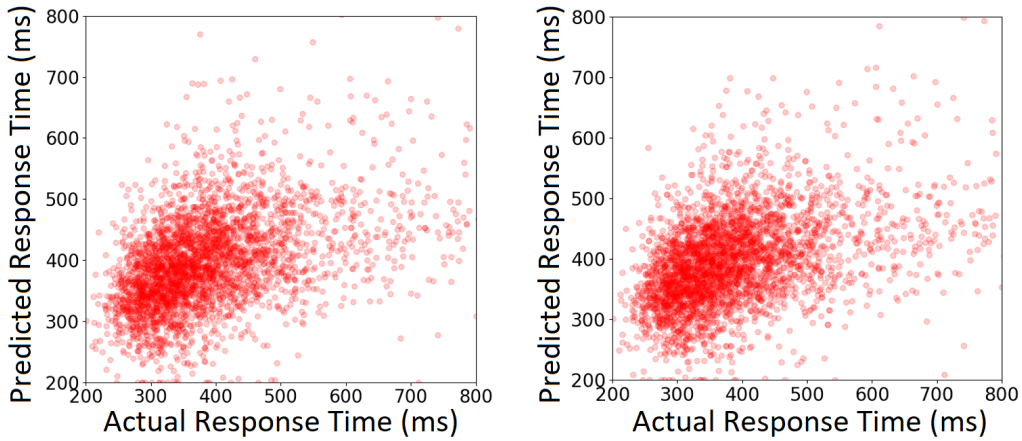
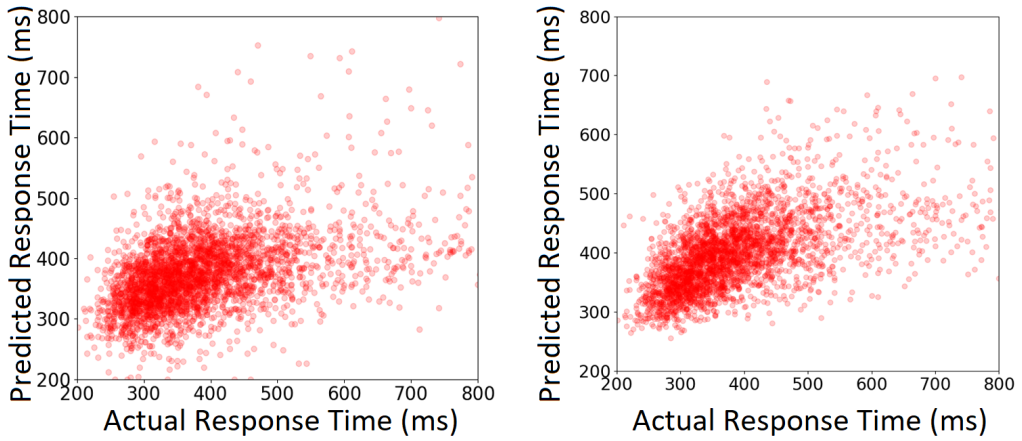


Figure 6.13: Actual and Predicted Reaction Time Using Random Forest Regression for Full Data and Additional Non-EEG Features.



(a) (b)

Figure 6.14: Actual vs Predicted Reaction Time for Full Data and Additional Non-EEG Features Using (a) Linear Regression & (b) Ridge Regression.



(a) (b)

Figure 6.15: Actual vs Predicted Reaction Time for Full Data and Additional Non-EEG Features Using (a) Support Vector Regression & (b) Extra Tree Regression.

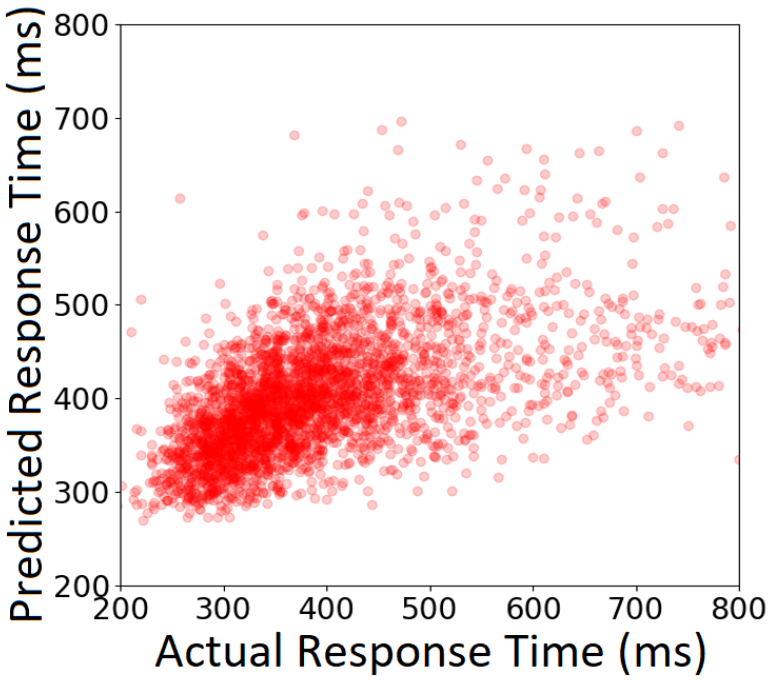


Figure 6.16: Actual vs Predicted Reaction Time for Full Data and Additional Non-EEG Features Using Random Forest Algorithm.

6.2.3 *Regression Using Custom Data*

As the full data-set has the majority of the data points with a reaction time between 300-500 ms, a custom data-set with a balanced number of data points in all the regions may show an improved set of results. Figure-6.17 shows a comparison between the full and the custom data-set.

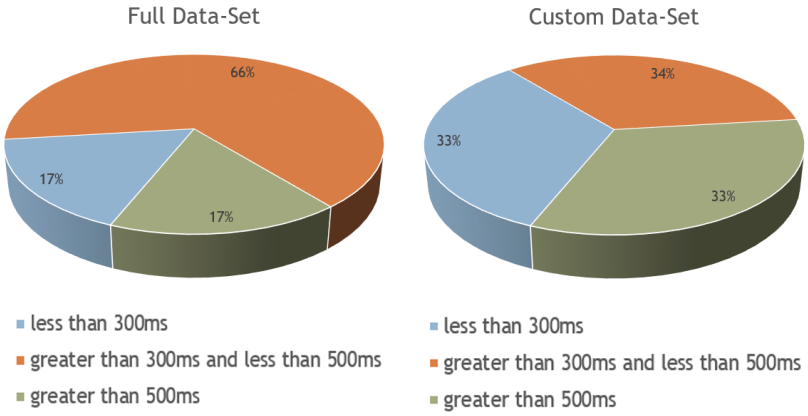


Figure 6.17: Full Data-Set and Custom Data-Set

Table-6.3 shows the outcomes. Actual and predicted reaction times and their comparisons are also provided in this segment.

Table 6.3: Regression-Based Analysis Results Using Custom Data

Algorithm	Correlation Coeff.	RMSE (ms)
Linear Regression	0.4	196.5
Ridge Regression	0.39	197.5
Support Vector Regression	0.46	175.7
Extra Tree Regression	0.68	136.1
Random Forest Regression	**0.69**	**135.7**

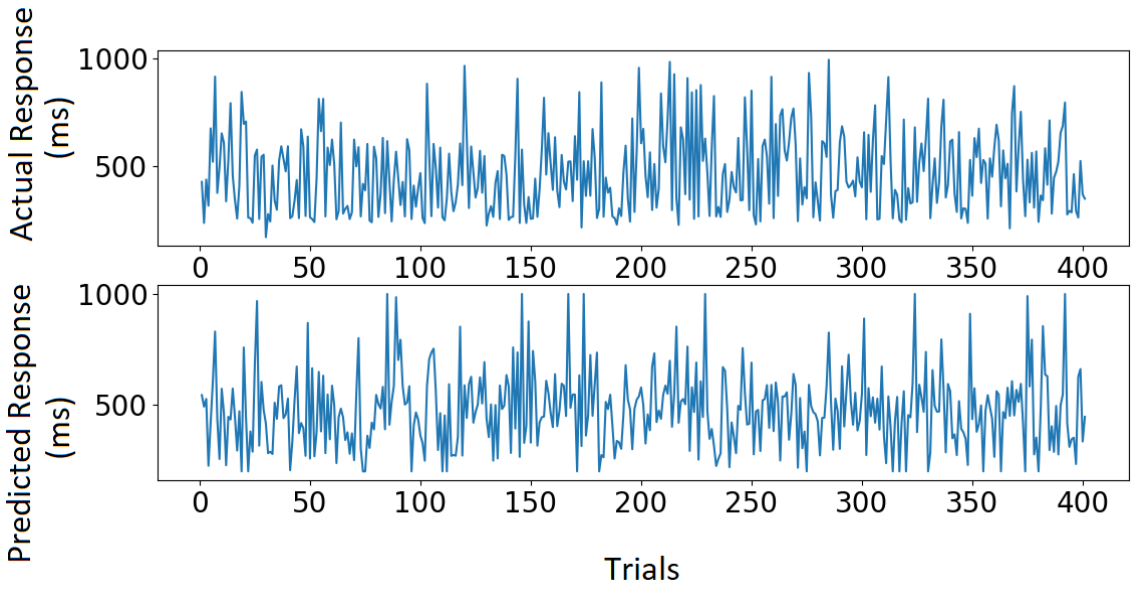


Figure 6.18: Actual and Predicted Reaction Time Using Linear Regression for Custom Data.

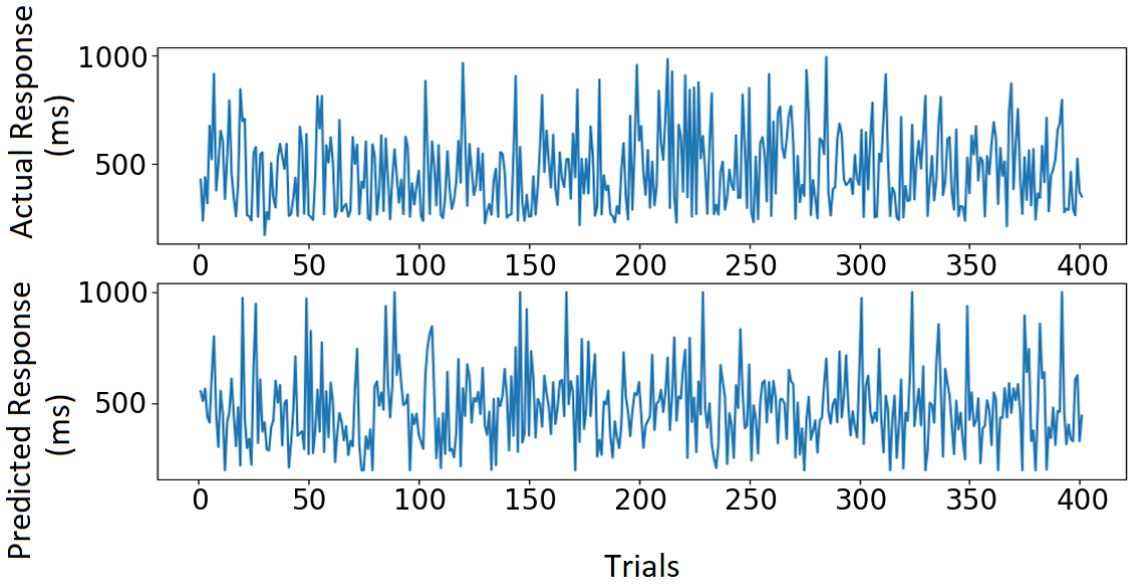


Figure 6.19: Actual and Predicted Reaction Time Using Ridge Regression for Custom Data.

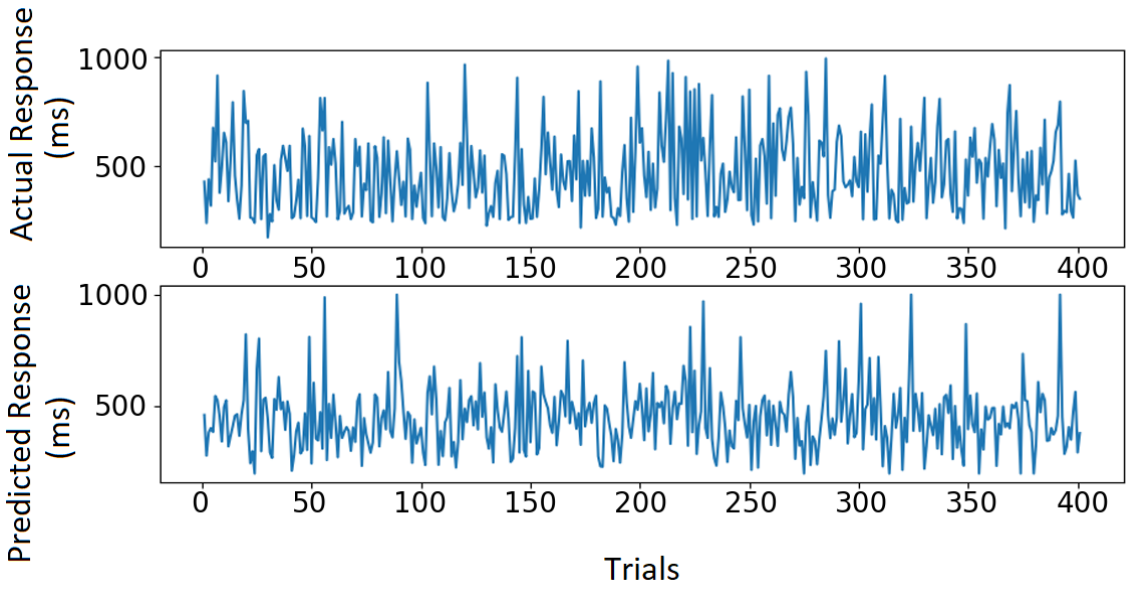


Figure 6.20: Actual and Predicted Reaction Time Using Support Vector Regression for Custom Data.

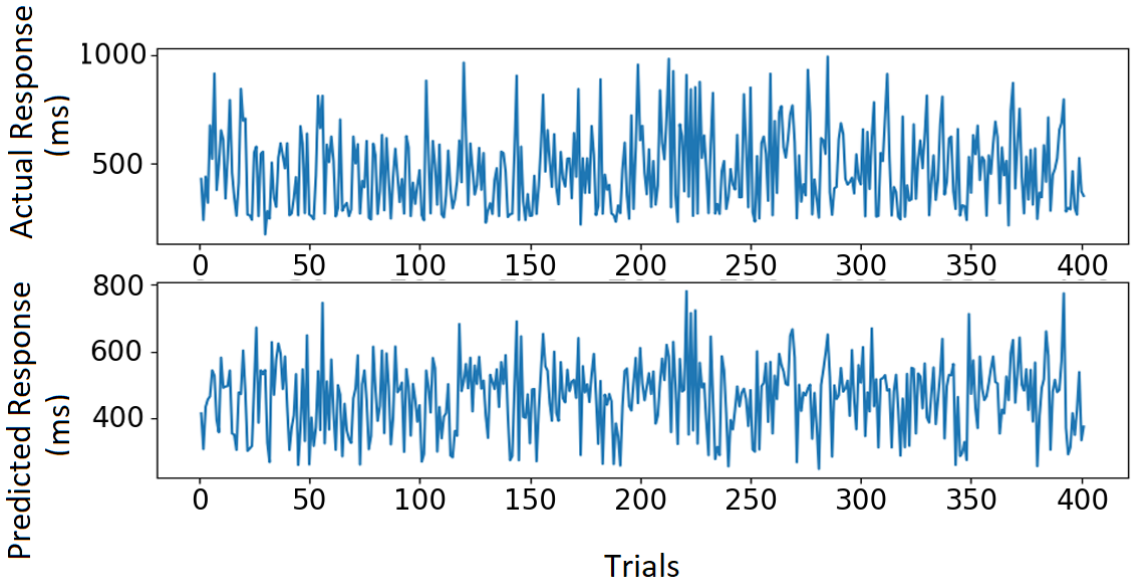


Figure 6.21: Actual and Predicted Reaction Time Using Extra Tree Regression for Custom Data.

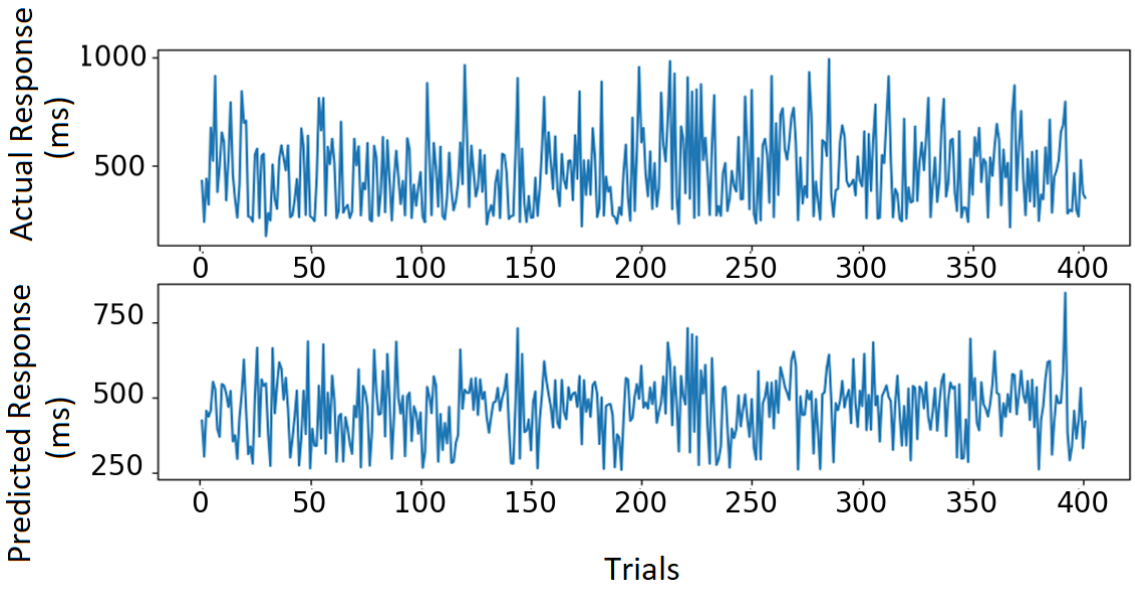
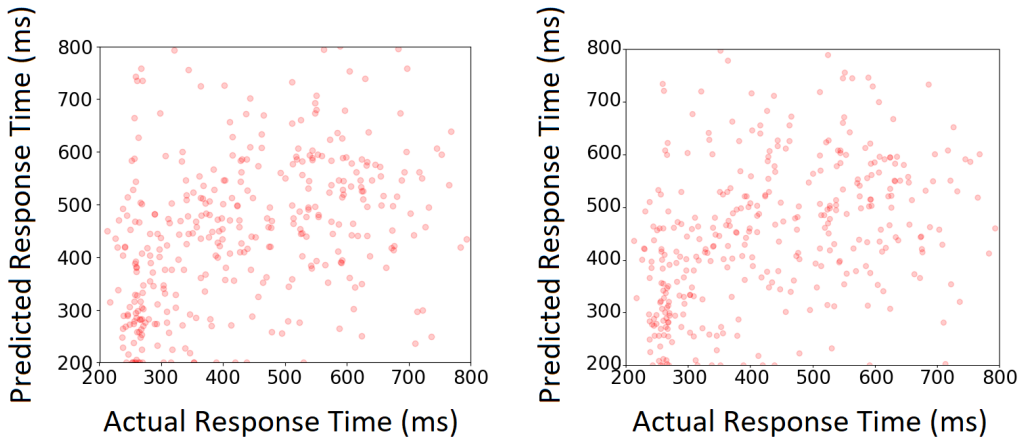
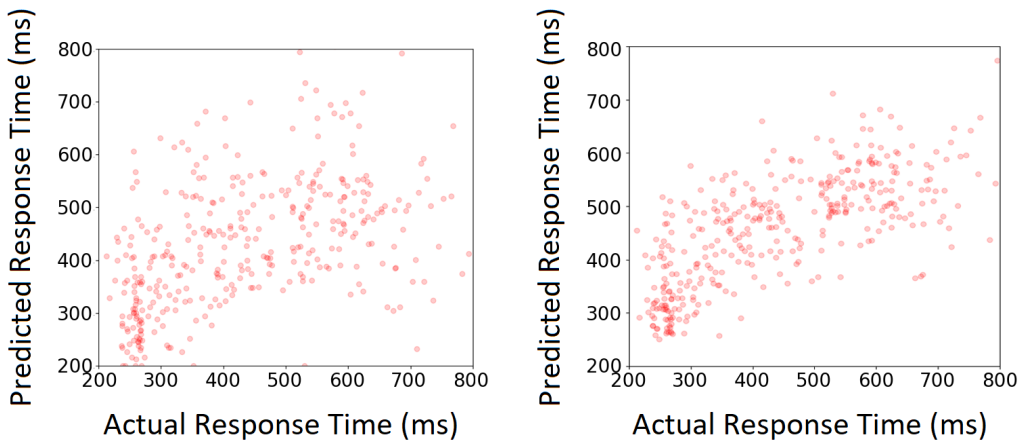


Figure 6.22: Actual and Predicted Reaction Time Using Random Forest Regression for Custom Data.



(a) (b)

Figure 6.23: Actual vs Predicted Reaction Time for Custom Data Using (a) Linear Regression & (b) Ridge Regression.



(a) (b)

Figure 6.24: Actual vs Predicted Reaction Time for Custom Data Using (a) Support Vector Regression & (b) Extra Tree Regression.

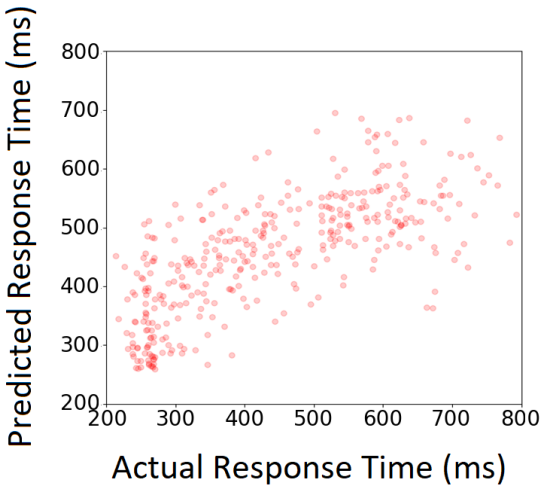


Figure 6.25: Actual vs Predicted Reaction Time for Custom Data Using Random Forest Algorithm.

Result Verification

Outcomes from 10 different randomizations of the data are averaged to make sure the results are consistent. The outcomes of this operation are:

1. For the full data-set, the average correlation coefficient value is 0.54.

2. For the custom data-set, the average correlation coefficient value is 0.65.

Chapter 7

CLASSIFICATION APPROACH

7.1 Classification Algorithms

Different classification algorithms ar taken into account for all types of analysis performed here (Sousa *et al.*, 2015; Liang *et al.*, 2011).

7.1.1 Linear Regression Classifier

Linear Regression is a statistical tool used to model the relationship between a dependent variable and some "explanatory" variables. Let the domain set $\mathcal{X}$ be a subset of $\mathbb{R}^d$ for some d, and let the label set $\mathcal{Y}$ belong to $\mathbb{R}$. Then the linear function h: $\mathbb{R}^d \longrightarrow \mathbb{R}$ best approximates the relationship between the variables. The hypothesis class of linear regression predictors can be denoted as

$$\mathcal{H}_{reg} = L_d = \{x \mapsto \langle \mathbf{w}, \mathbf{x} \rangle + b : \mathbf{w} \in \mathbb{R}^d, b \in \mathbb{R}\} \tag{7.1}$$

The definition of loss in classification is given by $\ell(h, (\mathbf{x}, y))$. This indicates whether h($\mathbf{x}$) correctly predicts y or not. For regression, the squared loss function is defined, which is given by,

$$\ell(h, (\mathbf{x}, y)) = (h(\mathbf{x}) - y)^2 \tag{7.2}$$

7.1.2 Decision Tree

DTs are designed with the following goals in mind: 1) accurately classify most of the training samples; 2) be adaptive enough to be equally or at least comparably good when working on the test dataset; 3) be easy to update for new incoming training

data; 4) and boast a pleasingly simple structure. While designing a DT the following are key points that are taken into account: the choice of the structure, the selection of feature subsets for each internal node and the choice of the strategy to be devised for each node. When the DT design is approached from a Bayesian perspective, the following optimization problem needs to be solved subject to limited training data:

$$\min_{T,F,d} p_e(T, F, d) \tag{7.3}$$

where p_e, is the total probability of error, T represents the selected tree structure and F and d are the feature subsets and strategies respectively which are to be used at the internal nodes. This optimization problem can be solved in two steps:

Step 1 For a given T and F search for d^* as follows:

$$d^* = d^*(T, F) \text{ such that } p_e(T, F, d^*(T, F)) = \min_d P_e(T, F, d) \tag{7.4}$$

Step 2 Find T^* and F^* such that:

$$P_e(T^*, F^*, d^*(T^*, F^*)) = \min_{T,F} p_e(T, F, d^*(T, F)) \tag{7.5}$$

7.1.3 Stochastic Gradient Descent

Gradient descent is an iterative optimization technique wherein the solution at each step is improved by taking a step along the negative of the gradient of the function to be minimized at the current point. SGD allows the optimization procedure to take a step along a random direction, as long as the expected value of the direction is the negative of the gradient. Here, the direction need not be updated to be based exactly on the gradient. Instead, the direction is set to be a random vector and its expected value at each iteration equals the gradient direction. In other words, the expected value of the random vector will be a subgradient of the function at the

current vector. SGD is a simple and an efficient approach to discriminative learning of linear classifiers under convex loss functions.

7.1.4 Support Vector Classifier

SVMs are binary classifiers and their primary goal is to locate a separating hyperplane in the space between the two classes by mapping the data into a higher dimensional space.

Let there be a dataset constituting l patterns where each l happens to be a pair of the type $(x_i, y_i) \forall i \in [1, ..., l], x_i \in \mathbb{R}^m$, and $y_i = \pm 1$. A standard binary SVM is formulated by minimizing a Convex Constrained Quadratic Programming (CCQP) objective function as follows:

$$\min_{\alpha} \tfrac{1}{2} \alpha^{\mathbf{T}} \mathbf{Q} \alpha - \mathbf{r}^{\mathbf{T}} \alpha \tag{7.6}$$

$$0 \leq \alpha_i \leq C \forall i \in [1, \ldots, l], \tag{7.7}$$

$$\mathbf{y}^{\mathbf{T}} \alpha = \mathbf{0}, \tag{7.8}$$

where C is the regularization parameter or penalty term which controls how well the SVM would perform on unseen test data, $r_i = 1 \forall i$ and Q is the symmetric positive semidefinite $l \times l$ kernel matrix where $q_{i,j} = y_i y_j K(x_i, x_j)$.

The solution of the CCQP function affords us with the $\alpha_i \forall i \in [1, \ldots, l]$ values which are plugged into the Feed-Forward Phase (FFP) of the SVM as follows in order

to do class prediction:

$$f(x) = \sum_{i=1}^{l} y_i \alpha_i K(x_i, x) + b \tag{7.9}$$

where b is the bias term and is generally computed based on the support vectors that lie in the margins.

7.1.5 Random Forest Classifier

Random Forest operates by adding multiple decision trees and using the Bootstrap Aggregation Method. It is a highly efficient ML algorithm for predictive analysis. The random forest model is very suitable at handling tabular data with numerical features. Unlike linear models, random forests are able to capture non-linear interaction between the features and the target. However, the tree-based models are not designed to work with very sparse features. When dealing with sparse input data, the sparse features need to either be pre-processed to generate numerical statistics or in such cases, a linear model should be incorporated.

7.2 Binary Classification

A binary classification case can be considered where the data separation point is 500ms (midpoint of response time range) and the classes can be labeled as:

1. above midpoint (response time>500 ms)
2. below midpoint (response time<=500ms)

7.3 3-Class Classification

As mean response time is around 400 ms, a 3-class classification case can also be formulated where the class boundaries are 400 ± 100 ms around. The classes can be labeled as:

1. low response time (response time<=315 ms)
2. normal response time (315ms<response time<=515ms)
3. high response time (response time>515ms)

As the number of data is way too much in the normal response time class, class weight balance or training data balance is required to perform 3-class classification.

7.4 High-Low Classification

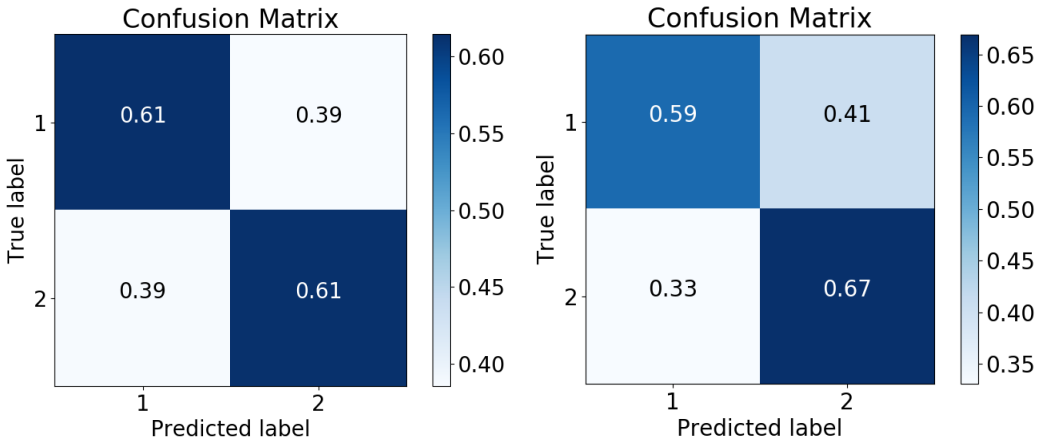
Lastly, it is also of interest to see how good the extremes can be seperated. So, taking the normal response class out of the calculation, a binary classification of high and low response time is taken into account.

7.5 Results

7.5.1 Binary Classification

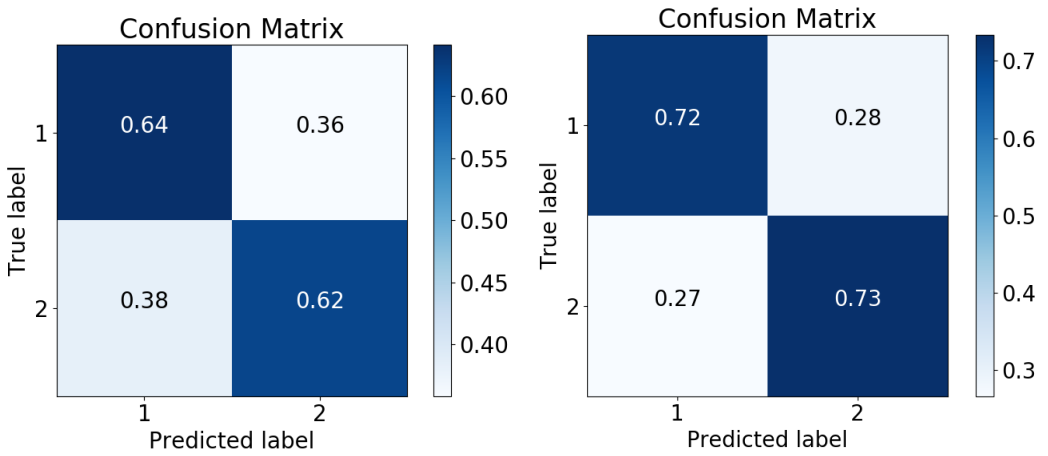
Table 7.1: Binary Classification Results for Below and Above Midpoint Categories

Algorithm	Accuracy(%)	Precision	Recall
Linear Regression	67	0.67	0.67
Decision Tree	63	0.63	0.63
SGD	63	0.63	0.63
SVM	73	0.72	0.73
Random Forest	**79**	**0.79**	**0.78**



(a) (b)

Figure 7.1: Confusion Matrix for Binary Classification Using (a) Linear Regression-Based Classifier & (b) Decision Tree Algorithm.



(a) (b)

Figure 7.2: Confusion Matrix for Binary Classification Using (a) Stochastic Gradient Descent & (b) Support Vector Machine Algorithm.

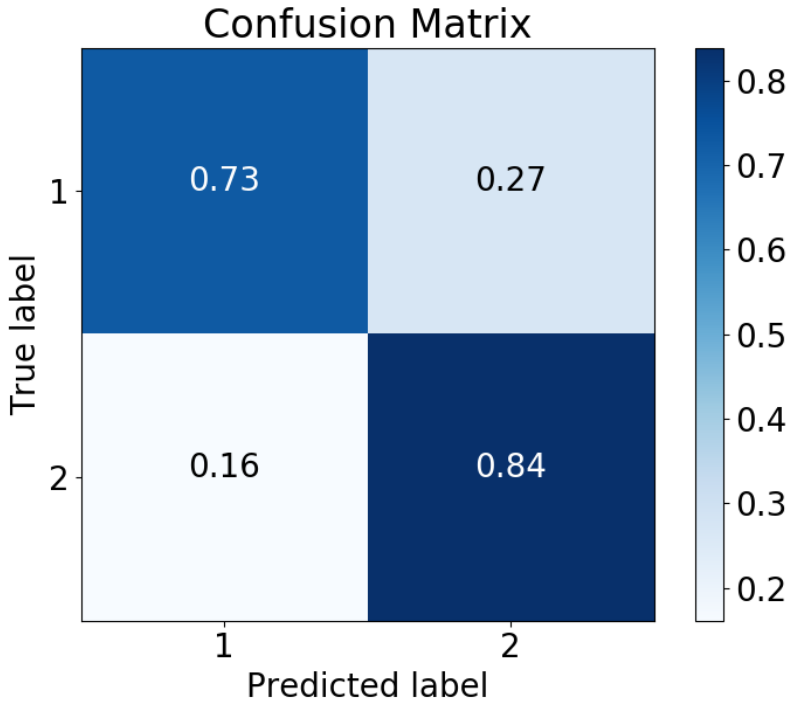
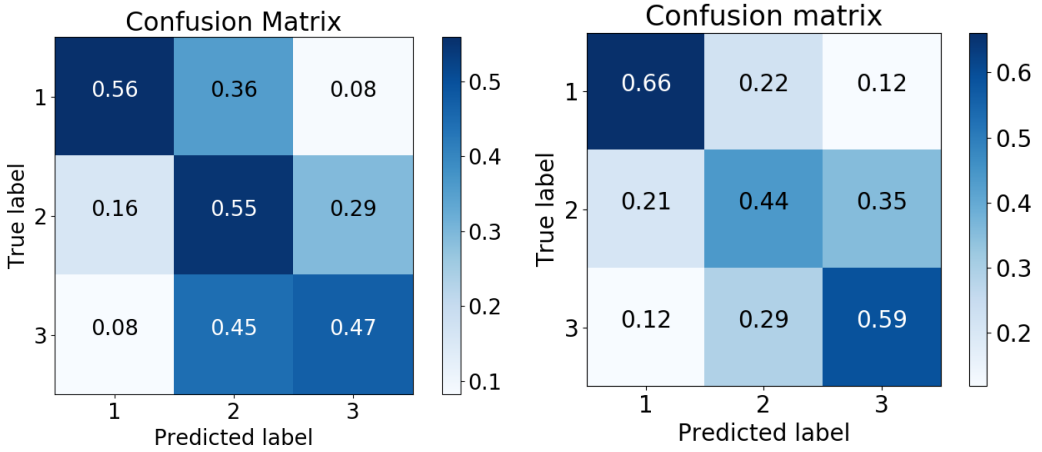


Figure 7.3: Confusion Matrix for Binary Classification Using Random Forest Algorithm.

7.5.2 3-Class Classification

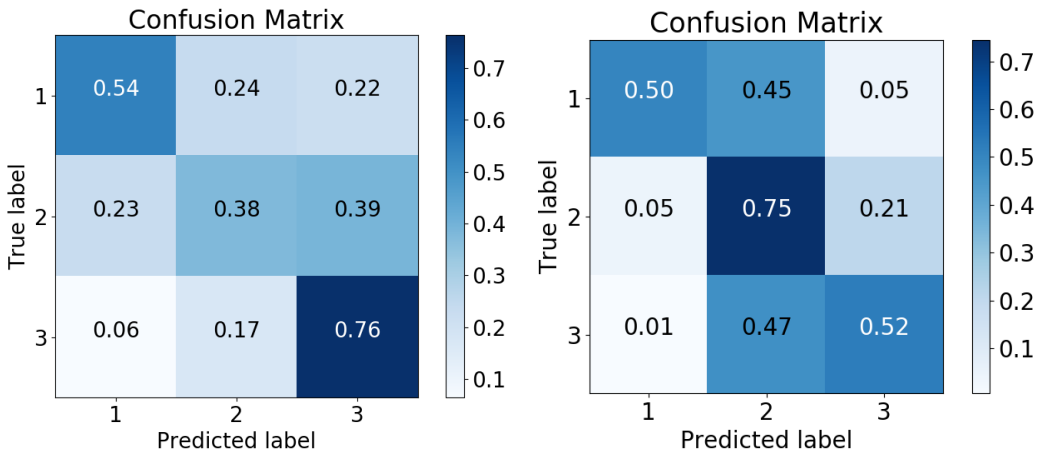
Table 7.2: 3-Class Classification Results for Low, Normal & High Response Time

Algorithm	Accuracy(%)	Precision	Recall
Linear Regression	52	0.54	0.53
Decision Tree	56	0.55	0.56
SGD	59	0.56	0.56
SVM	70	0.54	0.59
Random Forest	**72**	**0.71**	**0.70**



(a) (b)

Figure 7.4: Confusion Matrix for 3-Class Classification Using (a) Linear Regression-Based Classifier & (b) Decision Tree Algorithm.



(a) (b)

Figure 7.5: Confusion Matrix for 3-Class Classification Using (a) Stochastic Gradient Descent & (b) Support Vector Machine Algorithm.

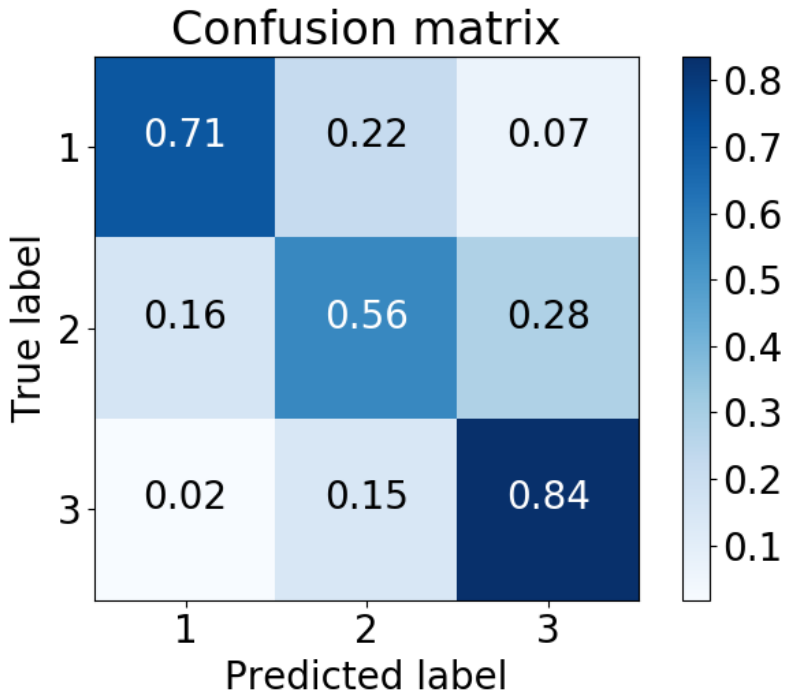
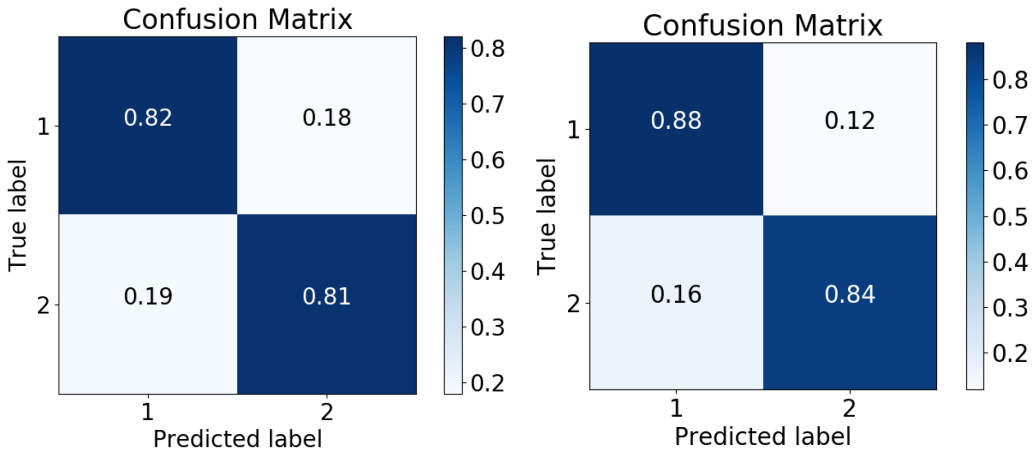


Figure 7.6: Confusion Matrix for 3-Class Classification Using Random Forest Algorithm.

7.5.3 High-Low Classification

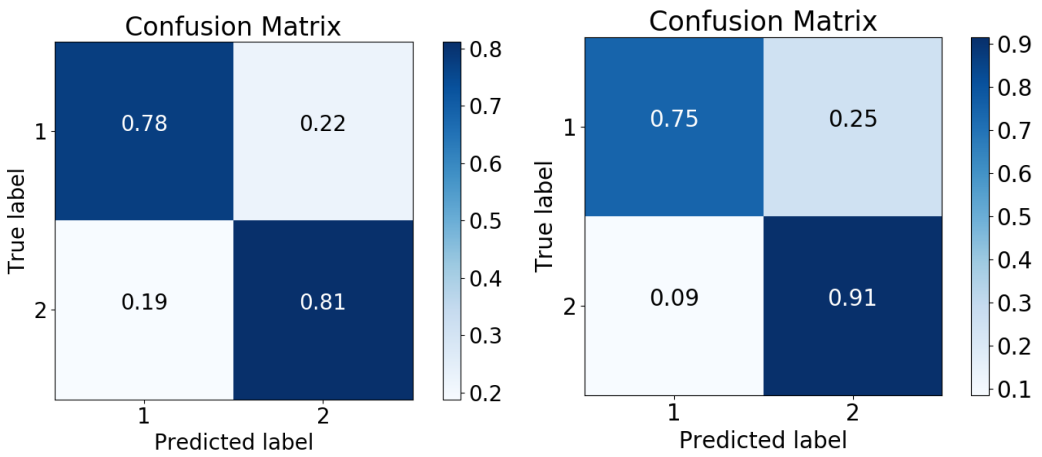
Table 7.3: Binary Classification Results for High and Low Categories

Algorithm	Accuracy(%)	Precision	Recall
Linear Regression	80	0.82	0.82
Decision Tree	85	0.84	0.86
SGD	80	0.78	0.79
SVM	85	0.85	0.83
Random Forest	**95**	**0.96**	**0.93**



(a) (b)

Figure 7.7: Confusion Matrix for High-Low Classification Using (a) Linear Regression-Based Classifier & (b) Decision Tree Algorithm.



(a) (b)

Figure 7.8: Confusion Matrix for High-Low Classification Using (a) Stochastic Gradient Descent & (b) Support Vector Machine Algorithm.

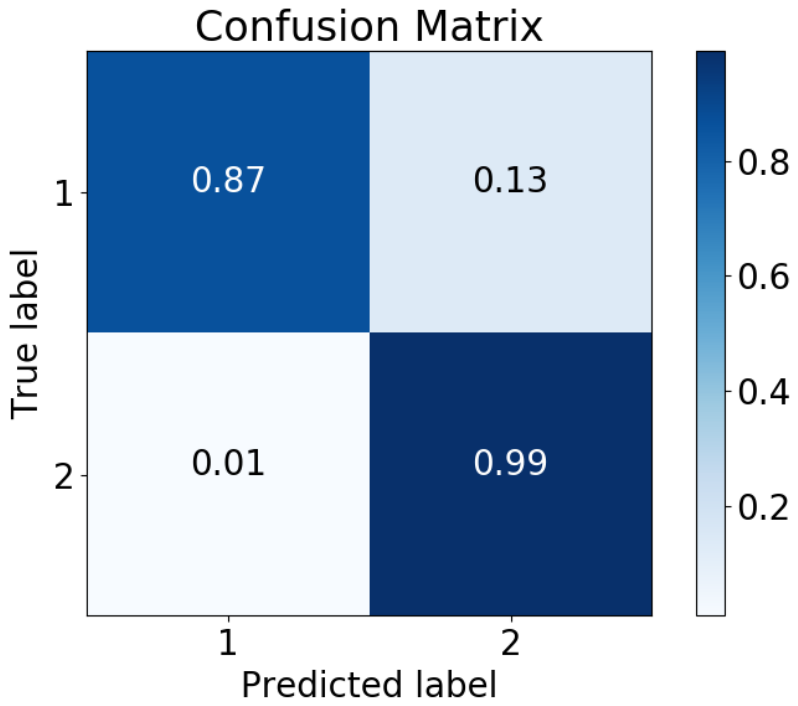


Figure 7.9: Confusion Matrix for High-Low Classification Using Random Forest Algorithm.

Chapter 8

EEG CHANNEL AND FEATURE ISOLATION

8.1 Channel Isolation

8.1.1 Method

Based on binary classification of high and low response using random forest classifier, important channels are isolated (Koley and Dey, 2012) combining the following methods:

1. Recursive elimination
2. Tree-based feature importance method
3. Mutual feature information criterion

Recursive elimination of unnecessary channels preserving or enhancing the overall accuracy is demonstrated in Algorithm-1.

8.1.2 Results

Using the stated set of methods, it's evident that the following channels are of the highest importance:

1. T7
2. C3
3. TP7
4. CP3
5. P7
6. P3

Figure-8.1 shows the important channels' locations.

Algorithm 1: Channel Isolation

Input: EEG features from all the channels

Output: useful channels

Initialization: $N \leftarrow 0$,

useful channels $\leftarrow$ {1,2,3,...,29,30}

```
while N ≠ size of useful channels do
    N ← size of useful channels
    X ← Random Forest accuracy for useful channels
    channels ← null
    for n ← 1 to N do
        modified features ← EEG features for (useful channels - n_th element of
         useful channels)
        accuracy ← Random Forest accuracy using modified features
        if accuracy >= X then
            X ← accuracy
        else
            channels.append ( n_th element of useful channels)
    useful channels ← channels
```

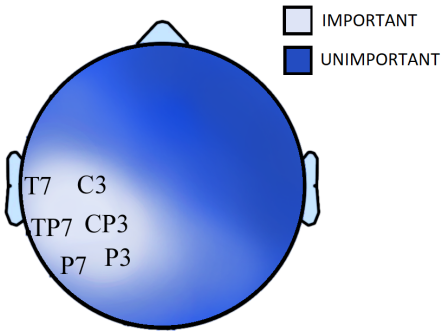
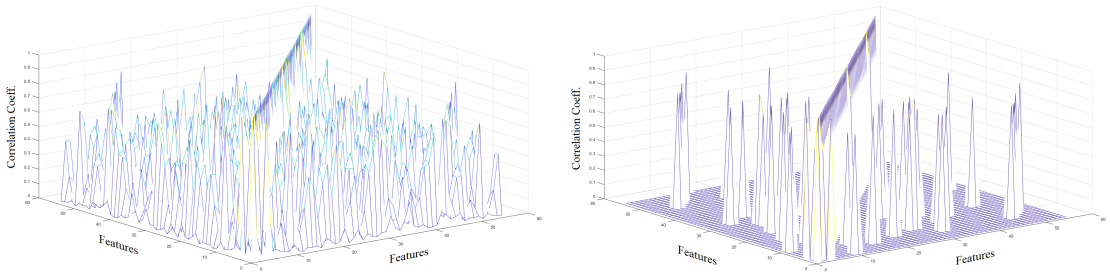


Figure 8.1: Important Channel Locations

8.2 Feature Isolation

8.2.1 Method

Feature selection is also based on binary classification of high and low response using random forest classifier and the same set of methods is used. Algorithm-2 shows the isolation of features using recursive elimination of features preserving or enhancing the accuracy. But before all of these, a correlation matrix is formed to identify highly correlated (correlation coeff. > 0.75) features and get rid of repetitions. Figure-8.2 shows feature correlation matrix and highly correlated features.



(a) (b)

Figure 8.2: (a) Feature Correlation Matrix & (b) Highly Correlated Features.

Algorithm 2: Feature Isolation

Input: EEG features from isolated channels

Output: useful features

Initialization: $N \leftarrow 0$,

useful features $\leftarrow$ {1,2,3,...,54,55}

```
1  while N ≠ size of useful features do
2      N ← size of useful features
3      X ← Random Forest accuracy for useful features
4      features ← null
5      for n ← 1 to N do
6          modified features ← EEG features for (useful features - n_th element of
             useful features)
7          accuracy ← Random Forest accuracy using modified features
8          if accuracy >= X then
9              X ← accuracy
10         else
11             features.append ( n_th element of useful features)
12     useful features ← features
```

8.2.2 Results

Based on the stated methods, Hjorth parameters and band power-based features have been found most important. Though all the band information has been proven to be important, alpha & theta band are comparatively more impactful. Table-8.1 shows the list of 10 most important features with their relative weights.

Table 8.1: 10 Most Important Features and Corresponding Weights

Rank	Feature	Weight
1	Activity	0.0513
2	Activity of last moment	0.0413
3	Complexity	0.0363
4	Complexity of last moment	0.0335
5	Minimum theta power/alpha power among moving windows	0.0283
6	Mobility	0.0247
7	Median theta power/alpha power among moving windows	0.0234
8	Maximum alpha energy percentage among moving windows	0.0228
9	Minimum delta power/theta power among moving windows	0.0219
10	Theta energy percentage	0.0217

Chapter 9

FUTURE WORK

This study is conducted on the basis of specific feature extraction from different domains and the implementation of these features in selected regression and classification-based algorithms. However, there are lots of other features that can be extracted from all the domains. Also, different algorithms can be implemented to create a better model.

Though our model is a generalized one, it can't predict a person's reaction time well based on a model trained on other participants' data. So, training with a bigger data-set of longer duration and more variation among subjects may contribute towards generalizing the model for better performance in any circumstances.

Similar experiments involving different visual tasks, levels of complexity and type of response may reveal new information about the relationship between EEG signal and reaction time.

Chapter 10

CONCLUSION

This experiment is performed on regular subjects in regular conditions without much change of mental state (unlike sleep stages) and the change of EEG pattern is very subtle. Moreover, the response may depend on eye motion, improvement of muscle memory with time or emotional parameters. So it is difficult to find any direct correlation between a particular feature and the output response. Considering the circumstances, both regression, and classification-based studies are challenging. In spite of all the stated constraints, a maximum correlation coefficient of 0.69 has been obtained from regression-based analysis and 95% of maximum accuracy from the classification of the extremes. So, it can be said that this study has shown some promising results compared to the earlier works.

Aboalayon, K. and M. Faezipour, "Multi-class svm based on sleep stage identification using eeg signal", in "IEEE Healthcare Innovation Conference (HIC)", pp. 181–184 (2014).

Aboalayon, K., M. Faezipour, W. Almuhammadi and S. Moslehpour, "Sleep stage classification using eeg signal analysis: A comprehensive survey and new investigation", Entropy **18(9)**, 272 (2016).

Ahirwal, M. and N. Londhe, "Power spectrum analysis of eeg signals for estimating visual attention", International Journal of Computer Applications **42(15)**, 34–40 (2012).

Charbonnier, S., L. Zoubek, S. Lesecq and F. Chapotot, "Self-evaluated automatic classifier as a decision-support tool for sleep/wake staging", Computers in Biology and Medicine **41(6)**, 380–389 (2011).

Cheng, M., J. Wu and T. Hung, "The relationship between reaction time and eeg activity in a cued reaction time task", Journal Of Sport & Exercise Psychology **32(s)**, 236–236 (2010).

Fraiwan, L., K. Lweesy, N. Khasawneh and H. Wenz, H.and Dickhaus, "Automated sleep stage identification system based on timefrequency analysis of a single eeg channel and random forest classifier", Computer Methods and Programs in Biomedicine **108(1)**, 10–19 (2012).

Gudmundsson, S., T. Runarsson and S. Sigurdsson, "Automatic sleep staging using support vector machines with posterior probability estimates", in "International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents", pp. 366–372 (2005).

Koley, B. and D. Dey, "An ensemble system for automatic sleep stage classification using single channel eeg signal", Computers in Biology and Medicine **42**, 1186–1195 (2012).

Liang, S.-F., C.-E. Kuo, Y.-H. Hu and Y.-S. Cheng, "A rule-based automatic sleep staging method.", in "33rd IEEE EMBS Annual International Conference of the Engineering in Medicine and Biology Society", pp. 6067–6070 (2011).

Luo, A. and P. Sajda, "Using single-trial eeg to estimate the timing of target onset during rapid serial visual presentation", in "Annual International Conference of the IEEE Engineering in Medicine and Biology Society", vol. I, pp. 79–82 (2006).

Mckay, S. M., Christine L. Luria, J. A. S. Kinney and M. S. Strauss, "Visual evoked responses, eeg's and reaction time during a normoxic saturation dive, nisat i", Undersea Biomedical Research **4(2)** (1977).

Morrell, L. K., "Eeg frequency and reaction time a sequential analysis", Neuropsychologia **4(1)**, 41–48 (1966).

Radha, M., G. Garcia-Molina, M. Poel and G. Tononi, "Comparison of feature and classifier algorithms for online automatic sleep staging based on a single eeg signal", in "36th IEEE Annual International Conference of Engineering in Medicine and Biology Society", pp. 1876–1880 (2014).

Sanders, T., M. McCurry and M. Clements, "Sleep stage classification with cross frequency coupling", in "36th IEEE Annual International Conference of Engineering in Medicine and Biology Society", pp. 4579–4582 (2014).

Sousa, T., A. Cruz, S. Khalighi, G. Pires and U. Nunes, "A two-step utomatic sleep stage classification method with dubious range detection", Computers in Biology and Medicine **59**, 42–43 (2015).

Takeda, Y., K. Yamanaka and Y. Yamamoto, "Temporal decomposition of eeg during a simple reaction time task into stimulus- and response-locked components", Neuroimage **39(2)**, 742–754 (2008).

Wu, D., B. J. Lance, V. J. Lawhern, S. Gordon, T.-P. Jung and C.-T. Lin, "Eeg-based user reaction time estimation using riemannian geometry features", IEEE Transactions on Neural Systems and Rehabilitation Engineering **25(11)**, 2157–2168 (2017).