CLASSIFYING AND PREDICTING WALKING SPEED FROM ELECTROENCEPHALOGRAPHY DATA

by

ALLEN RAHROOH
B.S. University of Cincinnati, 2017

A thesis submitted in partial fulfilment of the requirements
for the degree of Master of Science
in the Department of Mechanical and Aerospace Engineering
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Spring Term
2019

ABSTRACT

Electroencephalography (EEG) non-invasively records electrocortical activity and can be used to understand how the brain functions to control movements and walking. Studies have shown that electrocortical dynamics are coupled with the gait cycle and change when walking at different speeds. Thus, EEG signals likely contain information regarding walking speed that could potentially be used to predict walking speed using just EEG signals recorded during walking. The purpose of this study was to determine whether walking speed could be predicted from EEG recorded as subjects walked on a treadmill with a range of speeds (0.5 m/s, 0.75 m/s, 1.0 m/s, 1.25 m/s, and self-paced). We first applied spatial Independent Component Analysis (sICA) to reduce temporal dimensionality and then used current popular classification methods: Bagging, Boosting, Random Forest, Naïve Bayes, Logistic Regression, and Support Vector Machines with a linear and radial basis function kernel. We evaluated the precision, sensitivity, and specificity of each classifier. Logistic regression had the highest overall performance (76.6 +/- 13.9%), and had the highest precision (86.3 +/- 11.7%) and sensitivity (88.7 +/- 8.7%). The Support Vector Machine with a radial basis function kernel had the highest specificity (60.7 +/- 39.1%). These overall performance values are relatively good since the EEG data had only been high-pass filtered with a 1 Hz cutoff frequency and no extensive cleaning methods were performed. All of the classifiers had an overall performance of at least 68% except for the Support Vector Machine with a linear kernel, which had an overall performance of 55.4%. These results suggest that applying spatial Independent Component Analysis to reduce temporal dimensionality of EEG signals does not significantly impair the classification of walking speed using EEG and that walking speeds can be predicted from EEG data.

To my family members in the United States and Iran

ACKNOWLEDGMENTS

I would like to thank my committee chair Dr. Helen J. Huang for advising and supporting me throughout my time in the Biomechanics, Rehabilitation, and Interdisciplinary Neuroscience (BRaIN) Lab. My interest in her neuromechanical research had me intrigued to write a Master of Science thesis in Biomedical Engineering. Dr. Helen Huang and my committee member, Dr. Hsin-Hsiung (Bill) Huang, proposed to me a biomedical statistical project in August 2018 and the project immediately had me interested in pursing a thesis and ultimately decided to switch from the non-thesis to thesis track.

Also I would like to thank my fellow BRaIN lab member Cesar Castano for this research study in fluctuations in walking speed and without his data I would never be able to complete a thesis in biomedical engineering.

Finally, I would like to thank my family for providing me support throughout my graduate career.

TABLE OF CONTENTS

LIST OF FIGURES viii

CHAPTER 1: INTRODUCTION 1

1.1 Electroencephalography (EEG) 1

1.2 Dimension Reduction 2

1.2.1 Independent Component Analysis 3

1.3 Statistical Classification Methods 4

1.4 Ensemble Model 4

1.4.1 Bagging Model 4

1.4.2 Boosting Model 5

1.4.3 Random Forest Model 6

1.5 Logistic Regression Model 8

1.6 Probability Model 9

1.7 Hyper Plane Method 10

1.8 Purpose 11

CHAPTER 2: METHODOLOGY 13

2.1 EEG Data Collection 13

2.2 Pre-Processing EEG Data 13

2.3 Data Preparation 15

CHAPTER 3: RESULTS 19

3.1 Classification Results 19

CHAPTER 4: DISCUSSION & CONCLUSIONS 24

APPENDIX A: SUPPLEMENTARY DATA 27

APPENDIX B: SUPPLEMENTARY INFORMATION 41

REFERENCES 43

LIST OF FIGURES

Figure 1.1: Bagging visual representation the bag numbers represent the sampling of the training data set where 80% is the training data set and 20% is the testing data set. 5

Figure 1.2: Boosting visual representation where unlike in figure 1.1 boosting applies higher weights towards the better performing classifications and misclassified data to boost the classification rate. 6

Figure 1.3: Random Forest Algorithm Representation where the training data is broken down using classification trees and the classifications are grouped together for a final prediction. 7

Figure 1.4: Simulated data representing a logistic regression model over the data set. . 8

Figure 1.5: Simulated Data separated into three groups: blue, green, and red. The black line represents the Bayes Decision Boundary where the classification is separated between the three groups. The blue grid area were the classifier predicts as the blue area, the green grid area were the classifier predicts as the green group and the red grid area were the classifier predicts as the red group. 9

Figure 1.6: Simulated data with a hyper plane separating the data and the closest points are the support vectors. 11

Figure 2.1: Breakdown of the EEG Data Set (2019, Huang). 14

Figure 2.2: Breakdown of the classification methods (2019, Huang). 16

Figure 2.3: 2 x 2 Confusion Table of True & False Positives and True & False Negatives. 18

Figure 3.1: Confusion Tables for Subject 1. 21

Figure 3.2: Mean Confusion Table for All Subjects. 22

Figure 3.3: Bar plot of precision, sensitivity, and specificity values for subject 1 and the average across subjects with +1 standard deviation bars. 23

Figure A.1: Confusion Tables for Subject 2. 29

Figure A.2: Precision, Sensitivity, and Specificity Values for Subject 2. 30

Figure A.3: Confusion Tables for Subject 3. 31

Figure A.4: Precision, Sensitivity, and Specificity Values for Subject 3. 32

Figure A.5: Confusion Tables for Subject 4. 33

Figure A.6: Precision, Sensitivity, and Specificity Values for Subject 4. 34

Figure A.7: Confusion Tables for Subject 5. 35

Figure A.8: Precision, Sensitivity, and Specificity Values for Subject 5. 36

Figure A.9: Confusion Tables for Subject 6. 37

Figure A.10: Precision, Sensitivity, and Specificity Values for Subject 6. 38

Figure A.11: Confusion Tables for Subject 7. 39

Figure A.12: Precision, Sensitivity, and Specificity Values for Subject 7. 40

CHAPTER 1: INTRODUCTION

1.1 Electroencephalography (EEG)

Electroencephalography (EEG) has been studied by past researchers [Edla et al. (2018)] [Sun (2007)] to understand how the brain works during various tasks like taking a test, walking, and sleeping. An EEG signal is an electrical potential that is recorded non-invasively from the human scalp area. The EEG signal is originated from the brain, which are made of billions of cells called neurons. These neurons have axons which release neurotransmitters. The dendrites of the neurons receive these neurotransmitters from the axons of other neurons, which causes a electrical polarity change. This polarity change is the electrical potential that the EEG is recording from the human scalp area in volts. The neurotransmitters can be activated from various activities such as taking a test, walking, and sleeping. When a person is taking a test they have to constantly use different areas of the brain to answer the questions correctly, which inhibits more brain activity then say walking were the person is at a constant speed and is not answering questions.

Due to the nature of the EEG signal it contains a mixture of signals from eye movement (ECoG), muscle activity (EMG), brain activity (EEG), and heart activity (ECG). The mixture of signals makes EEG hard to collect and analyze. Past researchers [Lin et al. (2008)] [Artoni et al. (2018)] have looked at recording EEG mostly during non-movement tasks where the subject sits still and completes a task to minimize the amount of EEG movement artifact. Recently researchers have been looking into collecting mobile EEG data [Makeig (2009)], but with the mixture of signals it is very challenging to separate out the brain (cortical) activity from the non-cortical signals like muscle, cardiac, eye, and any other surrounding noise that would be recorded with the EEG unlike non-movement tasks. There has yet been a definitive solution for removing movement artifact from mobile EEG data [Oliverira et al. (2016)] [Nordin et al. (2018)].

To attempt to solve the challenge of separating out EEG signals past researchers have used a temporal Independent Component Analysis [Calhoun et al. (2003)] approach, which seeks to separate out the EEG channels into different signals like movement artifact, EMG, EOG, and ECoG. Other researchers have attempted to use a spatial Independent Component Analysis (2019, Huang) which reduces the temporal dimensionality of the EEG data. Researchers also believe that there is a correlation between gait cycle and cortical activity [Gwin et al. (2010)] [Oliverira et al. (2016)]. As the person is walking there is less brain activity as to when a person is running [Gwin et al. (2010)]. With this knowledge clinicians could potentially diagnosis diseases like Parkinson's, where the neurons in the brain break down or die and affects motor control like loss of balance and freezing of gait [Handojoseno et al. (2015)]. Studies have shown there are EEG differences in subjects with Parkinson's and subjects without Parkinson's [Handojoseno et al. (2015)] [Cozac et al. (2016)]. Mobile EEG [Elda et al. (2018)] could help scientists and clinicians better understand how the brain functions at different walking speeds for healthy younger, older adults, and neurological populations such as Parkinson's disease.

1.2 Dimension Reduction

EEG data can have a high temporal dimension. To reduce the temporal dimension previous researchers have implemented a principal component analysis (PCA) [Artoni et al. (2018)] and independent component analysis (ICA) [Gwin et al. (2010)] [Snyder et al. (2015)]. PCA seeks to find the uncorrelated sources [Holland (2018)], where as ICA seeks to find the independent source components [Karhunen (2001)]. These dimension reduction methods can be used to extract various EEG sources such as movement artifact, ECoG, EMG, and ECG.

1.2.1 Independent Component Analysis

Independent Component Analysis (ICA) is a statistical and computational method for dimension reduction and extracting source signals. The assumptions that ICA uses are that data variables are either linear or non-linear mixtures of latent variables. The latent variables are variables that are not directly observed and are assumed to be mutually independent and non-gaussian. These latent variables are the independent components or sources [Karhunen et al. (2001)]. Using ICA the independent components can be extracted.

The ICA is derived by $X = AS$ where $X = (X_1, ..., X_m)^T$ is the m x 1 continuous-valued random vector of the observable signals, $A = a_{ij}$ is the unknown constant (non random) and invertible square matrix mixing matrix of size m x m and $S = (S_1, ..., S_m)^T$ is the m x 1 continuous-valued random vector of the m unknown source signals to be recovered [Karhunen et al. (2001)]

ICA is a powerful method to use for EEG data because ICA can help extract and separate source signals such as movement artifact, muscle activity (electromyography, EMG), and cortical activity [Oliveira et al. (2016)]. How well ICA can extract and separate the source signals depends on the quality of the measurement of the observable data signals, the characteristics of the sources, and specifics of the mixture of sources.

The two types of ICA are temporal and spatial Independent Component Analysis (tICA & sICA). tICA assumes independence in time so that the original voices from the "cocktail" party analogy can be extracted from the mixture. This method has been shown to have the ability to partially separate motion artifact from electrophysical signals [Snyder et al. (2015)]. In contrast, sICA assumes spatially independent components where the high EEG temporal dimension is reduced [Huang et al. (2019)].

1.3 Statistical Classification Methods

The classification methods after dimension reduction that we used in this thesis are Bagging, Boosting, Random Forest, Logistic Regression, Naive Bayes, and Support Vector Machines with linear and radial basis function kernels. For this thesis the input variable was the EEG 128 channel data from each subject and the output variable were the corresponding speeds (0.5 m/s, 0.75 m/s, 1.0 m/s, 1.25 m/s and a self-paced speed). A self-paced speed is where the treadmill will speed up if the subject is going too fast and slow down if the subject is going too slow to maintain the subjects position in the center of the treadmill.

1.4 Ensemble Model

Ensemble methods are where multiple learning algorithms are used to obtain better predictive performance than using a solo learning algorithm. Ensemble methods consist of the models: bagging (bootstrap aggregation), boosting (adaboost), and random forest.

1.4.1 Bagging Model

The bagging model is an abbreviation for bootstrapping aggregation, which creates multiple replicates of data from the training data set. Bootstrapping is the process of taking random samples from the training set until n number of bags are generated. Once the data is put into a bag the data is then put back in the original training data set, which can result in replicate data points in other bags. Once n bags are generated classification tress are used to calculate a classification rate. These classification rates are then aggregated, which means to group the classification rates and take an equally weighted average. Then a majority vote is determined for a final prediction. Figure

1.1 shows the representation of the bagging model.

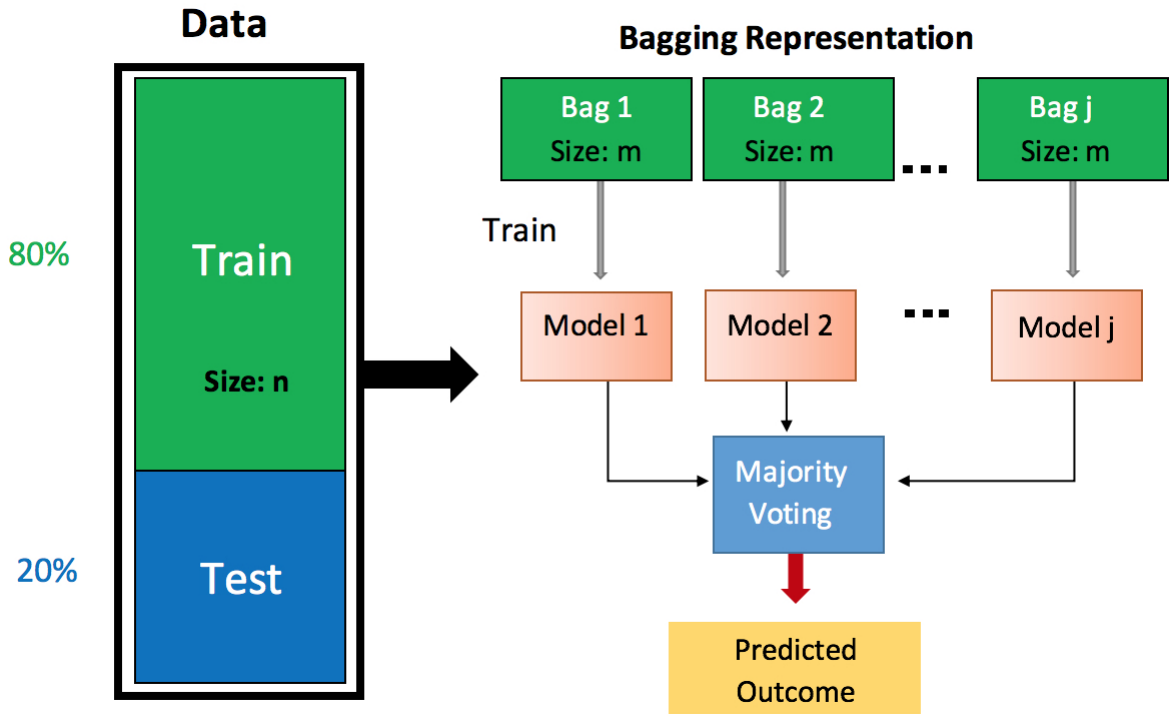

Figure 1.1: Bagging visual representation the bag numbers represent the sampling of the training data set where 80% is the training data set and 20% is the testing data set.

1.4.2 Boosting Model

The boosting model uses misclassified data points to boost the accuracy of the total model. The boosting algorithm also known as adaboost (adapative boosting) takes a random subset of training data but instead of putting the data back into the training data it puts higher weights on the misclassified points. This is repeated until n subsets are generated. After n subsets are generated a weighted average is taken instead of equally weighted average as the bagging model does. This weighted average boosts the overall classification and prediction rate. Figure 1.2 shows the representation of the boosting model.

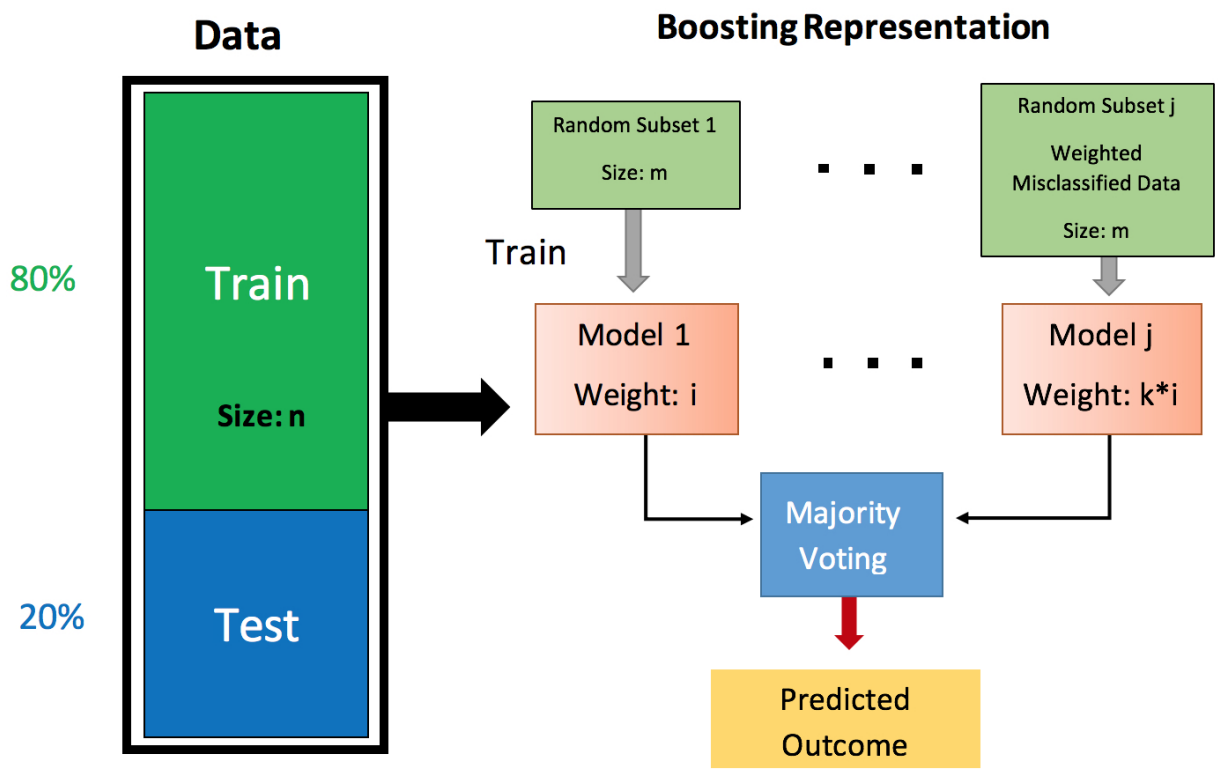

Figure 1.2: Boosting visual representation where unlike in figure 1.1 boosting applies higher weights towards the better performing classifications and misclassified data to boost the classification rate.

1.4.3 Random Forest Model

The random forest model is an ensemble of classification trees used for classification and prediction of the training data set. A decision tree breaks down the data into smaller subsets until a "stump" is reached in the tree. Hence the name random forest means multiple trees, which makes a forest. The classification rates are then aggregated just like in the bagging model and a final prediction is calculated. Figure 1.3 shows the representation of the random forest model.

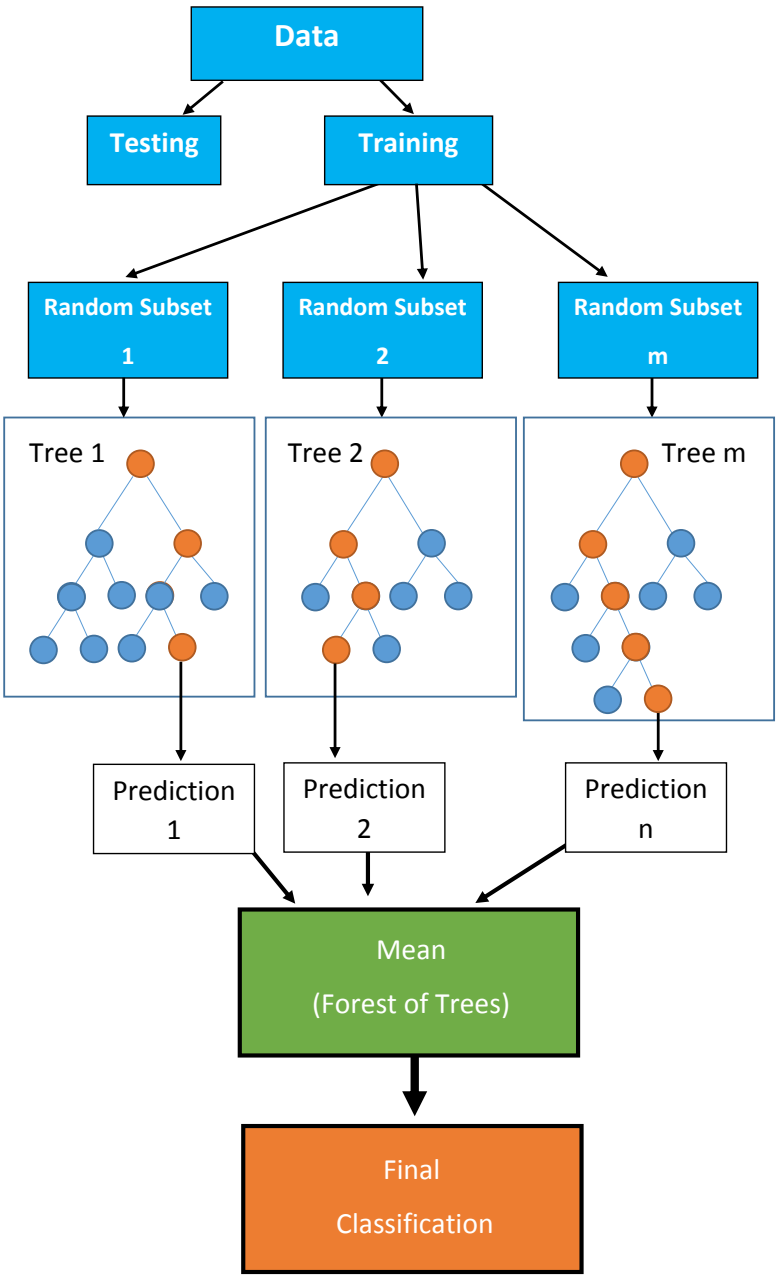

Figure 1.3: Random Forest Algorithm Representation where the training data is broken down using classification trees and the classifications are grouped together for a final prediction.

1.5 Logistic Regression Model

A logistic regression model estimates the correlation between the dependent (target) and independent variables (predictor) for numerical data. Logistic Regression categorizes into two models being: binomial and multinomial logistic regression. Binomial is when there are only two dependent numerical variables like Age/Weight. Multinomial Logistic Regression is when there are more then two dependent numerical variables like Age/Weight/Sex. Figure 1.4 shows an example of a fitted logistic regression model over a simulated data set.

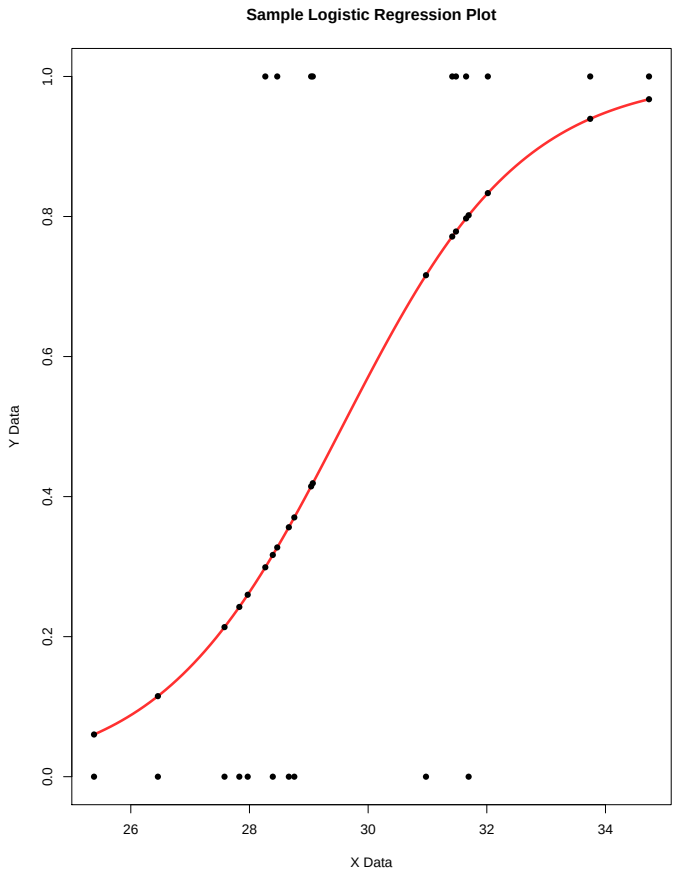

Figure 1.4: Simulated data representing a logistic regression model over the data set.

1.6 Probability Model

The Naive Bayes model is derived from Bayes' Rule (Bayes' Theorem or Bayes' Law) $P(Y = j|X = x_0)$. Bayes' Rule assumes conditional probability when Y = j there is a given observed predictor vector x_0 [James et al. (2013)]. This is known as the Bayes' classifier. The Bayes Decision Boundary as shown in figure 1.6 represents the probability classification of each group. The classification is then based on the Bayes Decision Boundary with observations either falling in the red, green or blue category.

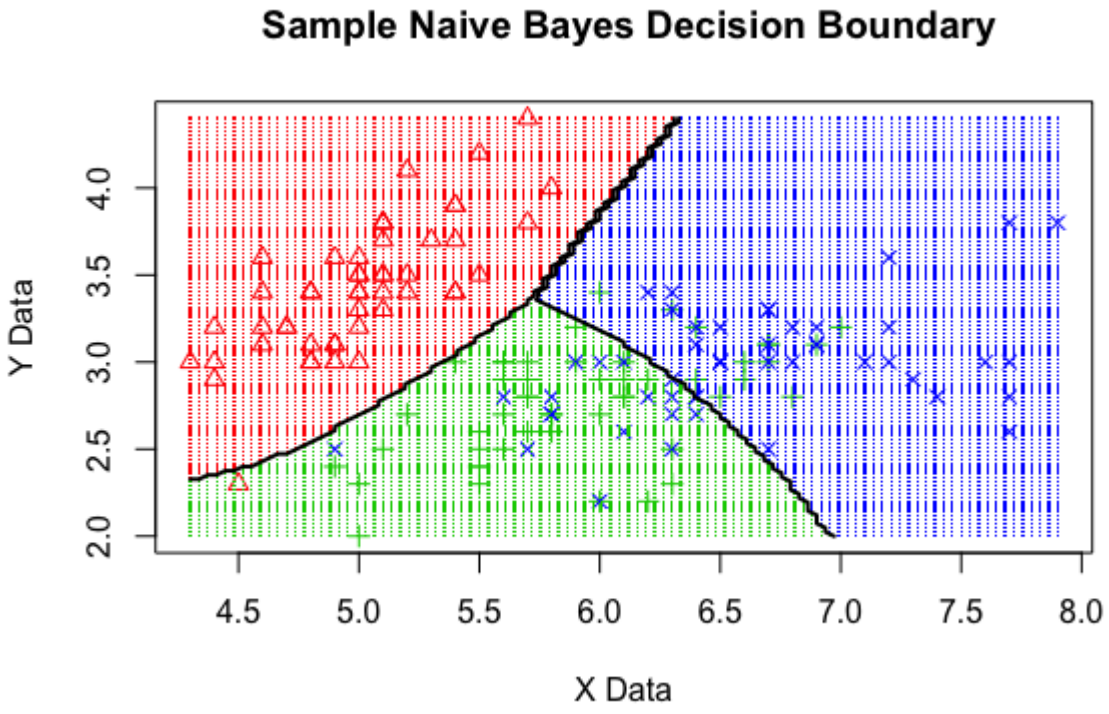

Figure 1.5: Simulated Data separated into three groups: blue, green, and red. The black line represents the Bayes Decision Boundary where the classification is separated between the three groups. The blue grid area were the classifier predicts as the blue area, the green grid area were the classifier predicts as the green group and the red grid area were the classifier predicts as the red group.

1.7 Hyper Plane Method

The last statistical classification approach uses a hyper plane approach, which is known as a Support Vector Machine. Support Vector Machines [Tong et al. (2000)] [Vapnik et al. (1982)] have been used in previous research studies for data classification [Huang et al. (2019)] [(Guyon et al. (2002)]. The support vector machine is a discriminative classifier that is a hyper plane to separate the data and predict the classification rate. A hyper plane separates the data by a maximal margin as shown in the left graph of figure 1.6. Given a training data set $\{x_1 \cdots x_n\}$ in the space of $X \subseteq \mathbb{R}^d$ also with the training data set is a set of data labels $\{y_1 \cdots y_n\}$ where $y_i \in \{-1, 1\}$.

As shown in figure 1.6 the vectors formed from the data set are either labeled -1 or 1 and the vectors that lie closest to the to the hyper plane are known as the support vectors, which are used for the classification. Support vector machines either use an inductive or transductive hyper plane approach where the inductive hyper plane builds a decision based solely on the training data set, where as the transductive hyper plane builds a decision based on the training and testing data sets [Kasabov et al. (2003)].

With a support vector machine a kernel function is used. A kernel is a similarity function where it quantifies the similarity between two objects and is often used for pattern analysis. For example, for EEG data a kernel would take a voltage value at a giving time and then quantify it with a cognitive task like level ground walking to predict the speed of the subject. The most common kernels used for support vector machines are the linear and radial basis function kernels. The linear kernel is simple the inner product of the two observations $K(x_i, x_{i'}) = \sum_{j=1}^{p} x_{ij} x_{i'j}$, and best used when the data can be linearly separated. The radial basis function kernel uses the tuned parameters: γ & cost (c) for cross-fold validation $K(x_i, x_{i'}) = exp(-\gamma \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2)$, and is best used when the data can not be linearly separated. The γ parameter controls the kernel (decision boundary/hyperplane). A high γ can create a decision boundary with a low margin, but can overfit

the data. A low γ can create a decision boundary with a high margin, but can misclassify points that are within the margin. The c parameter determines the cost (penalty) of misclassification of the training data points. A high c value gives a high penalty towards misclassified points and a smaller margin is used for classification. For a low c value it gives a low penalty towards the misclassified points and a larger margin is used for classification.

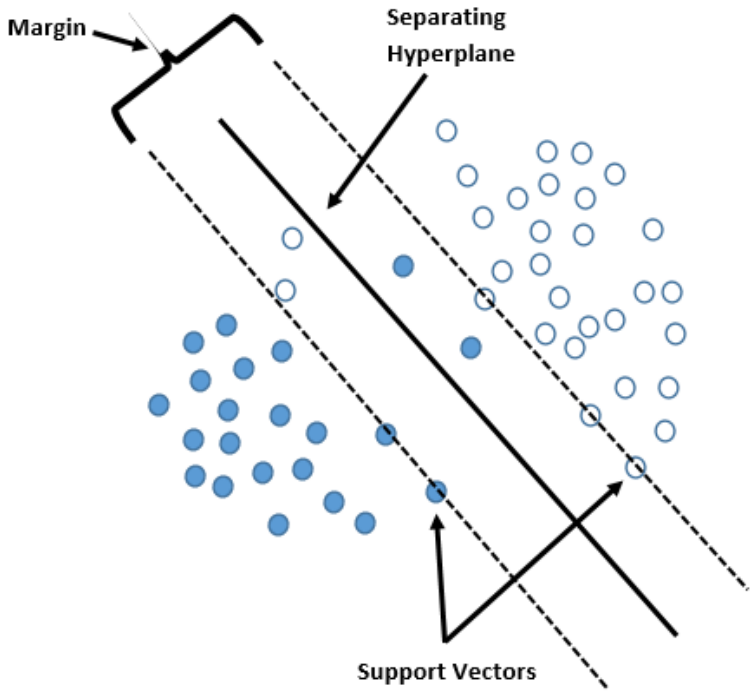

Figure 1.6: Simulated data with a hyper plane separating the data and the closest points are the support vectors.

1.8 Purpose

The purpose of this thesis was to develop a procedure to classify and predict walking speed from EEG data recorded as subjects walked on a treadmill using a range of speeds. For the first part of the procedure, we used a spatial Independent Component Analysis (sICA) to reduce the temporal

dimension of the EEG data. Then, we trained our training data set using seven statistical classification methods to the temporal reduced EEG data: Bagging, Boosting, Random Forest, Naïve Bayes, Logistic Regression, and Support Vector Machines with a linear and radial basis function kernels. After the classification methods we generated confusion tables using the testing data set, and calculated the precision, sensitivity, and specificity to measure each of the classification methods overall performance.

CHAPTER 2: METHODOLOGY

2.1 EEG Data Collection

We used EEG data recorded using a 128-channel system (BioSemi, ActiveTwo, Amsterdam, Netherlands) as healthy young adults (n = 7, 18 - 35 years old) walked on a treadmill using five different walking speeds, 0.5 m/s, 0.75 m/s, 1.0 m/s, 1.25 m/s, and a self-paced speed. Each EEG channel was filled with a non conductive gel and the electrodes were placed accordingly based on cortical brain location. The dimension reduction and classification methodology is adapted from previous work done by (2019, Huang), where they applied sICA and classification models to a 256-channel EEG artifact data.

2.2 Pre-Processing EEG Data

The raw EEG data was then preprocessed into MATLAB and a 1 Hertz (Hz) high pass filter was applied. The data was then converted into text files (.csv) for dimension reduction and classification modeling in R studio. The data was imported into R studio with each trial being five minutes (300 seconds) with a 512 Hz sampling rate per second, which on average was 150,000 time samples. This segment was then cut down to three 50,000 temporal point segments [0:50,000, 50,000:100,000, 100,000:150,00], and the 2^{nd} segment [50,000:100,000] was chosen to omit the treadmill speeding up and the subject adjusting to the speed. This data segment was further cut into five 10,000 temporal points per subject per speed with an added speed column with 0.5 m/s , 0.75 m/s , 1.0 m/s , 1.25 m/s, and Self-Paced for a total of 25 data sets per subject.

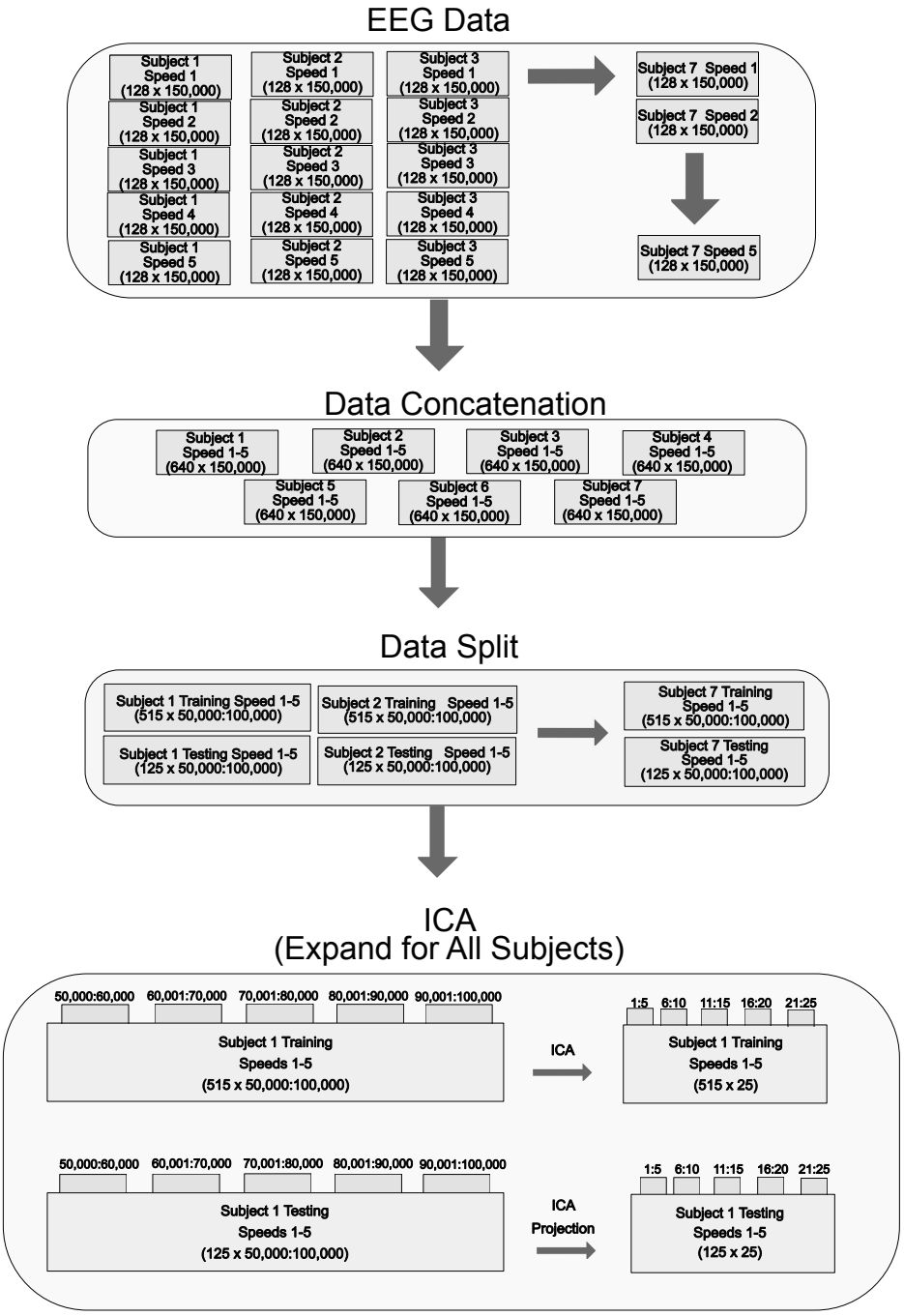

Figure 2.1: Breakdown of the EEG Data Set (2019, Huang).

2.3 Data Preparation

To perform a five-fold analysis we chose to use a training and testing data sets. The training data set is used to train the model. The testing data set is used to validate the results of the prediction and classification from the training data set. The training set consisted of 515 (103 per speed) EEG channels per each subject and 125 (25 per speed) EEG channels for the testing set. A sICA was performed using eegica from the package eegkit [Helwig (2015)] the training data set was reduced from 50,000 temporal points to 25 temporal points per subject for a matrix of dimensions 515 x 25. sICA was not used on the testing set because the source (S) matrix was found by doing the matrix cross product tcrossprod using the remaining 125 EEG channels after randomly chosen 515 EEG channels as the training data set before sICA, and the W matrix extracted after sICA from the training data set. The W matrix is chosen to maximize the negentropy, measure of distance to normality, approximation under the constraints that W is an orthonormal matrix [Helwig (2015)]. The matrix cross product was then computed to obtain the source matrix for the testing data set. This resulted in the testing data set to be reduced from a dimension of 125 x 50,000 to 125 x 25.

After applying sICA the EEG data is processed through the following classifiers: Bagging, Boosting, Random Forest, Naïve Bayes, Logistic Regression, and Support Vector Machine with Linear and Radial Basis Function (RBF) kernels.

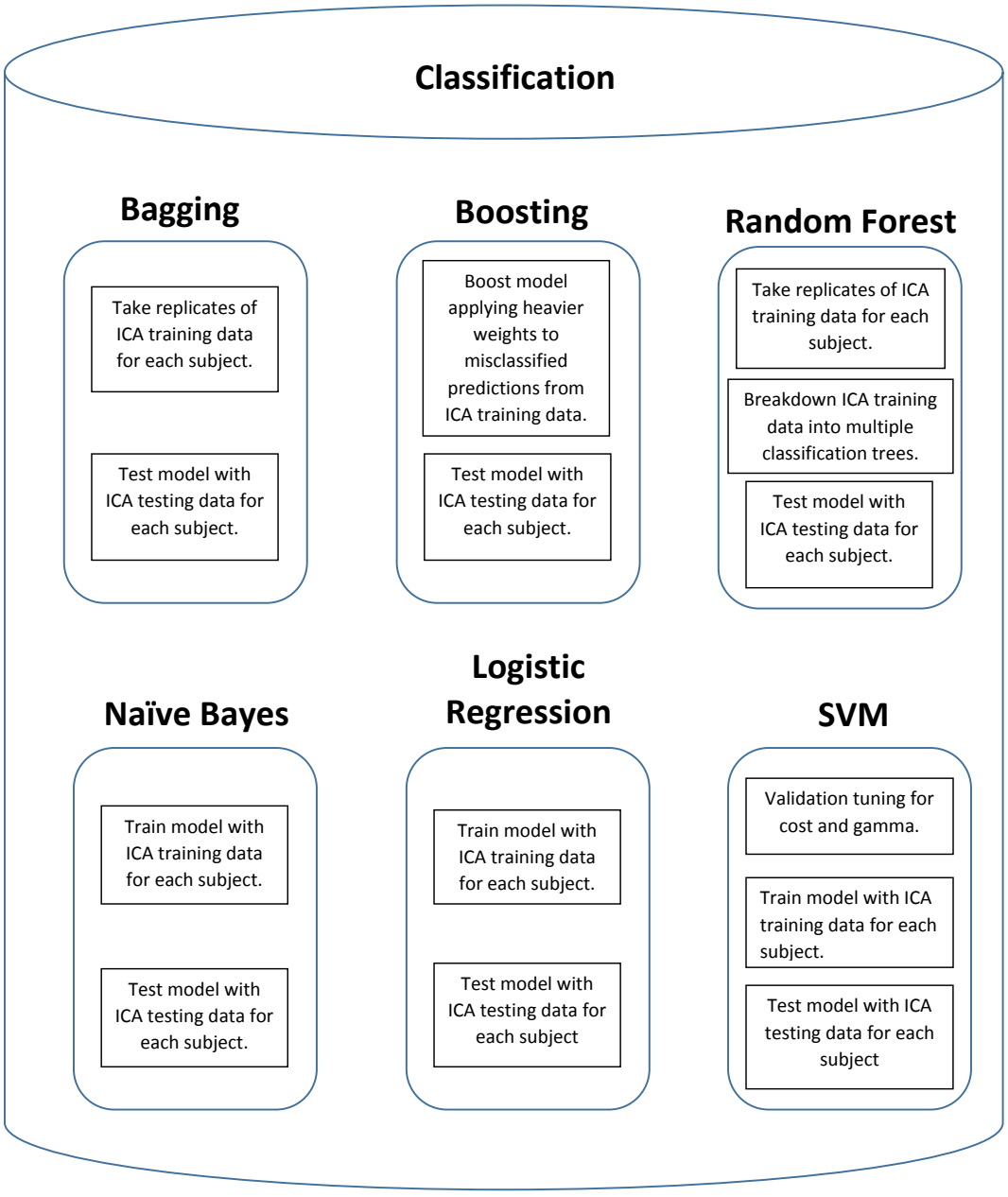

Figure 2.2: Breakdown of the classification methods (2019, Huang).

First we train the training data using the bagging classification model using the packages tree [Ripley (2018)] and randomForest (2012, Laiw). Next we trained the data using the boosting classification model, which was ran using the package adabag [Cortes et al. (2018)], which contains the boosting command. Next we trained the data with the random forest classification model, which utilizes the packages tree [Ripley (2018)] and randomForest [Liaw et al. (2018)]. Next we trained data using the naive bayes classification model with the package naivebayes [Majka (2017)]. Next we trained the data using the logistic regression classification model with the general linear model (glm) command and the package nnet [Venables et al. (2016)], which contains the multinom command. Finally, we trained the data using the support vector machine classification model the packages ISLR [Gareth et al. (2000)] and e1071 [Meyer et al. (2017)] were utilized. The support vector machine classifier has two methods one being with a linear kernel with cost (c) being set at 0.01, and the other being a radial basis function kernel with c set at 0.5 and γ set at 0.1. For the support vector machine models the c and γ values can be tuned to obtain various classification rates. After the training data was trained using the seven classifiers we generated 5 x 5 confusion tables using the testing data set.

From these confusion tables we calculated three statistical parameters that measure the overall performance of each confusion table. These parameters are precision, sensitivity, and specificity which are calculated by equation 2.1, 2.2, & 2.3.

$$Precision = \frac{True\ Positives}{Actual\ Positives} \tag{2.1}$$

$$Sensitivity = \frac{True\ Positives}{True\ Positives + False\ Negatives} \tag{2.2}$$

$$Specificity = \frac{True\ Negatives}{True\ Negatives + False\ Positives} \tag{2.3}$$

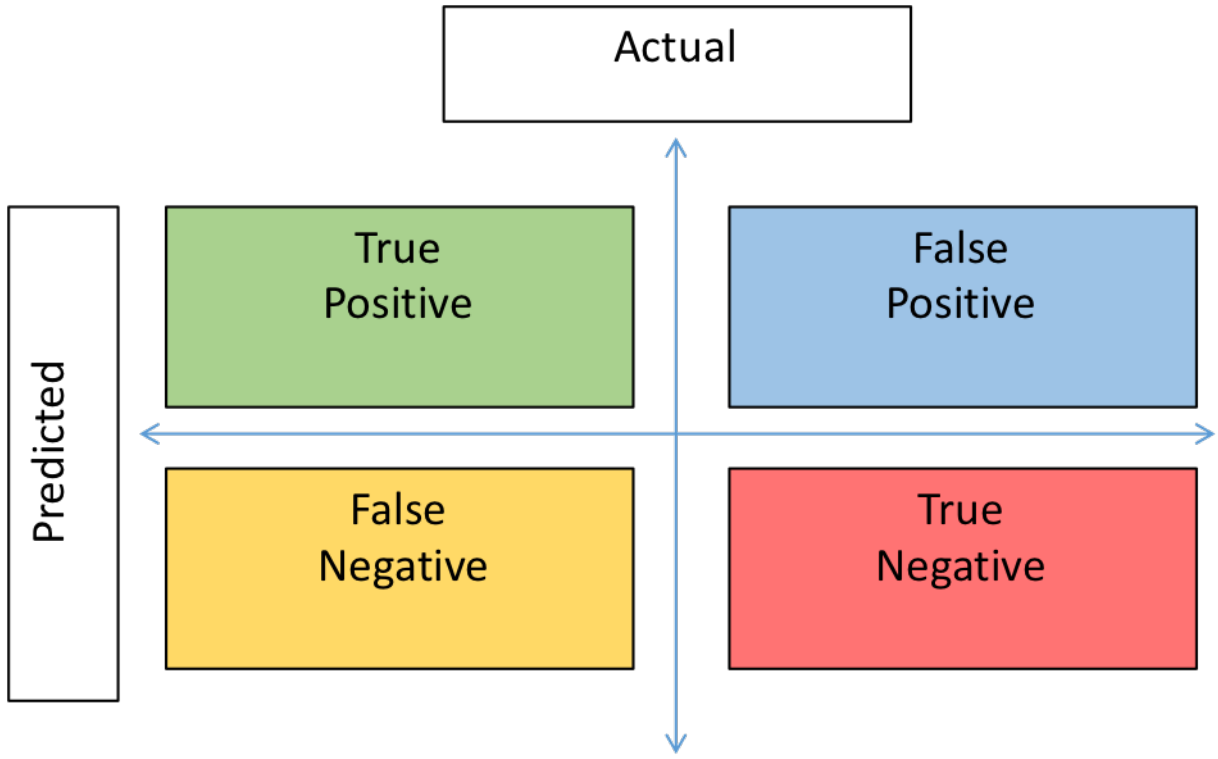

Figure 2.3: 2 x 2 Confusion Table of True & False Positives and True & False Negatives.

As shown in figure 2.3 is a 2 x 2 confusion table with True Positive, False Positive, False Negative, & True Negative with the predicted data as the rows and actual data as the columns. For a true positive the outcome is where the model correctly predicts the positive class. For a true negative the outcome is where the model correctly predicts the negative class. For a false positive the outcome is where the model incorrectly predicts the negative class. For a false negative the outcome is where the model incorrectly predicts the negative class. After we calculated the precision, sensitivity, and specificity for all subjects we calculated the average over all the subjects and generated an average confusion table and an average statistical parameter bar plot with positive standard deviation bars.

CHAPTER 3: RESULTS

3.1 Classification Results

The confusion tables show how well each classifier predicted the subjects speed using the testing data with 125 (25 per speed) EEG channels to test how well the training data classified. The predicted speeds are the rows and the actual speeds are the columns. Figure 3.1 shows the confusion table for subject 1 for all the classification methods. The diagonal components (True Positives) of each confusion table represent how well the the classification method performed and a color scale is used with 25 being the darkest (best) and 0 being the lowest (worst). The values surrounding the diagonal are misclassified predictions (True Negatives, False Negatives, & False Positives), and for subject 1 there was minimum misclassification with the highest misclassified prediction being 4 points for the support vector machine with a linear kernel. Figure 3.2 shows the mean confusion tables for all subjects and classifiers. The same color scale is used as figure 3.1 with 25 being the best classification and 0 being the worst classification. Decimal values are given for figure 3.2 because it is an average taken over 7 subjects unlike the individual subjects were an integer value is given. The remaining of the subjects confusion tables can be found in APPENDIX A: SUPPLEMENTARY DATA.

From the confusion tables we calculated precision, sensitivity, and specificity to measure the classification performance per each classification method. Figure 3.3 shows the precision, sensitivity, and specificity for subject 1 and all subjects. For subject 1 bagging, random forest, and support vector machines with a linear and radial basis function kernels had the highest precision (96%) followed by naive bayes (92%), boosting (88%), and logistic regression (68%). Bagging, boosting, random forest, naive bayes, and a support vector machine with a radial basis function kernel showed the highest sensitivity (100%) followed by logistic regression (85%), and support vec-

tor machine with a linear kernel (75%). For specificity bagging had the highest value (92.31%) followed by random forest, support vector machine with a radial basis function kernel (88.89%), naive bayes (88.46%), boosting (75.86%), support vector machine with linear kernel (63.16%), and logistic regression (39.53%). The remaining of the subjects classification performance values can be found in APPENDIX A: SUPPLEMENTARY DATA.

For the classification performance average across subjects the logistic regression showed the best classification performance for precision (86.3% + 11.7%) followed by naive bayes (82.3% +/- 36.4%), support vector machine with a radial basis function kernel (75.4% +/- 39.1%), random forest (72% +/- 41.1%), bagging (68% +/- 42%), boosting (66.3% +/- 41.4%), and support vector machine with a linear kernel (59.4% +/- 46.1%). The logistic regression also performed best for sensitivity (88.7% +/- 8.7%) followed by support vector machine with a radial basis function kernel (84.5% +/- 27.4%), boosting (82.1% +/- 36.6% +/- 36.6%), bagging (80.2% +/- 36.5%), random forest (66.3% +/- 46%), support vector machine with a linear kernel (63.2% +/- 44.4%), and naive bayes (62.1% +/- 45.8%). The support vector machine with a radial basis function kernel had the best specificity (60.7% +/- 39%) followed by random forest (60.1% +/- 38.8 %), naive bayes (58.8% + 36.5%), bagging (57.1% + 39.5%), boosting (55% +/- 40.4%), logistic regression (54.8% + 21.3 %), and support vector machine with a linear kernel (43.4% + 36.1%).

Subject 1					
Bagging					
Predicited/Actual Speeds					
	Speed 1	Speed 2	Speed 3	Speed 4	Speed 5
Speed 1	24	0	0	0	0
Speed 2	0	25	0	1	0
Speed 3	0	0	25	0	0
Speed 4	0	0	0	24	0
Speed 5	1	0	0	0	25
Boosting					
Speed 1	22	0	0	0	0
Speed 2	0	25	0	1	0
Speed 3	3	0	24	0	0
Speed 4	0	0	0	23	0
Speed 5	0	0	1	0	24
Random Forest					
Speed 1	24	0	0	0	0
Speed 2	0	24	0	1	0
Speed 3	0	0	25	0	0
Speed 4	0	0	0	24	0
Speed 5	1	1	0	0	25
Naïve Bayes					
Speed 1	23	0	0	0	0
Speed 2	1	25	0	1	0
Speed 3	0	0	25	0	0
Speed 4	0	0	0	24	0
Speed 5	1	0	0	0	25
Logistic Regression					
Speed 1	17	3	0	0	0
Speed 2	0	23	0	0	0
Speed 3	0	0	25	0	0
Speed 4	2	9	0	9	0
Speed 5	1	3	0	0	25
Linear SVM					
Speed 1	24	1	4	3	0
Speed 2	0	24	1	4	0
Speed 3	0	0	20	0	0
Speed 4	1	0	0	18	0
Speed 5	0	0	0	0	25
SVM RBF					
Speed 1	24	0	0	0	0
Speed 2	0	24	0	1	0
Speed 3	0	0	25	0	0
Speed 4	0	0	0	24	0
Speed 5	1	1	0	0	25

Color Scale
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25

Figure 3.1: Confusion Tables for Subject 1.

Mean Confusion Tables Across Subjects					
Bagging					
Predicted/Actual Speeds					
	Speed 1	Speed 2	Speed 3	Speed 4	Speed 5
Speed 1	17.00	0.14	0.86	0.00	1.14
Speed 2	3.29	21.14	2.14	1.57	2.57
Speed 3	2.57	2.57	20.43	1.00	1.57
Speed 4	1.86	1.00	1.57	22.43	1.14
Speed 5	0.29	0.14	0.00	0.00	18.57
Boosting					
Speed 1	16.57	0.00	0.14	0.00	0.57
Speed 2	1.86	20.14	1.43	1.14	2.43
Speed 3	4.00	2.43	19.86	1.43	2.86
Speed 4	2.57	2.43	3.14	22.14	1.57
Speed 5	0.00	0.00	0.43	0.14	17.57
Random Forest					
Speed 1	18.00	0.00	0.86	0.00	0.71
Speed 2	3.43	21.29	2.57	1.86	3.29
Speed 3	2.71	2.86	20.86	1.14	1.86
Speed 4	0.71	0.71	0.71	22.00	0.86
Speed 5	0.14	0.14	0.00	0.00	18.29
Naïve Bayes					
Speed 1	20.57	3.43	4.43	1.86	2.86
Speed 2	0.29	16.43	0.43	0.43	0.71
Speed 3	0.00	0.86	16.29	0.14	0.14
Speed 4	3.86	4.00	3.14	21.86	3.29
Speed 5	0.29	0.29	0.71	0.57	18.00
Logistic Regression					
Speed 1	21.57	0.71	0.43	0.86	0.71
Speed 2	2.00	19.86	1.14	1.14	0.57
Speed 3	1.14	1.86	20.71	1.00	0.29
Speed 4	1.57	2.86	0.14	19.43	0.29
Speed 5	0.71	1.29	1.29	0.86	21.43
Linear SVM					
Speed 1	14.86	0.57	0.57	0.43	0.29
Speed 2	3.71	19.86	5.00	4.29	3.86
Speed 3	5.86	4.43	19.00	4.00	5.57
Speed 4	0.29	0.00	0.00	16.14	0.00
Speed 5	0.43	0.14	0.43	0.00	15.29
SVM RBF					
Speed 1	18.86	0.71	1.71	0.14	0.29
Speed 2	3.71	22.14	3.71	4.00	2.86
Speed 3	0.00	0.00	16.86	0.14	0.00
Speed 4	2.14	2.00	2.00	19.86	1.71
Speed 5	0.29	0.14	0.71	0.86	20.14

Color Scale
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25

Figure 3.2: Mean Confusion Table for All Subjects.

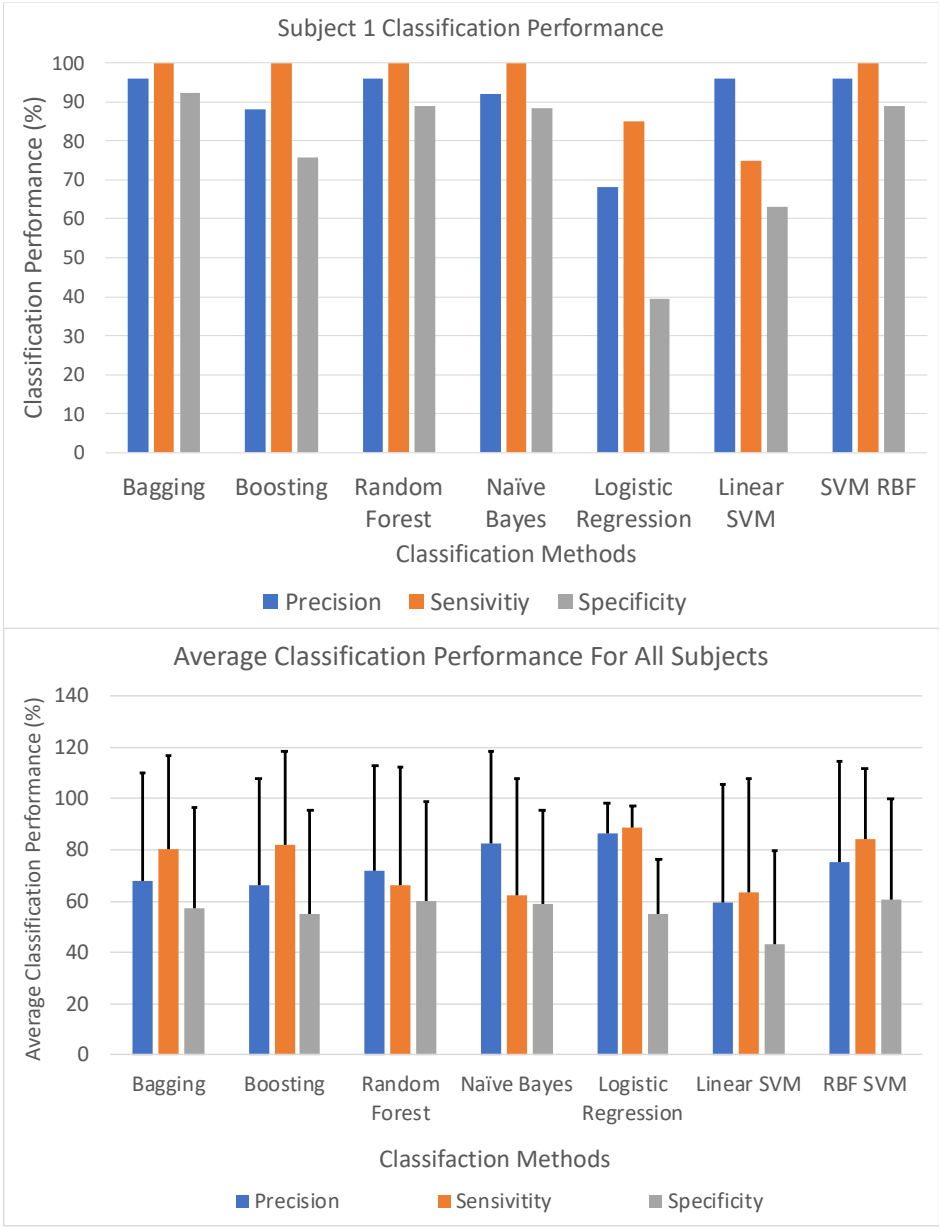

Figure 3.3: Bar plot of precision, sensitivity, and specificity values for subject 1 and the average across subjects with +1 standard deviation bars.

CHAPTER 4: DISCUSSION & CONCLUSIONS

In this thesis we sought to reduce the temporal dimension of EEG data recorded while walking. We applied a spatial Independent Component Analysis to reduce temporal dimensionality, and then applied seven classification methods being Bagging, Boosting, Random Forest, Naive Bayes, Logistic Regression, and Support Vector Machines with a linear and radial basis function kernel. After the training data was trained through the seven classifiers we generated the confusion tables and corresponding precision, sensitivity, and specificity values. We found that the logistic regression classifier performed best for precision and sensitivity and the support vector machine with a radial basis function kernel classifier performed best for specificity.

The main finding was that gait speeds could be predicted from EEG data with greater then 55.4% performance. This performance rate is low since for some of the confusion tables 0 out of the 25 EEG channels used for the testing data set were not classified at all as the actual speed. This resulted in low mean and high standard deviation values for all the classifiers, which suggests that some subjects had better quality data collections. This further demonstrates that EEG data can be classified to determine different types of activities such as sitting still while completing a task and dynamic activities like walking and running. This implies that only using a 1 Hz high pass filter does not significantly impair classification of walking using EEG and that walking speeds can be predicted from EEG data. Previous papers have implemented each of the seven classification methods. For bagging and boosting the abstract from the conference Multiple Classifier Systems [Sun (2007)] looked at classifying EEG signals during three mental imagery tasks being: the imagination of repetitive self-paced left hand movements, imagination of repetitive self-paced right hand movements,and generation of different words beginning with the same random letter. For bagging they got an overall accuracy of 56.31% with a standard deviation of 12.60%. For boosting they got an overall accuracy of 55.49% with a standard deviation of 11.96%. For random forest the paper

[Edla et al. (2018)] looked at the classification of EEG data for different mental states being: solving a mathematics problem (concentration) and a period of mediation (resting) with the subjects eyes closed. They found that the random forest classifier had a precision of 80% for concentration and 70% for the resting phase. For the naive bayes classification model the paper [Machado et al. (2013)] looked at EEG data during imaginary movement being the movement of both the right and left hand, which is similar to the imaginary tasks in the paper [Sun (2007]. They found the naive bayes had an overall performance of 78.57%. For the logistic regression model the paper [Subasi et al. (2005)] looked at the classification of EEG data from epileptic and normal subjects. They found the logistic regression model had an overall specificity of 90.3% and sensitivity of 89.2%. For the support vector machine classification model the paper [Lin et al. (2008)] looked at the classification of EEG data during emotional music listening and they got an overall accuracy of 92.73% with a standard deviation of 2.09%. These findings show that it is possible to classify EEG during different imagery, mental, dynamic, and static tasks.

We found logistic regression performed best for the precision and sensitivity with mean values of 86.3% +/- 11.7% and 88.7& +/- 8.7%. Also we found the support vector machine with a radial basis function kernel had the best specificity with values of 60.7 +/- 39.1%. The logistic regression outperformed the other classification methods because it does not have to be tuned like the support vector machine with a linear or radial basis function kernel as shown for subject 4 (figure A.5 & A.6), where the support vector machines had a very low performance, since c and γ are tuned parameters. As for the support vector machine with a radial basis function kernel performing the best for specificity the parameters γ and c can be tuned accordingly per each subject, which improves the overall performance.

Spatial Independent Component Analysis is used to reduce temporal dimensionality to preserve the spatial dimensions, which contain valuable cortical activity. The paper [Huang et al. (2019)] used a spatial Independent Component Analysis on EEG movement artifact data in attempt to

classify different speeds 0.4, 0.8, 1.2, and 1.6 m/s. We replicated their method by using a spatial Independent Component Analysis, but on EEG data while a subject was walking at level ground at 0.5 m/s, 0.75 m/s, 1.0 m/s, 1.25 m/s, and a self-paced speed. We also used naive bayes, logistic regression, and a support vector machine with a linear kernel like [Huang et al. (2019)] used. Both of our classification performance values showed that logistic regression performed the best, but the classification of EEG movement artifact performed better then the classification of EEG. We believe the actual EEG performed less because it contains a mixture of artifact signals such as EKG, EMG, and ECoG. In contrast the movement artifact EEG that [Huang et al. (2019)] analyzed contains less non-movement artifact.

Future work will aim at developing and implementing an online algorithm to predict overground walking speed in real-time, which could be used to enhance brain-machine interface gait-oriented devices. This online algorithm will use a moving average function to predict and forecast the speed as the person is doing a dynamic task like walking. Also we can apply more pre-processing of the EEG data, apply more classification methods, and tune the seven classification methods we used to get a better overall performance.

APPENDIX A: SUPPLEMENTARY DATA

Subject 2					
Bagging					
Predicited/Actual Speeds					
	Speed 1	Speed 2	Speed 3	Speed 4	Speed 5
Speed 1	25	0	0	0	0
Speed 2	0	25	1	0	0
Speed 3	0	0	24	0	0
Speed 4	0	0	0	25	0
Speed 5	0	0	0	0	25
Boosting					
Speed 1	25	0	0	0	0
Speed 2	0	25	1	0	0
Speed 3	0	0	24	0	0
Speed 4	0	0	0	25	0
Speed 5	0	0	0	0	25
Random Forest					
Speed 1	25	0	0	0	0
Speed 2	0	25	0	0	0
Speed 3	0	0	24	0	0
Speed 4	0	0	1	25	0
Speed 5	0	0	0	0	25
Naïve Bayes					
Speed 1	25	0	2	0	0
Speed 2	0	25	1	0	0
Speed 3	0	0	22	1	0
Speed 4	0	0	0	24	0
Speed 5	0	0	0	0	25
Logistic Regression					
Speed 1	25	0	0	0	0
Speed 2	0	25	0	0	0
Speed 3	2	0	22	1	0
Speed 4	0	0	0	25	0
Speed 5	0	0	0	0	25
Linear SVM					
Speed 1	24	0	0	0	0
Speed 2	1	25	1	0	0
Speed 3	0	0	24	0	2
Speed 4	1	0	0	24	0
Speed 5	0	0	0	0	23
SVM RBF					
Speed 1	25	0	0	0	0
Speed 2	0	25	0	0	0
Speed 3	0	0	25	1	0
Speed 4	0	0	0	24	0
Speed 5	0	0	0	0	25

Color Scale
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25

Figure A.1: Confusion Tables for Subject 2.

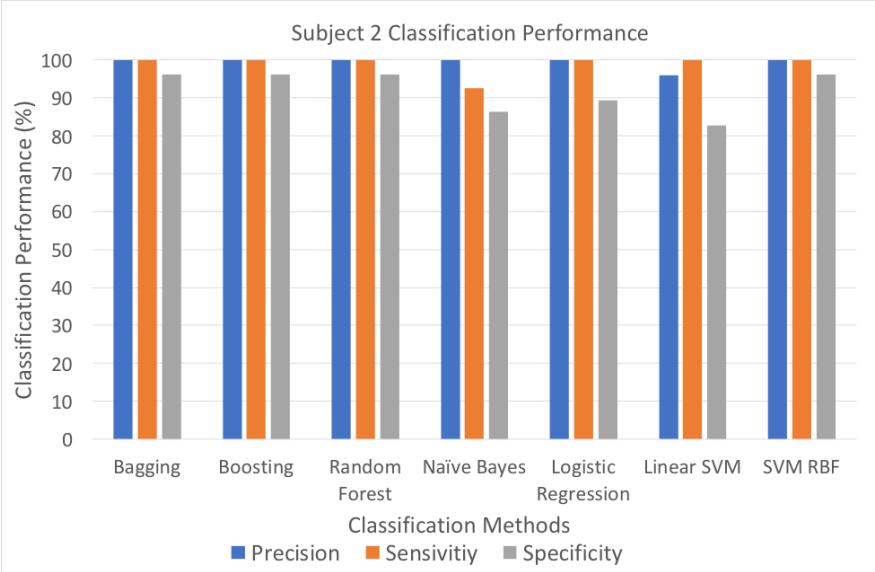

Figure A.2: Precision, Sensitivity, and Specificity Values for Subject 2.

Subject 3					
Bagging					
Predicited/Actual Speeds					
	Speed 1	Speed 2	Speed 3	Speed 4	Speed 5
Speed 1	25	1	4	0	0
Speed 2	0	24	0	0	0
Speed 3	0	0	21	0	1
Speed 4	0	0	0	25	0
Speed 5	0	0	0	0	24
Boosting					
Speed 1	25	0	1	0	0
Speed 2	0	25	0	0	0
Speed 3	0	0	24	0	1
Speed 4	0	0	0	25	0
Speed 5	0	0	0	0	24
Random Forest					
Speed 1	25	0	4	0	1
Speed 2	0	25	0	0	0
Speed 3	0	0	21	1	0
Speed 4	0	0	0	24	0
Speed 5	0	0	0	0	24
Naïve Bayes					
Speed 1	24	0	0	0	0
Speed 2	0	24	2	0	0
Speed 3	0	0	22	0	0
Speed 4	1	0	0	24	0
Speed 5	0	1	1	0	25
Logistic Regression					
Speed 1	18	0	1	1	5
Speed 2	0	14	4	5	2
Speed 3	0	0	24	1	0
Speed 4	0	0	0	25	0
Speed 5	0	0	1	2	22
Linear SVM					
Speed 1	25	0	0	0	2
Speed 2	0	25	8	2	2
Speed 3	0	0	17	0	0
Speed 4	0	0	0	23	0
Speed 5	0	0	0	0	21
SVM RBF					
Speed 1	25	0	0	0	0
Speed 2	0	25	2	1	0
Speed 3	0	0	23	0	0
Speed 4	0	0	0	24	0
Speed 5	0	0	0	0	25

Color Scale
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25

Figure A.3: Confusion Tables for Subject 3.

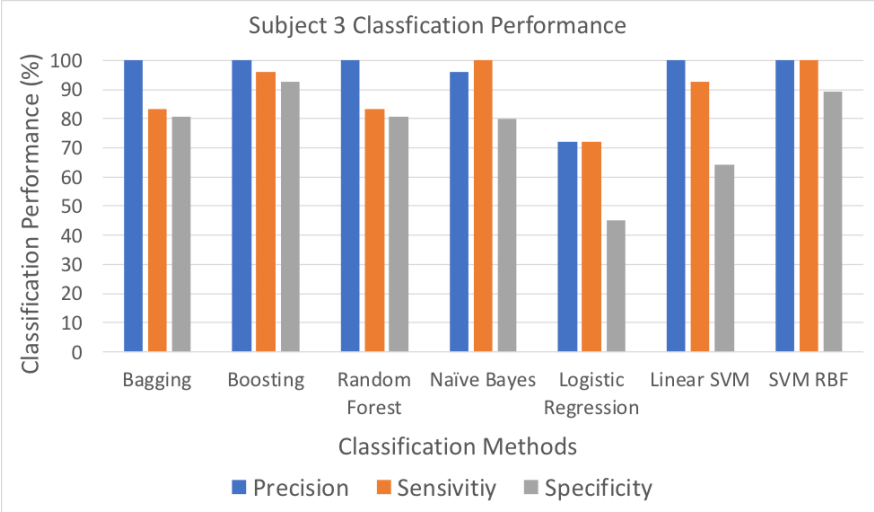

Figure A.4: Precision, Sensitivity, and Specificity Values for Subject 3.

Subject 4					
Bagging					
Predicited/Actual Speeds					
	Speed 1	Speed 2	Speed 3	Speed 4	Speed 5
Speed 1	0	0	0	0	3
Speed 2	23	20	14	10	18
Speed 3	0	0	4	1	0
Speed 4	2	5	7	14	3
Speed 5	0	0	0	0	1
Boosting					
Speed 1	0	0	0	0	1
Speed 2	13	18	8	7	16
Speed 3	12	6	15	10	7
Speed 4	0	1	0	7	0
Speed 5	0	0	2	1	1
Random Forest					
Speed 1	0	0	0	0	0
Speed 2	24	23	18	12	23
Speed 3	1	1	6	1	0
Speed 4	0	1	1	12	0
Speed 5	0	0	0	0	2
Naïve Bayes					
Speed 1	0	0	0	0	0
Speed 2	0	0	0	0	0
Speed 3	0	0	2	0	0
Speed 4	25	25	22	25	22
Speed 5	0	0	1	0	3
Logistic Regression					
Speed 1	23	2	0	0	0
Speed 2	0	25	0	0	0
Speed 3	2	13	8	0	2
Speed 4	8	10	0	5	2
Speed 5	0	6	0	0	19
Linear SVM					
Speed 1	0	0	0	0	0
Speed 2	0	1	0	0	0
Speed 3	25	24	25	25	25
Speed 4	0	0	0	0	0
Speed 5	0	0	0	0	0
SVM RBF					
Speed 1	9	0	0	0	0
Speed 2	0	12	0	1	0
Speed 3	0	0	8	0	0
Speed 4	15	13	14	18	12
Speed 5	1	0	3	6	13

Color Scale
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25

Figure A.5: Confusion Tables for Subject 4.

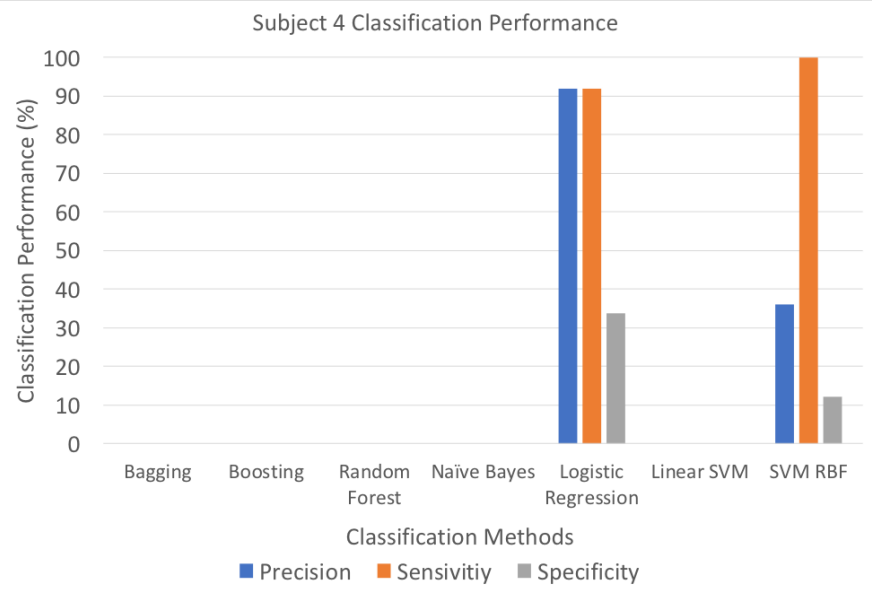

Figure A.6: Precision, Sensitivity, and Specificity Values for Subject 4.

Subject 5					
Bagging					
Predicted/Actual Speeds					
	Speed 1	Speed 2	Speed 3	Speed 4	Speed 5
Speed 1	6	0	0	0	0
Speed 2	0	5	0	0	0
Speed 3	18	18	25	6	10
Speed 4	0	1	0	19	3
Speed 5	1	1	0	0	12
Boosting					
Speed 1	6	0	0	0	1
Speed 2	0	3	0	0	1
Speed 3	8	10	10	0	10
Speed 4	11	12	15	25	9
Speed 5	0	0	0	0	5
Random Forest					
Speed 1	7	0	0	0	0
Speed 2	0	3	0	0	0
Speed 3	18	19	25	6	13
Speed 4	0	3	0	19	4
Speed 5	0	0	0	0	8
Naïve Bayes					
Speed 1	23	16	17	12	14
Speed 2	0	2	0	2	5
Speed 3	0	6	5	0	1
Speed 4	1	0	0	7	0
Speed 5	1	1	3	4	5
Logistic Regression					
Speed 1	22	0	1	2	0
Speed 2	11	10	1	2	1
Speed 3	4	0	21	0	0
Speed 4	1	1	0	23	0
Speed 5	3	0	8	3	11
Linear SVM					
Speed 1	0	0	0	0	0
Speed 2	25	25	25	24	25
Speed 3	0	0	0	0	0
Speed 4	0	0	0	1	0
Speed 5	0	0	0	0	0
SVM RBF					
Speed 1	1	0	0	0	2
Speed 2	24	24	24	23	20
Speed 3	0	0	1	0	0
Speed 4	0	1	0	2	0
Speed 5	0	0	0	0	3

Color Scale
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25

Figure A.7: Confusion Tables for Subject 5.

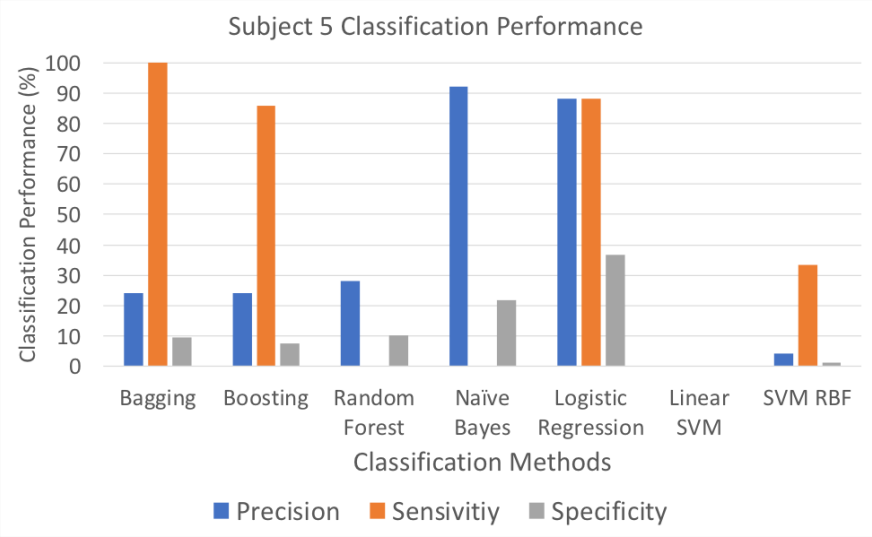

Figure A.8: Precision, Sensitivity, and Specificity Values for Subject 5.

Subject 6					
Bagging					
Predicted/Actual Speeds					
	Speed 1	Speed 2	Speed 3	Speed 4	Speed 5
Speed 1	25	0	2	0	5
Speed 2	0	25	0	0	0
Speed 3	0	0	23	0	0
Speed 4	0	0	0	25	1
Speed 5	0	0	0	0	19
Boosting					
Speed 1	25	0	0	0	2
Speed 2	0	24	1	0	0
Speed 3	0	1	24	0	2
Speed 4	0	0	0	25	1
Speed 5	0	0	0	0	20
Random Forest					
Speed 1	25	0	2	0	4
Speed 2	0	25	0	0	0
Speed 3	0	0	23	0	0
Speed 4	0	0	0	25	1
Speed 5	0	0	0	0	20
Naïve Bayes					
Speed 1	24	0	2	0	0
Speed 2	1	25	0	0	0
Speed 3	0	0	23	0	0
Speed 4	0	0	0	25	0
Speed 5	0	0	0	0	25
Logistic Regression					
Speed 1	23	0	1	1	0
Speed 2	1	19	3	1	1
Speed 3	0	0	25	0	0
Speed 4	0	0	1	24	0
Speed 5	0	0	0	1	24
Linear SVM					
Speed 1	22	0	0	0	0
Speed 2	0	24	0	0	0
Speed 3	0	0	22	0	0
Speed 4	0	0	0	25	0
Speed 5	3	1	3	0	25
SVM RBF					
Speed 1	23	0	0	0	0
Speed 2	2	25	0	2	0
Speed 3	0	0	23	0	0
Speed 4	0	0	0	23	0
Speed 5	0	0	2	0	25

Color Scale
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25

Figure A.9: Confusion Tables for Subject 6.

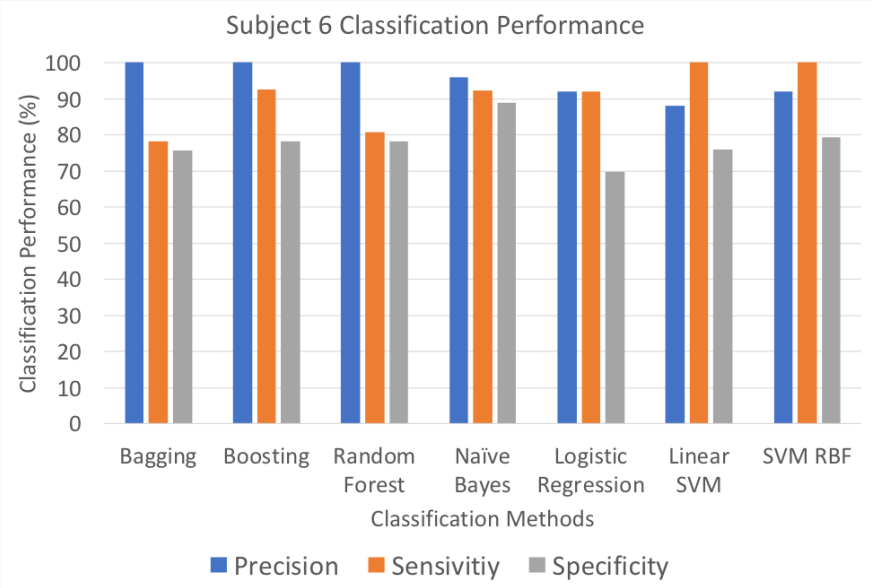

Figure A.10: Precision, Sensitivity, and Specificity Values for Subject 6.

Subject 7					
Bagging					
Predicted/Actual Speeds					
	Speed 1	Speed 2	Speed 3	Speed 4	Speed 5
Speed 1	14	0	0	0	0
Speed 2	0	24	0	0	0
Speed 3	0	0	21	0	0
Speed 4	11	1	4	25	1
Speed 5	0	0	0	0	24
Boosting					
Speed 1	13	0	0	0	0
Speed 2	0	21	0	0	0
Speed 3	5	0	18	0	0
Speed 4	7	4	7	25	1
Speed 5	0	0	0	0	24
Random Forest					
Speed 1	20	0	0	0	0
Speed 2	0	24	0	0	0
Speed 3	0	0	22	0	0
Speed 4	5	1	3	25	1
Speed 5	0	0	0	0	24
Naïve Bayes					
Speed 1	25	8	10	1	6
Speed 2	0	14	0	0	0
Speed 3	0	0	15	0	0
Speed 4	0	3	0	24	1
Speed 5	0	0	0	0	18
Logistic Regression					
Speed 1	23	0	0	2	0
Speed 2	2	23	0	0	0
Speed 3	0	0	20	5	0
Speed 4	0	0	0	25	0
Speed 5	1	0	0	0	24
Linear SVM					
Speed 1	9	3	0	0	0
Speed 2	0	15	0	0	0
Speed 3	16	7	25	3	12
Speed 4	0	0	0	22	0
Speed 5	0	0	0	0	13
SVM RBF					
Speed 1	25	5	12	1	0
Speed 2	0	20	0	0	0
Speed 3	0	0	13	0	0
Speed 4	0	0	0	24	0
Speed 5	0	0	0	0	25

Color Scale
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25

Figure A.11: Confusion Tables for Subject 7.

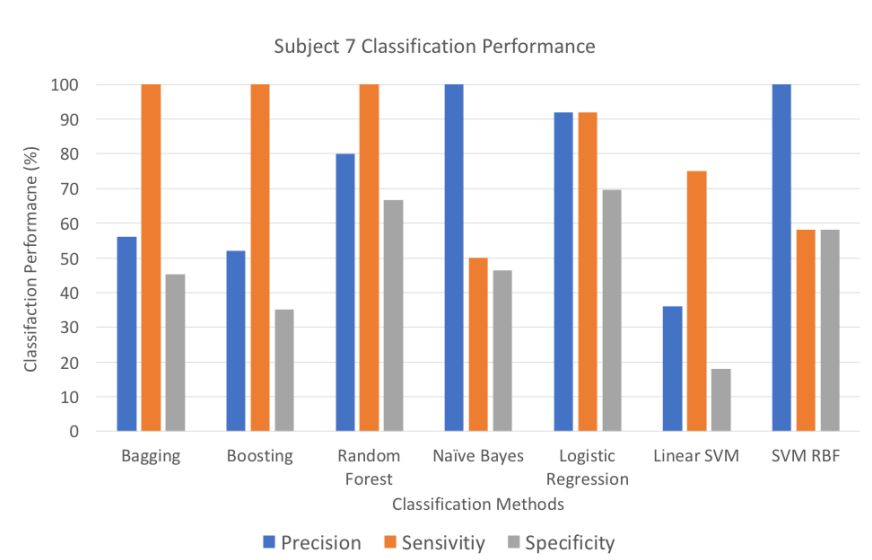

Figure A.12: Precision, Sensitivity, and Specificity Values for Subject 7.

APPENDIX B: SUPPLEMENTARY INFORMATION

REFERENCES

[1] A. D. Nordin, W. D. Hairston, and D. P. Ferris, "Dual-electrode motion artifact cancellation for mobile electroencephalography," J. Neural Eng., 2018.

[2] A. Liaw, M. Wiener, L. Breiman, and A. Cutler, "Package ' randomForest '. Breiman and Cutler's random forests for classification and regression," Package "randomForest." 2012.

[3] A. M. A. Handojoseno et al., "An EEG study of turning freeze in Parkinson's disease patients: The alteration of brain dynamic on the motor and visual cortex," in Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS, 2015.

[4] A. S. Oliveira, B. R. Schlink, W. D. Hairston, P. König, and D. P. Ferris, "Induction and separation of motion artifacts in EEG data using a mobile phantom head device," J. Neural Eng., 2016.

[5] A. Subasi and E. Erçelebi, "Classification of EEG signals using neural network and logistic regression," Comput. Methods Programs Biomed., 2005.

[6] Calhoun, V. D., Adali, T., Hansen, L. K., Larsen, J. J. Pekar, J. J. (2003).ICA of functional MRI Data: An Overview.In Proceedings of the 4th International Sympo-sium on Independent Component Analysis and Blind Signal Separation, 2003, 281288.

[7] Cortes, Esteban Alfaro, Matias Gamez Martinez, and Noelia Garcia Rubio. adabag: Applies Adaboost.M1 and Bagging. R package, 2018, version 4.2.

[8] D. Meyer et al., "Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TUWien," The Comprehensive R Archive Network. 2017.

[9] D. R. Edla, K. Mangalorekar, G. Dhavalikar, and S. Dodia, "Classification of EEG data for human mental state analysis using Random Forest Classifier," in Procedia Computer Science, 2018.

[10] F. Artoni, A. Delorme, and S. Makeig, "Applying dimension reduction to EEG data by Principal Component Analysis reduces the quality of its subsequent Independent Component decomposition," Neuroimage, 2018.

[11] Holland, S. (2008). Principal Components Analysis. strata.uga.edu.

[12] Huang HH, Condor A, and Huang H. (2019) Classification of EEG Motion Artifact Signals Using Spatial ICA. Statistical modeling for biomedical research: contemporary topics in the field, Springer, 2019. Accepted.

[13] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," Mach. Learn., 2002.

[14] J. Gareth, W. Daniela, H. Trevor, and T. Rober, An Introduction to Statistical Learning with Applications in R. 2000.

[15] J. Machado, A. Balbinot, and A. Schuck, "A study of the Naive Bayes classifier for analyzing imaginary movement EEG signals using the Periodogram as spectral estimator," in ISSNIP Biosignals and Biorobotics Conference, BRC, 2013.

[16] J. T. Gwin, K. Gramann, S. Makeig, and D. P. Ferris, "Removal of Movement Artifact From High-Density EEG Recorded During Walking and Running," J. Neurophysiol., 2010.

[17] Karhunen, J., Hyvarinen, A. and Oja, E. (2001). Independent Component Analysis. New York: John Wiley & Sons, Inc.

[18] K. L. Snyder, J. E. Kline, H. J. Huang, and D. P. Ferris, "Independent Component Analysis of Gait-Related Movement Artifact Recorded using EEG Electrodes during Treadmill Walking," Front. Hum. Neurosci., 2015.

[19] Majka, M. (2017) naivebayes: High Performance Implementation of the Naive Bayes Algorithm. R package, 2017, version 0.9.1.

[20] N. E. Helwig and <helwig@umn Edu> Maintainer, "Package 'eegkit' Title Toolkit for Electroencephalography Data," R Packag. version, 2015.

[21] N. Kasabov and S. Pang, "Transductive support vector machines and applications in bioinformatics for promoter recognition," in Proceedings of 2003 International Conference on Neural Networks and Signal Processing, ICNNSP'03, 2003.

[22] Ripley, B. (2018) tree: Classification and Regression Trees. R package, 2018, version 1.0.39

[23] S. Sun, "Ensemble Learning Methods for Classifying EEG Signals," in Multiple Classifier Systems, 2007.

[24] S. Tong and D. Koller, "Support Vector Machine Active Learning with Applications to Text Classification Simon," CrossRef List. 2000.

[25] V. V. Cozac, U. Gschwandtner, F. Hatz, M. Hardmeier, S. Rüegg, and P. Fuhr, "Quantitative EEG and Cognitive Decline in Parkinson's Disease," Parkinsons. Dis., 2016.

[26] W. N. Venables and B. D. Ripley, "nnet: Feed-forward Neural Networks and Multinomial Log-Linear Models," R Packag., 2013.

[27] Y. P. Lin, C. H. Wang, T. L. Wu, S. K. Jeng, and J. H. Chen, "Support vector machine for EEG signal classification during listening to emotional music," in Proceedings of the 2008 IEEE 10th Workshop on Multimedia Signal Processing, MMSP 2008, 2008.