5-2018

Computational Analysis of Upper Extremity Movements for People Post-Stroke

Zachariah Edward Nelson
University of Tennessee, znelson1@vols.utk.edu

To the Graduate Council:

I am submitting herewith a thesis written by Zachariah Edward Nelson entitled "Computational Analysis of Upper Extremity Movements for People Post-Stroke." I have examined the final electronic copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Mechanical Engineering.

Eric R. Wade, Major Professor

We have read this thesis and recommend its acceptance:

Jay I. Frankel, William R. Hamel, Jeffrey A. Reinbolt, Daniel Caleb Rucker, Jindong Tan

Accepted for the Council:
Dixie L. Thompson

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Computational Analysis of Upper Extremity Movements for People Post–Stroke

A Thesis Presented for the

Master of Science

Degree

The University of Tennessee, Knoxville

Zachariah Edward Nelson

May 2018

To my parents, Todd and Juli Nelson, and my grandparents, Jerold and Barbara Nelson. None of this would have been possible without their love, encouragement, and support. To my great-uncle, Richard Nelson, for the motivation to pursue research in medical technology.

Acknowledgments

I would like to thank Dr. Eric Espinoza–Wade (Mechanical, Aerospace, and Biomedical Engineering Department, MABE) of the University of Tennessee for the guidance to become a more dedicated researcher and pursue a thesis, Dr. Bruce Maclennan (Electrical Engineering and Computer Science) of the University of Tennessee for assistance in utilizing machine learning techniques, Drs. Frankel, Hamel, Rheinbolt, Rucker, and Tan (MABE) for being a part of my thesis committee, and Dr. Richard Gregory of Lipscomb University for setting me down the path to pursue graduate research. I would also like to acknowledge Elissa Phillips, PT, DPT, and Amanda Waynick, OT, for assistance with data collection, and Sarvenaz Chaeibakhsh.

Abstract

Wearable sensors have been beneficial in assessing motor impairment after stroke. Individuals who have experienced stroke may benefit from the use of wearable sensors to quantify and assess quality of motions in unobserved environments. Seven individuals participated in a study wherein they performed various gestures from the Fugl–Meyer Assessment (FMA), a measure of post–stroke impairment. Participants performed these gestures while being monitored by wearable sensors placed on each wrist. A series of MATLAB functions were written to process recorded sensor data, extract meaningful features from the data, and prepare those features for further use with various machine learning techniques. A combination of linear and nonlinear regression was applied to frequency domain values from each gesture to determine which can more accurately predict the time spent performing the gesture, and the associated gesture FMA score. General performance suggests that linear regression techniques appear to better fit paretic gestures, while nonlinear regression techniques appear to better fit non–paretic gestures. A use of classifier techniques were used to determine if a classifier can distinguish between paretic and non–paretic gestures. The combinations include determining if a higher performance is obtained through the use of either accelerometer, rate gyroscope, or both modalities combined. Our findings indicate that, for upper–extremity motion, classifiers trained using a combination of accelerometer and rate gyroscope data performed the best (accuracy of 73.1%). Classifiers trained using accelerometer data alone and rate gyroscope data alone performed slightly worse than the combined data classifier (70.2% and 65.7%, respectively). These results suggest specific features and methods suitable for the quantification of impairment after stroke.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Hemiparetic stroke is a form of stroke where an individual experiences more coordination issues along one side than the other. Many individuals who experience stroke may undergo physical or occupational therapy to regain motor coordination to a degree that they may have possessed prior to stroke. Wearable sensors may allow for improvements in feedback of therapy sessions as they can record motion data from individuals in a natural and unobserved environment. Machine learning, a method of utilizing programs to predict outcomes and interpret information based on previous experiences, may be used for various applications with motion data post stroke, such as predicting features or classifying gestures.

A larger percentage of research in wearable sensors has been focused on lower extremities than upper. While evaluating the motion data of upper extremities is more complex than lower extremities, as the former has a larger range of motion than the latter, assessing upper extremity gestures is a relevant goal for anyone recovering from stroke.

The research conducted in this study sought out to determine the nature that machine learning can be applied to motion data post stroke. One avenue of research focused on how the total time of gestures can be measured based on results of frequency domain values. Another avenue of research focused on classifying gestures as paretic or non-paretic based on features of various modalities with the use of the K-Nearest Neighbors (KNN) classifier. It is hypothesized that the motion data recorded from accelerometers are more beneficial towards recognizing paretic and non-paretic gestures than data recorded from rate gyroscopes.

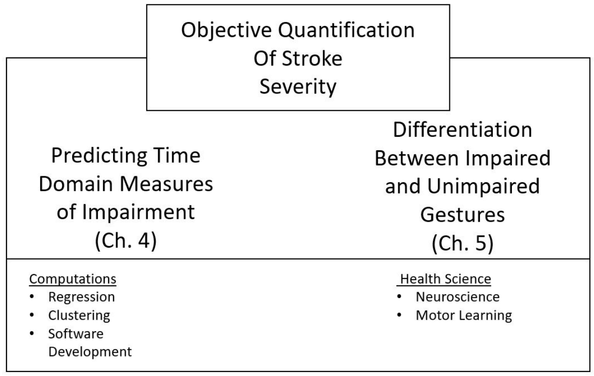

Figure 1.1: Focus areas and conceptual framework of the proposed research.

Figure 1.1 illustrates several topics covered in this thesis. A primary goal was research into the objective quantification of stroke severity with the use of wearable sensors. This research can potentially impact the fields of computations and health science. A modest amount of software was developed to organize and implement motion data for analysis via machine learning methods like regression and cluster classification.

Chapter 2 discusses prior research that has been conducted regarding topics such as hemiparetic stroke and their relevance to the thesis. Wearable sensors are discussed regarding the importance of monitoring motion data in natural environments. Inertial sensing and stroke is cited for relevance in functional assessments to assess effects of stroke in a quantifiable value, the relevance of sensor modalities used to collect information, the type of features measured from inertial data, the importance of gesture recognition, and the valid use of machine learning.

Chapter 3 goes into detail about the software developed in MATLAB to interpret data collected from wearable sensors for further implementation. A variety of software programs and functions were written to efficiently automate many steps of the process, such as interpreting raw data files and filtering to remove noise. Many programs included features to allow for user-specified choices and importing large quantities of data into a special file format to reduce processing time.

Chapter 4 covers research into predicting time domain measures of impairment. Data from chapter three is utilized for a variety of linear and nonlinear regression models to determine if a continuous model may be designed for a correlation between frequency domain values and total time spent performing a gesture. A further analysis is conducted to determine if the same concept can be applied to predict the a quantifiable score through the use of frequency domain values.

Chapter 5 discusses research into differentiating between impaired and unimpaired gestures. Data from chapter three is assessed by means of KNN classification techniques to determine if a specific form of sensor modalities and data pre-processing can improve classification performance. Several models were compared to determine the optimal conditions.

Chapter 6 discusses several of the relevant findings taken from the research conducted in chapters four and five. Results from both chapters are elaborated into credible findings that may hold relevance to health sciences and stroke rehabilitation.

Chapter 7 concludes by summarizing the important factors related to the thesis. A concise elaboration is presented regarding the importance of the research conducted, why it was conducted, and how it may further ongoing medical research.

Chapter 2

Background

2.1 Overview and Problem Motivation

Stroke is currently ranked the fifth largest cause of death behind heart disease and cancer. In 2017, someone experienced a stroke, on average, every 40 seconds, while someone died of a stroke every four minutes [1]. Individuals affected by stroke will experience loss of motor coordination and require extensive physical, occupational, and speech therapy to regain significant motor function.

Stroke results in the loss of coordination and other difficulties due to poor communication between an individual's brain and muscles. This causes weakened muscles and reduces synergistic pattern capability activities where multiple muscles groups are active in a cooperative pattern. A lack of muscle synergy in an individual post stroke may alternate between utilizing several muscles incrementally compared to a more natural simultaneous motion. Hemiparetic stroke is a form of stroke where an individual experienced a loss of motor coordination on one side of the body. This weaker side of the body is referred to as the 'paretic' side while the alternate side is referred to as non-paretic.

Individuals with hemiparetic stroke go on to live with numerous deficits. One such deficit is known as learned nonuse. This phenomenon represents voluntary nonuse of the limb (beyond the limitations resulting from the injury) [2]. It is thought that learned nonuse generally evolves in the home setting. Specifically, when an individual attempts to use the limb, they receive negative feedback in the form of slow movement, pain, or task failure. The individual chooses to use another limb, and thus reduces practice with the paretic limb. This process is known as the 'vicious cycle' and leads to numerous detrimental comorbidities [3, 4]. It is thought that this process evolves primarily in the home setting, outside of the purview of the health care professional. Thus, it is critically important to understand how individuals behave outside of the clinical setting to understand the evolution of disease, and to fully understand the effects of therapy.

2.2 Wearable Sensors for Monitoring Activity

Approaches to home monitoring have evolved with advances in technology. Generally, such monitoring has limited validity when an individual is more cognizant of being monitored. In early ecological validity studies, human monitoring was performed by other humans; researchers followed participants and recorded their physical activity with pen and paper [5, 6]. While this provided accurate information, an individual is less likely to act naturally when they know they are being observed. This implies that the documented activities are accurate but not valid because the person performing the gestures is aware they are being observed and recorded. Self-report logs allow people to practice and rate their own performances and return the results to a physical therapist. However, self report data are known to suffer from recall bias [7, 8]. Wearable sensors will be able to provide raw information recorded directly from the people wearing them, with little bias. Figures 2.1 and 2.2 illustrate wearable sensors used in motion analysis.

Figure 2.1: APDM Wearable Sensor for measuring linear acceleration and angular velocity [9][10]

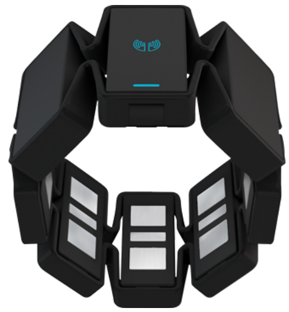

Figure 2.2: Myo Armband Wearable Sensor for measuring motion with accelerometer, rate gyroscope, and EMG sensors. [11]

Through monitoring individuals post-stroke in their everyday environment with wearable sensors, their physical or occupational therapist can interpolate relevant information about their physical activity and use that information to better assess what is the best course of action for optimal recovery. It is important to monitor the physical activities of post-stroke individuals in a home environment as the majority of their time will not be spent with a therapist who can properly evaluate the accuracy of their actions. An individual may not always be honest about the frequency that they are performing certain gestures or may be performing a gesture incorrectly.

Improvements in efficacy, cost, and size of sensors have allowed for the development of non-invasive devices suitable for external environments (outside of a research lab or medical facility). While some individuals may consider wearable sensors to be intrusive to their privacy, many individuals with disabilities applaud the use of rehabilitation technology in the context of their home and community at a low cost [12]. The presence of wearable sensors is generally forgotten after an initial adjustment period, assuming that the sensors are positioned on rigid body parts, relatively small, and produce little to no heat while active [13]. This suggests that the use of wearable sensors to assess physical activity of individuals affected by hemiparetic stroke may be viable as many people are comfortable with digital watches that incorporate various sensors to record and catalog physical activities, as well as being inconspicuous enough that the individuals wearing the sensors will not notice over time.

There are multiple versions of wearable sensors and technology designed for various applications. Several fitness devices already exist with the intention of being worn by an individual to measure activity, sometimes using a cell phone connected via Bluetooth to process and analyze the information [14, 15]. Many commercial devices are capable of measuring activity in the form of distance traveled or amount of calories burned, possibly through measuring the swing of an arm.

2.3 Inertial Sensing and Stroke

Inertial sensors have emerged as useful tools for the management of stroke by recognizing gestures and assessing quality of motion through machine learning techniques. The machine learning techniques can be applied through the use of information derived from functional assessments to quantify gesture performance and feature measurement to utilize meaningful outcomes.

2.3.1 Functional Assessments

Individuals undergoing physical therapy may be asked to perform a series of exercises to better gauge the extent that individual has been affected by stroke. These exercises are generally conducted by an occupational or physical therapist and measured by a quantifiable metric.

Figure 2.3: Participant performing Fugl–Meyer Assessment under the guidance of a trained occupational therapist.

The Fugl-Meyer Assessment (FMA) is an objective impairment index commonly used for individuals undergoing stroke rehabilitation [16, 17, 18]. The FMA contains several gestures that are performed using the upper and lower-extremities of individuals and are scored based on visual analysis by a trained physical therapist (Figure 2.3). Any gesture conducted in the FMA by an individual is scored with a three-point ordinal scale based on the degree of accuracy when performing gestures for a total score of 66. The gestures used in the FMA focus primarily on range of motion and muscular synergy, starting from more simple gestures focusing on the upper arm to multi-joint gestures to testing the efficacy of distal grip. This will suggest that any data we collect from individuals performing gestures from the FMA will be relevant to our research as they will have an emphasis on the range of motion performed by the individual as well as coordination between portions of their upper extremities.

The Wolf Motor Function Test (WMFT) is a performance based assessment of upper extremity functional capabilities for individuals post-stroke. The WMFT uses a six-point ordinal rating scale for a total score of 75. Gestures performed in the WMFT are usually arranged from lower to higher complexity, beginning with forearm exercises and ending with finger and coordination exercises [19, 20, 21]. Much of the WMFT is very similar in nature to the FMA, given that they both have individuals post-stroke perform similar gestures.

Wearable sensors, such as accelerometers, can be used with the FMA or WMFT to interpret meaningful data. Data collected and analyzed from wearable sensors can be accurate to the extent that FMA or WMFT scores can be predicted with relative accuracy through the use of machine learning techniques [22, 23, 24, 25].

2.3.2 Sensors for Measuring Symptoms of Stroke

Several factors of an individual's daily life are affected after experiencing a stroke. A loss of coordination can cause symptoms such as impairment with posture, gait, limb use, and other general activities.

The use of wearable sensors may also determine when an individual is performing a gesture incorrectly. Some individuals may compensate for their paretic extremity by utilizing their trunk or becoming more dependent on the non-paretic extremity [26, 27]. Such activities can be seen as detrimental or counterintuitive to the intentions of physical therapy.

Accelerometers can be an efficient method for measuring stroke along upper extremities as part of the interest is the use of measuring degrees of movement in an individual [28, 29, 30].

Accelerometers are also sensitive enough to recognize the effect of gravity, and can be applied to evaluating the orientation of an individual.

Wearable sensors can be utilized in measuring the acceleration and velocity of an individual's gait and providing feedback to improve the effect of therapy [31, 32, 33]. The sensors can also be implemented to detect how often an individual is utilizing either upper extremity, ensuring that an individual is less likely to be developing 'learned non-use' [34, 35]. This can also be applied to recognize various activities of daily living (ADL), gestures that are most commonly performed in an individual's natural environment.

2.3.3 Features Extracted

Assessing and measuring features from motion data allows for large quantities of data to be represented in smaller and more discrete values. This method can allow for professional therapists who may not find the same meaningful information from raw Inertial Measurement Unit (IMU) data to better understand and assess the quality of motion of an individual.

Analysis of motion data in the form of quantifiable metrics allows for the reduction in computational requirements when measuring and assessing gestures with various techniques, and also prevents model over-fitting. This may also compensate for data that may be reflected due to being performed on opposite extremities and measured along different axes.

Previous studies have investigated the meaningfulness of several feature metrics based on accelerometer data collected from upper extremities. Tested features include mean, median, variance, standard deviation, signal magnitude area, root mean square, mean-squared jerk cost, power ratios, jerk, and total peaks. Many of these values are time domain metrics, which represent data in terms of the dimensions of amplitude and time.

Individuals post-stroke can experience issues with coordination and smoothness of motion when performing gestures, resulting in sudden jerked motions. Jerk is measured as the rate of change of acceleration with respect to time. It is worth measuring jerk as individuals post-stroke will register higher jerk values when performing a gesture when compared to someone who has not experienced a stroke [36]. Mean Square Jerk Cost is utilized a it can measure smoothness while remaining independent of movement duration [37, 38, 39].

The mean value and signal magnitude area (SMA) are used to measure the average value of activity and total magnitude of activity over a period of time, respectively. These values were utilized to represent an 'amount' of motion. The root mean square (RMS) measures the square root of the mean of the squares of a series of numbers and can represent the gross muscle activation in a gesture [40, 41, 42]. While the mean value represents the average, the median value is used as it can represent typical activity over a period of time [43, 44, 40, 42].

Standard deviation is a metric to measure the extent of deviation of a series of data from the mean value. This also carries significance as the mean value is a component of the feature and has been used for gesture recognition [42]. The total peaks is a measurement of relative maximum peaks detected in the motion data, and has been in use prior for activity recognition [42].

The power ratio consists as a ratio of a particular power spectrum across a complete spectrum that was measured prior. The selected spectrum values were between 0.1 to 1 Hz, 1 to 2 Hz, and 2 to 10 Hz. The power ratio can determine a difference in activity between paretic and non-paretic actions, as the paretic extremity will generally measure more of its activity in the 2 to 10 Hz range than non-paretic [37].

Time domain features can be relevant for gesture recognition, as their is often a difference between the time required to perform a gesture or activity between a paretic or non-paretic extremity. They can simplify multidimensional raw data into singular values, similar to the FMA or WMFT, resulting in easier use for physical therapists and non-engineers to extract meaning from the data. These quantitative values are also more useful for machine learning techniques.

2.3.4 Gesture Recognition

An important application of wearable sensors is the ability to recognize specific gestures performed in the natural environment. Recognizing ADL can allow for sensors to accurately interpret recorded data and to evaluate performance. Based on the position of wearable sensors on the body, several gestures can be classified with high accuracy. Sensors on the waist and ankle have been shown to be sensitive to gestures related to running, walking, sitting, and various other types of gait [45]. While these results are more focused on gait and lower extremity, the concept suggests similar application for upper extremities.

2.3.5 Machine Learning

Machine learning is a broad suite of computational tools allowing computers to analyze large quantities of data and is often used for separating and/or classifying data based on probabilistic inference models [46]. If properly trained, a machine learning algorithm can be applied to interpret and recognize large quantities of human data to assess relevant information, such as the frequency a gesture is being performed and their accuracy.

While highly efficient, machine learning usually requires large quantities of data in order to generate a relatively accurate and unbiased result as more data and larger sampling size suggests that more average data will appear and outliers will have less of an influence on any models created. Some versions of machine learning focus on binary classification methods, which involve filtering data with two possible outcomes, while others are designed to classify data with multiple outcomes based on proximity of other data samples [46]. Several machine learning models are separating the collected data into two sets: a training and testing set. Training sets are predominantly larger than the testing set and are used to generate a model based on the utilized input values of the set and the respective output classifying values. The testing set is used to assess the accuracy of the model built by the training set by running the the testing set through the model and comparing the output classifications assumed by the model against the output values previously recorded.

Linear regression is a fairly simple form of machine learning that is mainly used to determine if a correlation exists between various statistical values. A linear regression model is typically assessed by minimizing the total error, summation of distances between the data points and the regression model. Linear regression is often an ideal place to start when utilizing machine learning techniques by determining if a more simple technique can be implemented before utilizing more complex models.

Bagging Forest and Decision Trees can be very ideal as they can classify data as well as explain what choices were made in the model. Decision Trees are a form of machine learning where data begins in a single node and branches are formed to separate data based on the measured value of one or more features if the following nodes have filtered one or more sets of data that are classified. While models generated by a decision tree can explain how the data is being filtered, it can take a large amount of processing time to generate a model.

K-nearest neighbors (KNN) is a form of machine learning that assesses data based on proximity to local data values. Based on a choice of nearby data points or 'neighbors', KNN will classify documents based on their proximity to data points classified in the model created from the testing set. If a data point is measuring proximity to various classifications, the model will decide a classification based on the majority of neighbors closest to the unclassified sample.

The use of machine learning is relevant as it can be utilized to approximate gestures or assess the quality of gestures by interpreting values calculated from the motion data. With large samples of gestures performed, machine learning methods like regression or KNN classification can be applied to extracted features and potentially return meaningful results from motion data that can supplement and improve the quality of therapy for individuals post-stroke.

Wearable sensors demonstrate a capacity to operate in external environments with minimal impact to the lives the people wearing them. With the improvements in sensor cost and measurement accuracy, sensors are more viable in recording motion data from individuals post-stroke for the use of machine learning applications. Through the implementation of machine learning in human motion recorded from wearable sensors, relevant data may be interpreted in a fashion that can benefit physical or occupational therapists by providing feedback regarding the quality of everyday motion in patients.

Chapter 3

Software Development and Data Processing

3.1 Introduction

To maximize the efficiency of data analyses and to minimize the opportunity for errors, a software framework was created using primarily MATLAB software. This framework consisted of custom and existing MATLAB functions designed to read raw sensor data from study participants, filter those data, extract meaningful features, and provide an indexing tool capable of querying specific data corresponding to: participants; paretic/non–paretic limb; gesture; and sensor modality. The details of this system are provided in the following.

The intention of this section is to outline and explain the steps necessary for the use of the MATLAB code used to read the data from the H5 files stored in the APDM IMU sensors and interpret values that are relevant for further machine learning techniques. The APDM IMU sensors store all collected data into a Hierarchal Data Format version 5 (HDF5/H5) file. In order to properly analyze specific gestures and data, special programs were written in order to achieve these goals. A combination of built-in and personalized functions were implemented for the various steps in the process to organize all the relevant data. While both the Regression and clustering studies incorporate their own unique MATLAB scripts,

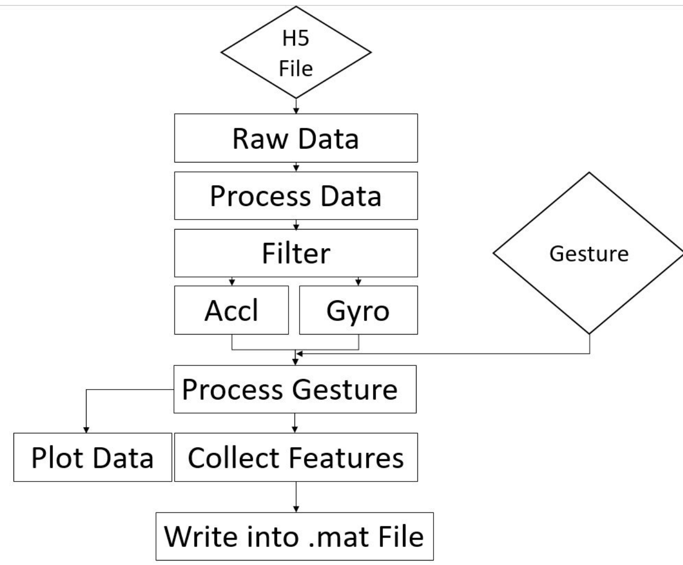

Figure 3.1: Diagram of code structure for MATLAB code Setup.m

both models utilize this script for the purpose of storing relevant values into a more efficient and manageable format.

Several custom functions were previously designed by Sarvenaz Chaeibakhsh, a research assistant. Further refinements were made to correct minor errors and bugs in the code.

3.2 Background

MATLAB is a programming language heavily derived from C++ and Python that has a strong focus on matrix-based arithmetic [47]. MATLAB is also capable of utilizing a multitude of libraries and packages, such as machine learning methods. For the majority of procedures performed in MATLAB, we have used the *Statistics and Machine Learning Toolbox* is used as it includes a variety of machine learning programs such as linear regression, radial basis function, k-means clustering, and k-nearest neighbors clustering techniques.

3.3 Methods

3.3.1 Main Function

All functions were written to be performed by a MATLAB script designated as `Setup.m` to document various measures and perform necessary debugging techniques. Overall, the purpose of `Setup.m` is to utilize a user-interface to select a participant, analyze a user-selected gesture, extract several features into a special file for further machine learning models, and produce relevant images. A diagram of the general structure of the MATLAB script is presented in Figure 3.1.

3.3.2 Reading Input Files

The initial process is to begin importing the raw H5 data files from the APDM sensors into a computer. The APDM sensors have a special docking setup that extract the H5 files and label them with a time-stamp. A MATLAB function titled `PatientCall.m` was designed to generate a user-interface that can allow an individual to select which participant data will be analyzed. Depending on the choice selected by the user, the H5 data related to an individual will be uploaded to the MATLAB program's local memory for further use.

The H5 file selected by the user will then be loaded by a custom MATLAB function titled `H5Reader.m`, which will take the raw H5 file and generate an output of the various data formats measured. While the function generates an output of accelerometer, rate gyroscope, magnetometer, quaternion orientation, sampling rate, and total time data, only the accelerometer, rate gyroscope, sampling rate, and total time data were utilized in all studies, as several sensors had corrupted magnetometer and quaternion orientation data. `H5Reader.m` measured and processed all H5 files available from all five APDM IMU sensors.

3.3.3 Visual Analysis/Documentation

Having provided informed consent, participants were filmed while performing the FMA under administration of a physical therapist. The recorded video footage was used to catalog gestures and record any data for further use.

At the beginning of each individual's video, a therapist will either shake all five APDM IMU sensors, or will tap an IMU located on the wrist three times. The intention behind this was to synchronize IMU and video data using an excel spreadsheet. The spreadsheet contained measurements of when each gesture from the FMA begins and ends for both paretic and non-paretic upper extremities, and was titled for each individual who participated in the study. The standard time and approximate video frame are documented to allow the data related to each gesture to be compartmentalized for further assessment. Up to five attempts by an individual to perform a gesture were documented into the Excel Spreadsheets. If a participant did not perform one of the FMA gestures, then the values will be replaced with "NaN", which stands for "Not A Number" and is interpreted by MATLAB as a non-existent number. This is more efficient than "0" or many other null values as MATLAB has several logic functions that operate depending on if a NaN value is detected.

The data also contained an offset value that will be recorded in the Excel spreadsheet read from `PatientCall.m` to offset delays between video and IMU data. This was usually found by analyzing raw acceleration data from `H5Reader.m`, looking for the relative peaks of data that coincide with the action performed by the physical therapist, and confirming by determining if the acceleration profiles of various gestures appeared accurate with the assumed offset value.

3.3.4 Gesture Cataloging

The data extracted from H5 files and documented with Excel spreadsheets will be utilized in another custom MATLAB function, `NewSampleRead.m`, to isolate data relevant to each gesture with each upper extremity.

`NewSampleRead.m` reads the documented Excel spreadsheet and records the documented frame data. The relevant data were recorded in two structures labeled as 'Left' or 'Right' while all relevant FMA gestures are listed as the branches for both structures. The function is only required to run once as the recorded frames apply to all gestures performed by an individual's paretic and non-paretic upper extremities.

Once the data has been filtered, a custom MATLAB function, `GestureSaver30.m`, was used to compartmentalize the modality results into subsections based on documented video frames. The function requires the 'Left' and 'Right' structures generated by `NewSampleRead.m`, the filtered data modalities, and a pair of values that designate the sensors which represent the left and right extremity. The function will take the beginning and ending frames documented for each gesture, divide by the framerate to more accurately correlate to the sensor data, and repeat the process to document the acceleration, rate gyroscope, and total time required to perform the gestures for each gesture performed by an individual.

The output is a structure documenting relevant data for one iteration of each gesture by an individual. This process is repeated five times to process the maximum five documented iterations of each gesture performed by an individual. This procedure is required to run five separate times due to complications in creating structures of structures. The function will also generate NaN values if the same value is read from the Excel spreadsheet.

3.3.5 Data Filtering

To remove noise from the sensor data, the accelerometer and rate gyroscope data were filtered using the written MATLAB functions `filtacc.m` and `filtgyro.m`, respectively. Both functions utilize another written MATLAB function, `filtmake.m`, which allows for use of either lowpass, highpass, and bandpass Butterworth filters and controlling order and cutoff frequencies. Both `filtac.m` and `filtgyro.m` use a fourth order Butterworth band pass filter with cutoff frequencies of 0.1 and 10Hz. These cutoff frequencies were chosen as anything below 0.1Hz can be considered noise from gravity, and anything above 10Hz can be considered noise as a majority of human motion does not exceed 10Hz.

The intention of these functions are to filter noise from the acceleration and rate gyroscope data that can be caused by jostling sensors or external interference while not removing data that can be considered relevant to any experiments where the tremors experienced by an individual's paretic upper extremity can be visually observed and interpreted from profile data. These functions filter data automatically for all three axis of a sensor modality simultaneously, but only operate one sensor at a time, so a built-in looping function is necessary to process all the modalities from all the sensors.

3.3.6 Feature Extraction

After the gestures are cataloged and filtered, features considered relevant are measured from the isolated portions of data corresponding to a specified gesture. A custom MATLAB function, `gesturetests.m`, used the axis data (accelerometer or rate gyroscope) and total time data to generate multiple features. The output of `gesturetests.m` are the mean, median, standard deviation, total number of peaks, power ratio from 0.1-1Hz, power ratio from 1-2Hz, power ratio from 2-10Hz, jerk, mean square jerk, average root mean square, root mean square, initial peak value, signal mass average, and total time of the gesture.

This function will generate a feature for all three axes measured by a specific sensor mode. The procedure is required to run five times to process all five gesture structures generated previously by `GestureSaver30.m`.

3.3.7 Data Plotting

To assist with debugging, a custom MATLAB function, `Testplot.m`, was used to generate plots of a user-selected gesture that contain both the acceleration and rate gyroscope profile for both upper extremities from a gesture. The function also differentiates between paretic and non-paretic gestures. This function was designed with the intention of comparing the differences between paretic and non-paretic versions of various gestures and further confirming the validity of previous offset values.

3.3.8 Data Compiling/Storage

To allow for data analysis, a MATLAB function was written to save extracted features. The MATLAB function, `DataCompiler.m`, will take the output data from `GestureSaver30.m`, run `gesturetests.m`, and store the resulting features in a .mat file for future use.

The intention of storing the extracted features into a .mat format is to reduce processing time required to replicate machine learning models by keeping all features saved. The .mat format was considered more ideal than writing results into a Microsoft Excel spreadsheet as less data was required and minimized the risk of overwriting data. This procedure was performed for each individual gesture for each participant. A visual representation of the .mat file is presented in Figure 3.2.

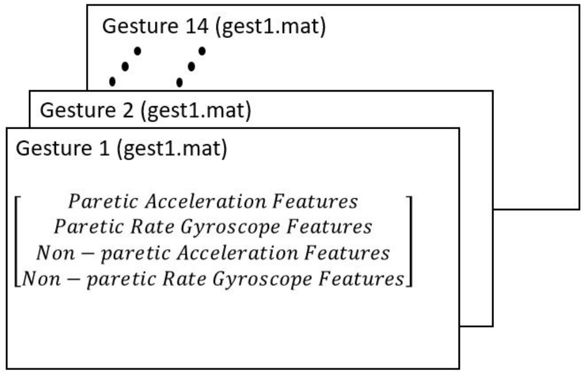

Figure 3.2: Conceptual representation of layered feature data storage in MATLAB .mat files.

Chapter 4

Predicting Measures of Impairment

4.1 Introduction

In continuous, in–home data sets, one outstanding challenge is how to segment gestures [48, 49]. In a lab setting with video data, relating sensor data to task performance is a fairly trivial task. However, in continuous data, this process has not been solved. Segmentation techniques (for instance, zero–crossings) have been utilized but these do not necessarily determine gesture start and stop times. The goal of the current approach is to adapt a technique typically used in speech and audio processing known as a moving window. Rather than segmenting gestures, a moving window consists of a fixed number of data points that 'slides' along the collected data [50]. Features from each window are calculated, and a variety of modeling techniques can then be used to match the windowed data to a known template. The current approach is designed to determine if such features, extracted from sliding windows, are predictive of segmented, time–domain features sensitive to impairment.

4.2 Background

The motivation of this assessment was to take values of frequency domain and determine if it may be capable of predicting time. One form of processing and recognizing gestures is gesture segmentation, which utilizes segmentation techniques to recognize gestures. Data segmentation often involves the use of separating large streams of data into smaller portions of data, either with or without any overlap in the segmented portions of data. [42, 51]. This process can be difficult to process and recognize gestures, due to variations in gesture time. This process is based off of continuous analysis, utilizing the frequency domain values measured at any instance to predict the total time. [52].

4.3 Methods

When implementing machine learning techniques, it is always worthwhile to begin with more simplistic models. Linear regression is one of the more fundamental forms of machine learning. Previous research had discussed how gestures performed with paretic extremities generally took longer than gestures performed with non-paretic extremities, and a difference in measurement of frequency domain values between paretic and non-paretic motion data. A method of predicting the total time of a gesture based on frequency domain values may serve as a method to assess the quality of upper extremity gestures. In this application, regression was considered to be a relevant field of study. Regression is a form of machine learning that generates an approximate output through the use of an unknown function and previous instances of input and their respectively known output values. This technique can be used to measure the significance of particular values to predict the values of a desired output.

4.3.1 Participants

Seven participants who had experienced stroke and met inclusion criteria were recruited from the University of Tennessee Medical Center (Table 4.1). All participants were recruited according to the rules of the Institutional Review Board of the University of Tennessee.

The participants were asked to perform the FMA while wearing the IMU sensors utilized for this research, where the motion data profiles for each gesture were recorded and documented. An instance of the motion data is presented in Figure 4.1.

Table 4.1: Demographics of Study Participants

Participant ID#	Study ID#	Gender	Impairment	Age (years)	FMA Score
5	1	F	R	24	59
7	2	F	R	67	48
8	3	F	L	68	53
9	4	M	R	86	36
10	5	F	L	43	45
12	6	F	R	86	30
13	7	M	R	64	27

4.3.2 Initial Data Partition

Through the application of several written MATLAB functions, The measured features for all individuals who performed the FMA were recorded into a large stacked structure.

4.3.3 Regression Models

The linear and nonlinear regression models generated through the linear and nonlinear fitting functions, `fitlm.m` and `fitnlm.m`, respectively, both utilize the same series of equations to determine if there is any difference in the error values. The general equation for the regression analysis is presented as $t = a * x^3 + b * y^2 + c * z$, where t represents the predicted output of the equation, and x, y, and z represent the features selected for the analysis, respectively. The values a, b, and c represent the weighted values that are generated by the regression models.

The combinations of frequency domain variables were selected to be only variables that belong to the same frequency range or to the same measured axis. The equations with variables that are in the same frequency domain range were chosen based on previous findings that frequency values measured from paretic gestures have a tendency to measure more values in a particular frequency range [37]. The equations with variables that are all measured along the same directional axis were chosen under the assumption that some gestures mainly require only one degree of freedom, such as shoulder flexion or shoulder abduction.

4.3.4 Linear Models

The MATLAB program, `NewRegression.m`, was written to run a large variety of linear regression fitting functions using data cataloged from recorded values from individuals who had previously performed the FMA. The results were documented into a spreadsheet for further assessment. The use of linear regression begins with generating a table containing the frequency domain values and total completion time for each gesture. Values that contained missing data were filtered as some gestures contained fewer gestures than others.

The tables were split into a training and testing set to test any models that will be generated within the program. A MATLAB function, `LinearRandomizer.m`, was written to consistently generate a randomized training and testing set that are closest to a determined ratio. The function was set such that each iteration will generate training and testing sets where the training set comprised of approximately 75% of the total data.

The pre-built MATLAB function, `fitlm.m`, is implemented to generate 36 linear models that attempt to fit the various combinations of frequency domain features to the desired output value through the use of training models. Values relevant to the training models, such as weights and intercepts, were documented and recorded on a separate spreadsheet.`fitlm.m` operates with a variety of options to customize linear fitting models, such as allowing intercept values, allowing robust fitting, and labeling input and output variables for clarification. The default setting to allow intercept values to be generated in the fitting models was allowed as forcing the fitting model to begin at zero would greatly reduce accuracy of any model generated. These fitted models are intended to operate in small windows and removing the intercept value may overgeneralize the results. The robust fitting option was not implemented in the analysis as a variety of robust fitting options are available in MATLAB and would increase the complexity of the regression function.

The pre-built MATLAB function, `predict.m`, is utilized to predict the output values through the training models and testing set values. The function, `LinearErrorTest.m`, performs two versions of error analysis. These two forms of error analysis are the squared error and absolute error equations and are utilized for measuring error in regression models [46].

$$\text{Squared Error: } E(g|X) = \frac{1}{N}\sum_{t=1}^{N}[r^t - g(x^t)]^2 \tag{4.1}$$

$$\text{Absolute Error: } E(g|X) = \frac{1}{N}\sum_{t=1}^{N}|[r^t - g(x^t)]| \tag{4.2}$$

The error values are also recorded in an Excel spreadsheet for further analysis.

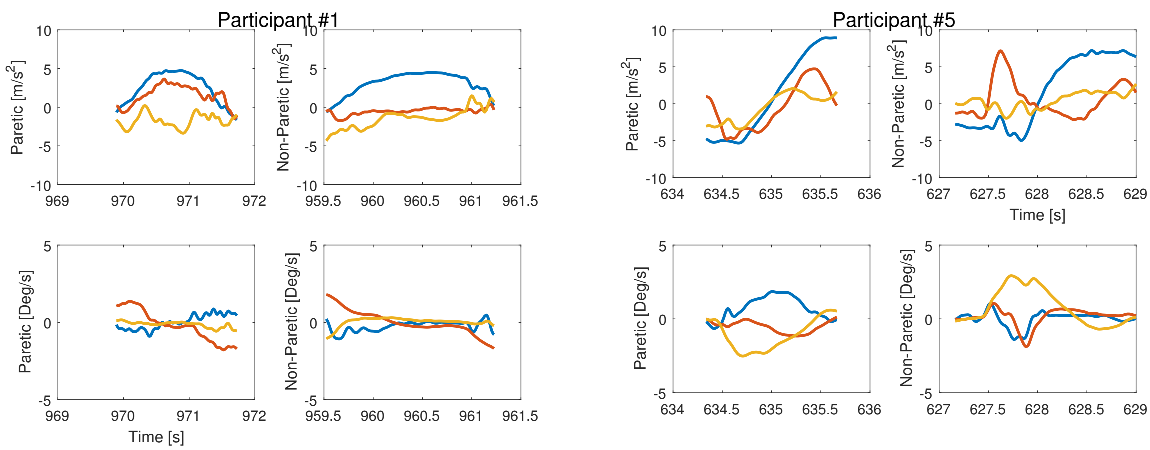

Figure 4.1: Accelerometer and rate gyroscope data for the *Out of Synergy, Shoulder Abduction* task for the paretic and non–paretic limbs of Participants 1 and 5. These data demonstrate patient variability as well as the typically larger amplitude and shorter duration of paretic limb performance.

4.3.5 Nonlinear Models

The MATLAB program, `NewRegression_Nonlinear.m`, was written to run a large variety of nonlinear regression models using the same data as in the linear model analysis. Similar to the linear model, a table is also created to record the same values and filter all missing data.

The tables were split into a training and testing set to test any models that will be generated within the program. A MATLAB function, `LinearRandomizer.m`, was written to consistently generate a randomized training and testing set that are closest to a determined ratio. The function was set such that each iteration will generate training and testing sets where the training set comprised of approximately 75% of the total data.

The pre-built MATLAB function, `fitnlm.m`, is implemented to generate the same 36 models from the linear analysis to create nonlinear regression models that predict the total time spent performing a gesture against various combinations of frequency domain features through the use of the training sets. Values relevant to the training models, such as weights and intercepts, were documented and recorded on a separate spreadsheet.`fitnlm.m` operates with a variety of options to customize linear fitting models, such as limiting attempts at fitting, allowing robust fitting, and labelling input and output variables for clarification. These fitted models are intended to operate in small windows and removing the intercept value may overgeneralize the results. The robust fitting option was not implemented in the analysis as a variety of robust fitting options are available in MATLAB and would increase the complexity of the regression function. The option to search and remove any samples that contained missing or invalid values from the analysis was left active as a secondary check to ensure that the analysis operates as intended. The default value for iteration attempts to minimize error was kept at the default value of 200 as it appeared to be a reasonable number of attempts for each equation to be analyzed and return a relative degree of accuracy.

The pre-built MATLAB funciton, `predict.m`, is utilized to predict the output values through the training models and testing set values. This process utilizes the same function from the linear models, `LinearErrorTest.m`, to perform error analysis.

4.3.6 Fugl-Meyer Data

The linear and nonlinear regression model fits were recreated where the output values are the Fugl-Meyer scores recorded for each individual when performing the gestures. A unique set of gesture features was created as the Fugl-Meyer score only applies to the gesture attempt that was the most accurate. The video footage of each individual's FMA was reviewed to visually assess the most optimal gesture attempts. These gestures were collected through the use of the written MATLAB program, `FMA_Score_Setup.m`, and saved into a .mat file for convenience.

Modified versions of the linear and nonlinear regression fitting programs, `FMA_Linear_Regression.m` and `FMA_Nonlinear_Regression.m`, were written to process, generate models, and document error values using FMA scores as an alternative to gesture time duration. Both programs utilize the same functions to generate training and testing sets, generate models, test models, predict values from testing sets, and measure error values from predicted values.

4.4 Results

4.4.1 Linear Time-based Regression

Tables 4.2 and 4.3 illustrate that the squared error for paretic and non-paretic linear regression models are lower than the absolute errors. The error values for either error metric for non-paretic regression models are lower than nonlinear regression models for paretic gesture errors. The paretic gesture with the lowest squared error and absolute error is Out of Synergy, Shoulder Flexion. The non-paretic gesture with the lowest squared error and absolute error is Flexor Synergy.

4.4.2 Nonlinear Time-based Regression

Tables 4.4 and 4.5 illustrate that the squared error for paretic and non-paretic nonlinear regression models are lower than the absolute errors. The error values for either error metric for non-paretic regression models are lower than nonlinear regression models for paretic gesture errors. The paretic gesture with the lowest squared error and absolute error is Out of Synergy, Shoulder Abduction. The non-paretic gesture with the lowest squared error and absolute error is Out of Synergy, Shoulder Flexion.

4.4.3 Linear FMA-based Regression

Tables 4.6 and 4.7 illustrate that the squared error for the FMA-based linear regression models appear to be higher than the absolute error values measured for a majority of flexor, extensor, wrist, and coordination-based gestures. The squared error values for the modified FMA-based linear regression models appear to be lower for than the absolute error with exception to Extensor (Elbow Extension and Forearm Pronation), coordination tremor, and coordination dysmetria. Some models generate error values that were equivalent for all tested regression models for both FMA and modified FMA-based linear regression models. Some equations generated results where all equations were measured with the same error values, and are documented as 'All Equal.'

4.4.4 Nonlinear FMA-based Regression

Tables 4.8 and 4.9 illustrate that the squared error values for the FMA-based nonlinear regression models were larger than absolute error values with the exception of all wrist-based gestures, coordination dysmetria, and coordination speed. The squared error values for the modified FMA-based nonlinear regression were either equal or lower than the absolute error, with the exception of coordination tremor and coordination dysmetria. The gesture with the lowest squared error and absolute error for the FMA-based nonlinear regression model is Coordination Dysmetria. The gesture with the lowest squared error and absolute error for the modified FMA-based nonlinear regression model is Out of Synergy, Pronation/Supination. Some equations generated results where all equations were measured with the same error values, and are documented as 'All Equal.'

Table 4.2: Results from Linear Analysis (Paretic)

Gesture	**Opt Eq.**	**Sq Diff Err**	**Abs Diff Err**
Flexor Synergy	$t = pxa_z^3 = pxc_z^2 + pxb_z$	0.0491	0.16054
Extensor Synergy	$t = pxc_z^3 + pxb_z^2 + pxa_z$	0.01329	0.09933
Synergies, Hand to Lumbar	$t = pxb_y^3 + pxa_y^2 + pxc_y$	0.02175	0.08507
Synergies, Shoulder Flexion	$t = pxa_x^3 + pxb_x^2 + pxc_x$	0.00188	0.03581
Synergies, Pronation/Supination	$t = pxc_y^3 + pxa_y^2 + pxb_y$	0.02108	0.11871
OoS, Shoulder Abduction	$t = pxc_x^3 + pxa_x^2 + pxb_x$	0.00386	0.04638
OoS, Shoulder Flexion	$t = pxb_y^3 + pxb_z^2 + pxb_x$	0.17782	0.32221
OoS, Pronation/Supination	$t = pxa_x^3 + pxa_y^2 + pxa_z$	0.00738	0.08127
Wrist-Stability, elbow at 90	$t = pxb_x^3 + pxa_x^2 + pxc_x$	0.01097	0.1017
Wrist-Flexion/extension,elbow at 90	$t = pxc_z^3 + pxc_x^2 + pxc_y$	0.24438	0.2976
Wrist-Stability, elbow at 0	$t = pxa_z^3 + pxa_y^2 + pxa_x$	0.07203	0.22022
Wrist-Flexion/extension, elbow at 0	$t = pxc_y^3 + pxc_x^2 + pxc_z$	0.01423	0.07911
Wrist Circumduction	$t = pxa_z^3 + pxa_y^2 + pxa_x$	0.27423	0.36172
Coordination Tremor	$t = pxa_z^3 + pxa_x^2 + pxa_y$	0.0032	0.0504

Table 4.3: Results from Linear Analysis (Non-Paretic)

Gesture	**Opt Eq.**	**Sq Diff Err**	**Abs Diff Err**
Flexor Synergy	$t = pxa_z^3 + pxb_z^2 + pxc_z$	0.00079	0.01954
Extensor Synergy	$t = pxb_x^3 + pxa_x^2 + pxc_x$	0.03394	0.14153
Synergies, Hand to Lumbar	$t = pxb_z^3 + pxc_z^2 + pxa_z$	0.00832	0.08459
Synergies, Shoulder Flexion	$t = pxa_x^3 + pxb_x^2 + pxc_x$	0.00545	0.07156
Synergies, Pronation/Supination	$t = pxc_z^3 + pxa_z^2 + pxb_z$	0.097	0.29371
OoS, Shoulder Abduction	$t = pxc_y^3 + pxb_y^2 + pxa_y$	0.02609	0.15634
OoS, Shoulder Flexion	$t = pxa_x^3 + pxb_x^2 + pxc_x$	0.02451	0.13161
OoS, Pronation/Supination	$t = pxa_x^3 + pxa_z^2 + pxa_y$	0.00324	0.04374
Wrist-Stability, elbow at 90	$t = pxc_x^3 + pxb_x^2 + pxa_x$	0.03	0.16796
Wrist-Flexion/extension,elbow at 90	$t = pxb_x^3 + pxa_x^2 + pxc_x$	0.11668	0.30069
Wrist-Stability, elbow at 0	$t = pxc_x^3 + pxb_x^2 + pxa_x$	0.0037	0.06031
Wrist-Flexion/extension, elbow at 0	$t = pxb_x^3 + pxa_x^2 + pxc_x$	0.93606	0.63567
Wrist Circumduction	$t = pxa_x^3 + pxb_x^2 + pxc_x$	0.10599	0.2707
Coordination Tremor	$t = pxa_x^3 + pxa_z^2 + pxa_y$	0.02275	0.12186

Table 4.4: Results from Nonlinear Analysis (Paretic)

Gesture	**Opt Eq.**	**Sq Diff Err**	**Abs Diff Err**
Flexor Synergy	$t = pxa_z^3 + pxb_z^2 + pxc_z$	0.10584	0.26799
Extensor Synergy	$t = pxa_y^3 + pxa_x^2 + pxa_z$	0.01657	0.11333
Synergies, Hand to Lumbar	$t = pxa_y^3 + pxb_y^2 + pxc_y$	0.00931	0.08599
Synergies, Shoulder Flexion	$t = pxc_x^3 + pxb_x^2 + pxa_x$	0.01447	0.09293
Synergies, Pronation/Supination	$t = pxa_z^3 + pxa_x^2 + pxa_y$	0.048	0.17744
OoS, Shoulder Abduction	$t = pxa_y^3 + pxa_z^2 + pxa_x$	0.00215	0.03053
OoS, Shoulder Flexion	$t = pxb_y^3 + pxc_y^2 + pxa_y$	0.00452	0.05846
OoS, Pronation/Supination	$t = pxb_z^3 + pxa_z^2 + pxc_z$	0.01521	0.1019
Wrist-Stability, elbow at 90	$t = pxa_x^3 + pxb_x^2 + pxc_x$	0.0124	0.10874
Wrist-Flexion/extension,elbow at 90	$t = pxc_x^3 + pxb_x^2 + pxa_x$	0.00913	0.06211
Wrist-Stability, elbow at 0	$t = pxa_x^3 + pxb_x^2 + pxc_x$	0.01677	0.11428
Wrist-Flexion/extension, elbow at 0	$t = pxb_z^3 + pxb_y^2 + pxb_x$	0.00406	0.05563
Wrist Circumduction	$t = pxc_x^3 + pxb_x^2 + pxa_x$	0.00602	0.07227
Coordination Tremor	$t = pxa_y^3 + pxa_z^2 + pxa_x$	0.00313	0.05049

Table 4.5: Results from Nonlinear Analysis (Non-Paretic)

Gesture	**Opt Eq.**	**Sq Diff Err**	**Abs Diff Err**
Flexor Synergy	$t = pxb_z^3 + pxa_z^2 + pxc_z$	0.00233	0.04193
Extensor Synergy	$t = pxc_z^3 + pxc_x^2 + pxc_y$	0.02011	0.10493
Synergies, Hand to Lumbar	$t = pxa_z^3 + pxb_z^2 + pxc_z$	0.00343	0.04834
Synergies, Shoulder Flexion	$t = pxa_x^3 + pxb_x^2 + pxc_x$	0.0018	0.03111
Synergies, Pronation/Supination	$t = pxc_y^3 + pxc_z^2 + pxc_x$	0.00634	0.062
OoS, Shoulder Abduction	$t = pxa_y^3 + pxc_y^2 + pxb_y$	0.00469	0.03968
OoS, Shoulder Flexion	$t = pxb_z^3 + pxb_x^2 + pxb_y$	0.000028	0.00529
OoS, Pronation/Supination	$t = pxa_y^3 + pxa_x^2 + pxa_z$	0.00072	0.01949
Wrist-Stability, elbow at 90	$t = pxa_z^3 + pxa_x^2 + pxa_y$	0.00217	0.04403
Wrist-Flexion/extension,elbow at 90	$t = pxc_y^3 + pxa_y^2 + pxb_y$	0.0027	0.04743
Wrist-Stability, elbow at 0	$t = pxc_x^3 + pxa_x^2 + pxb_x$	0.00072	0.02428
Wrist-Flexion/extension, elbow at 0	$t = pxc_z^3 + pxa_z^2 + pxb_z$	0.0045	0.06449
Wrist Circumduction	$t = pxb_x^3 + pxa_x^2 + pxc_x$	0.00109	0.02953
Coordination Tremor	$t = pxb_x^3 + pxc_x^2 + pxa_x$	0.00074	0.02316

Table 4.6: Results from Linear FMA Analysis

Gesture	**Opt Eq.**	**Sq Diff Err**	**Abs Diff Err**
Flexor-Elevation	$t = pxb_z^3 + pxa_z^2 + pxc_z$	0.130336	0.356801
Flexor-Shoulder Retraction	$t = pxb_z^3 + pxa_z^2 + pxc_x$	0.130336	0.356801
Flexor-Abduction(at least 90)	$t = pxb_x^3 + pxc_x^2 + pxa_x$	0.121482	0.344261
Flexor-External Rotation	All Equal	2.966892	1.274831
Flexor-Elbow Flexion	$t = pxb_y^3 + pxb_x^2 + pxb_z$ & $t = pxb_y^3 + pxb_z^2 + pxb_x$	6.692356	2.580006
Flexor-Forearm Supination	$t = pxc_x^3 + pxb_x^2 + pxa_x$	0.238191	0.460558
Extensor-Shoulder add./int.rot	All Equal	1.722568	1.186177
Extensor-Elbow Extension	$t = pxa_x^3 + pxa_z^2 + pxa_y$ & $t = pxa_z^3 + pxa_x^2 + pxa_y$	2.542927	1.21927
Extensor-Forearm pronation	$t = pxa_x^3 + pxa_z^2 + pxa_y$ & $t = pxa_z^3 + pxa_x^2 + pxa_y$	2.542927	1.21927
Movement combining synergies-Hand to Lumbar spine	$t = pxb_y^3 + pxb_x^2 + pxb_z$	1.605669	1.265285
Movement combining synergies-Shoulder flexion to 90	$t = pxc_x^3 + pxb_x^2 + pxa_x$	0.145824	0.289636
Movement combining synergies-Pronation of forearm	$t = pxa_y^3 + pxc_y^2 + pxb_y$	1.215257	0.965828
Movement out of synergy-Shoulder abduction to 90	All Equal	3.686884	1.762316
Movement out of synergy-Shoulder flexion 90-180	$t = pxa_x^3 + pxb_x^2 + pxc_x$ & $t = pxa_x^3 + pxc_x^2 + pxb_x$ & $t = pxb_x^3 + pxa_x^2 + pxc_x$ & $t = pxb_x^3 + pxc_x^2 + pxa_x$ & $t = pxc_x^3 + pxa_x^2 + pxb_x$ & $t = pxc_x^3 + pxb_x^2 + pxa_x$	0.068601	0.261917
Movement out of synergy-Pronation of forearm	$t = pxa_z^3 + pxb_z^2 + pxc_z$	3.151402	1.759335
Wrist-Stability, elbow at 90, shoulder at 0	$t = pxc_x^3 + pxc_z^2 + pxc_y$	4.428973	1.739308
Wrist-Flexion/extension,elbow at 90, shoulder at 0	$t = pxa_y^3 + pxb_y^2 + pxc_y$	3.175785	1.554728
Wrist-Stability, elbow at 0, shoulder at 30	$t = pxc_x^3 + pxc_y^2 + pxc_z$	3.275699	1.809865
Wrist-Flexion/extension, elbow at 0, shoulder at 30	$t = pxb_y^3 + pxc_y^2 + pxa_y$	0.879418	0.809071
Wrist Circumduction	$t = pxb_x^3 + pxc_x^2 + pxa_x$	0.840987	0.907166
Coordination-Tremor	$t = pxa_x^3 + pxa_y^2 + pxa_z$ & $t = pxa_y^3 + pxa_x^2 + pxa_z$	6.420031	2.533765
Coordination-Dysmetria	$t = pxc_x^3 + pxc_y^2 + pxc_z$	2.300377	1.37417
Coordination-Speed	$t = pxc_y^3 + pxb_y^2 + pxa_y$	2.319446	1.227444

Table 4.7: Results from Modified Linear FMA Analysis

Gesture	**Opt Eq.**	**Sq Diff Err**	**Abs Diff Err**
Flexor-Elevation	$t = pxb_x^3 + pxb_y^2 + pxb_z$ & $t = pxb_x^3 + pxb_z^2 + pxb_y$ & $t = pxb_y^3 + pxb_x^2 + pxb_z$ & $t = pxb_y^3 + pxb_z^2 + pxb_x$ & $t = pxb_z^3 + pxb_x^2 + pxb_y$ & $t = pxb_z^3 + pxb_y^2 + pxb_x$	0.55974	0.74777
Flexor-Shoulder Retraction	$t = pxb_x^3 + pxb_y^2 + pxb_z$ & $t = pxb_x^3 + pxb_z^2 + pxb_y$ & $t = pxb_y^3 + pxb_x^2 + pxb_z$ & $t = pxb_y^3 + pxb_z^2 + pxb_x$ & $t = pxb_z^3 + pxb_x^2 + pxb_y$ & $t = pxb_z^3 + pxb_y^2 + pxb_x$	0.33629	0.52431
Flexor-Abduction(at least 90)	$t = pxa_x^3 + pxa_y^2 + pxa_z$ & $t = pxa_x^3 + pxa_z^2 + pxa_y$ & $t = pxa_y^3 + pxa_x^2 + pxa_z$ & $t = pxa_y^3 + pxa_z^2 + pxa_y$ & $t = pxa_z^3 + pxa_y^2 + pxa_x$	0.10891	0.30864
Flexor-External Rotation	All Equal	0.5	0.5
Flexor-Elbow Flexion	$t = pxb_x^3 + pxb_y^2 + pxb_z$ & $t = pxb_x^3 + pxb_z^2 + pxb_y$ & $t = pxb_y^3 + pxb_x^2 + pxb_z$ & $t = pxb_y^3 + pxb_z^2 + pxb_x$ & $t = pxb_z^3 + pxb_x^2 + pxb_y$ & $t = pxb_z^3 + pxb_y^2 + pxb_x$	0.22659	0.47291
Flexor-Forearm Supination	$t = pxa_x^3 + pxb_x^2 + pxc_x$ & $t = pxa_x^3 + pxc_x^2 + pxb_x$ & $t = pxb_x^3 + pxa_x^2 + pxc_x$	0.53305	0.68672
Extensor-Shoulder add./int.rot	All Equal	1	1
Extensor-Elbow Extension	$t = pxc_x^3 + pxa_x^2 + pxb_x$	0.71367	0.62908
Extensor-Forearm pronation	$t = pxc_x^3 + pxa_x^2 + pxb_x$	0.71367	0.62908
Movement combining synergies-Hand to Lumbar spine	$t = pxb_x^3 + pxb_y^2 + pxb_z$ $t = pxb_x^3 + pxb_z^2 + pxb_y$ & $t = pxb_y^3 + pxb_x^2 + pxb_z$ & $t = pxb_y^3 + pxb_z^2 + pxb_x$ & $t = pxb_z^3 + pxb_x^2 + pxb_y$ & $t = pxb_z^3 + pxb_y^2 + pxb_x$	0.40758	0.63641
Movement combining synergies-Shoulder flexion to 90	$t = pxa_z^3 + pxc_z^2 + pxb_z$ & $t = pxb_z^3 + pxc_z^2 + pxa_z$ & $t = pxc_z^3 + pxa_z^2 + pxb_z$ & $t = pxc_z^3 + pxb_z^2 + pxa_z$	0.0647	0.24262
Movement combining synergies-Pronation of forearm	$t = pxa_z^3 + pxc_z^2 + pxb_z$	0.2182	0.44179
Movement out of synergy-Shoulder abduction to 90	All Equal	1	1
Movement out of synergy-Shoulder flexion 90-180	$t = pxa_x^3 + pxb_x^2 + pxc_x$ & $t = pxa_x^3 + pxc_x^2 + pxb_x$ & $t = pxb_x^3 + pxa_x^2 + pxc_x$ & $t = pxb_x^3 + pxc_x^2 + pxa_x$ & $t = pxc_x^3 + pxa_x^2 + pxb_x$ & $t = pxc_x^3 + pxb_x^2 + pxa_x$	0.068601	0.261917
Movement out of synergy-Pronation of forearm	$t = pxb_z^3 + pxc_z^2 + pxa_z$ & $t = pxc_z^3 + pxb_z^2 + pxa_z$	0.04248	0.16206
Wrist-Stability, elbow at 90, shoulder at 0	$t = pxa_x^3 + pxb_x^2 + pxc_x$ & $t = pxa_x^3 + pxc_x^2 + pxb_x$ & $t = pxb_x^3 + pxa_x^2 + pxc_x$ & $t = pxc_x^3 + pxa_x^2 + pxb_x$	0.02033	0.13149
Wrist-Flexion/extension,elbow at 90, shoulder at 0	All Equal	0.5	0.5
Wrist-Stability, elbow at 0, shoulder at 30	$t = pxa_x^3 + pxc_x^2 + pxb_x$ & $t = pxb_x^3 + pxc_x^2 + pxa_x$	0.46065	0.56497
Wrist-Flexion/extension, elbow at 0, shoulder at 30	All Equal	0.5	0.5
Wrist Circumduction	$t = pxa_x^3 + pxb_x^2 + pxc_x$ & $t = pxb_x^3 + pxa_x^2 + pxc_x$	0.2572	0.37601
Coordination-Tremor	$t = pxc_z^3 + pxc_y^2 + pxc_x$	1.23423	1.05579
Coordination-Dysmetria	$t = pxa_y^3 + pxc_y^2 + pxb_y$ & $t = pxb_y^3 + pxc_y^2 + pxa_y$	1.88914	1.12425
Coordination-Speed	$t = pxa_y^3 + pxc_y^2 + pxb_y$ & $t = pxb_y^3 + pxc_y^2 + pxa_y$	0.05191	0.22784

Table 4.8: Results from Nonlinear FMA Analysis

Gesture	**Opt Eq.**	**Sq Diff Err**	**Abs Diff Err**
Flexor-Elevation	$t = pxa_z^3 + pxb_z^2 + pxc_z$	19.84509	4.099171
Flexor-Shoulder Retraction	$t = pxa_z^3 + pxb_z^2 + pxc_z$	19.84509	4.099171
Flexor-Abduction(at least 90)	$t = pxa_x^3 + pxa_y^2 + pxa_z$	24.84882	4.8836
Flexor-External Rotation	All Equal	25.6274	4.93
Flexor-Elbow Flexion	$t = pxa_y^3 + pxb_y^2 + pxaz$	13.37032	3.22738
Flexor-Forearm Supination	$t = pxa_y^3 + pxb_y^2 + pxc_y$	18.28482	3.651849
Extensor-Shoulder add./int.rot	All Equal	17.1421	4.11
Extensor-Elbow Extension	$t = pxc_y^3 + pxc_x^2 + pxc_z$	5.461308	1.657551
Extensor-Forearm pronation	$t = pxc_y^3 + pxc_x^2 + pxc_z$	5.461308	1.657551
Movement combining synergies-Hand to Lumbar spine	$t = pxc_x^3 + pxc_z^2 + pxc_y$	4.998796	2.130839
Movement combining synergies-Shoulder flexion to 90	$t = pxa_x^3 + pxa_z^2 + pxa_y$	9.099362	3.006457
Movement combining synergies-Pronation of forearm	$t = pxa_y^3 + pxa_z^2 + pxa_x$	1.818578	1.327413
Movement out of synergy-Shoulder abduction to 90	All Equal	4.20345	1.965
Movement out of synergy-Shoulder flexion 90-180	Err	Err	Err
Movement out of synergy-Pronation of forearm	$t = pxa_x^3 + pxa_y^2 + pxa_z$	6.584486	2.560697
Wrist-Stability, elbow at 90, shoulder at 0	$t = pxb_x^3 + pxc_x^2 + pxa_x$	0.284018	0.381928
Wrist-Flexion/extension,elbow at 90, shoulder at 0	$t = pxc_z^3 + pxb_z^2 + pxa_z$	0.097153	0.238633
Wrist-Stability, elbow at 0, shoulder at 30	$t = pxc_y^3 + pxa_y^2 + pxb_y$	0.866339	0.903895
Wrist-Flexion/extension, elbow at 0, shoulder at 30	$t = pxa_z^3 + pxa_y^2 + pxa_x$	0.079325	0.209875
Wrist Circumduction	$t = pxb_x^3 + pxa_x^2 + pxc_x$	0.495568	0.650132
Coordination-Tremor	$t = pxa_y^3 + pxc_y^2 + pxb_y$	0.814573	0.800932
Coordination-Dysmetria	$t = pxc_y^3 + pxc_x^2 + pxc_z$	0.005217	0.067957
Coordination-Speed	$t = pxc_x^3 + pxc_z^2 + pxc_y$	0.189099	0.43402

Table 4.9: Results from Modified Nonlinear FMA Analysis

Gesture	**Opt Eq.**	**Sq Diff Err**	**Abs Diff Err**
Flexor-Elevation	$t = pxb_x^3 + pxb_y^2 + pxb_z$	0.33398	0.55697
Flexor-Shoulder Retraction	$t = pxa_z^3 + pxb_z^2 + pxc_z$	0.18644	0.36439
Flexor-Abduction(at least 90)	$t = pxa_y^3 + pxa_z^2 + pxa_x$	0.1081	0.31167
Flexor-External Rotation	All Equal	0.5	0.5
Flexor-Elbow Flexion	$t = pxb_x^3 + pxb_y^2 + pxb_z$	0.07725	0.21023
Flexor-Forearm Supination	$t = pxb_y^3 + pxb_x^2 + pxb_z$	0.46384	0.6502
Extensor-Shoulder add./int.rot	All Equal	1	1
Extensor-Elbow Extension	$t = pxa_z^3 + pxa_y^2 + pxa_x$	0.24621	0.46361
Extensor-Forearm pronation	$t = pxa_z^3 + pxa_y^2 + pxa_x$	0.24621	0.46361
Movement combining synergies-Hand to Lumbar spine	$t = pxb_y^3 + pxb_x^2 + pxb_z$	0.2661	0.3822
Movement combining synergies-Shoulder flexion to 90	$t = pxc_z^3 + pxb_z^2 + pxa_z$	0.09673	0.29684
Movement combining synergies-Pronation of forearm	$t = pxb_z^3 + pxb_y^2 + pxb_x$	0.01748	0.1317
Movement out of synergy-Shoulder abduction to 90	All Equal	1	1
Movement out of synergy-Shoulder flexion 90-180	Err	Err	Err
Movement out of synergy-Pronation of forearm	$t = pxb_z^3 + pxc_z^2 + pxa_z$	0.0011	0.0277
Wrist-Stability, elbow at 90, shoulder at 0	$t = pxa_x^3 + pxb_x^2 + pxc_x$	0.00592	0.07656
Wrist-Flexion/extension,elbow at 90, shoulder at 0	All Equal	0.5	0.5
Wrist-Stability, elbow at 0, shoulder at 30	$t = pxa_x^3 + pxb_x^2 + pxc_x$	0.22784	0.38499
Wrist-Flexion/extension, elbow at 0, shoulder at 30	All Equal	0.5	0.5
Wrist Circumduction	$t = pxa_y^3 + pxa_z^2 + pxa_x$	0.07025	0.24348
Coordination-Tremor	$t = pxb_y^3 + pxc_y^2 + pxa_y$	5.14305	1.96445
Coordination-Dysmetria	$t = pxa_y^3 + pxc_y^2 + pxb_y$	1.49395	0.94765
Coordination-Speed	$t = pxa_y^3 + pxc_y^2 + pxb_y$	0.00377	0.05414

4.5 Discussion

4.5.1 Nonlinear vs. Linear for Paretic

Our results suggest that Nonlinear and Linear regression models for gestures performed with a paretic extremity may perform more optimally depending on the gesture in question. These findings may suggest that due to the varying level of complexity in a gesture, as well as variables in performance of gestures, it is more difficult to perform regression analysis on paretic data. Dokkum et al. discuss that hemiparetic movement involves more sub-movements and results in a less smooth trajectory [53]. This is only increased for gestures that require inter-joint coordination and are not treating the upper extremity as a rigid body, resulting in more variance in the measured data. Regression models can attempt to overfit data when there is more variation in the input data. Nonlinear models are more likely to overfit data with higher variance, while linear regression models are less capable due being more limited in robust nature [46].

These findings can show relevance in further research towards gesture recognition in paretic upper extremities. These results demonstrate that models utilized for paretic gestures should be considered with respect to the particular gesture when implementing linear or nonlinear regression models. Some limitations in the study include the limitation of samples for use in the study, as smaller samples may increase the likelihood of machine learning algorithms carrying bias in any analysis. This could be improved in further research by collecting data from more participants to expand the data sample size implemented. Further research could look into the use of other regression models, such as logarithmic regression.

4.5.2 Nonlinear vs. Linear for Non-paretic

Our findings suggest that Nonlinear and Linear regression models for gestures performed with a non-paretic extremity generally perform more accurately with a nonlinear model as compared to a linear model. These findings may suggest that while nonlinear models may be more complex, they are generating more accurate results. While non-paretic extremities are still affected by hemiparetic stroke, their performance is relatively closer to how they performed prior to stroke. When there are a relatively abundant amount of samples and a relatively low value of variance, regression models operate the most ideally [46].

These findings can show relevance in further research towards gesture recognition in non-paretic upper extremities. Unlike the results with comparing nonlinear and linear regression models for paretic gestures, these results suggest that nonlinear regression models are much more ideal for any regression model.

Some limitations in the study are mentioned prior with with comparing linear and nonlinear gestures in that there are a small sample size utilized in this research. Recommendations for further research have also been mentioned prior regarding logarithmic regression.

4.5.3 Paretic vs. Non-paretic

Overall results between Paretic and Non-paretic based models suggest that regression models generally produce more accurate models for non-paretic gestures as compared to paretic. This can suggest that non-paretic gestures are more capable of predicting with regression models than non-paretic. Dokkum et al. discuss the larger amounts of sub-movements in hemiparetic stroke and Alpaydin discusses how regression models have a tendency to generate more accurate models when there is less variation [53, 46].

This may be relevant to fields such as computer or health science, as it may be work performing regression models for paretic extremities based on models derived for non-paretic extremities.

Some limitations in the study are mentioned prior with with comparing linear and nonlinear gestures in that there are a small sample size utilized in this research. Recommendations for further research have also been mentioned prior regarding logarithmic regression.

Chapter 5

Differentiating between Paretic and Non-Paretic Limb Performance

5.1 Introduction

A number of features, from visual inspection, differentiate paretic and non–paretic limb performance. Many of these features are time domain dependent [54]. The goal of the current approach is to determine if a broader set of features are useful for differentiating between paretic and non–paretic limb performance. If successful, such a tool may be useful for tracking changes in paretic limb capability over time in continuous in–home data.

5.2 Background

Individuals affected by stroke will have a likelihood to perform movements in a method that is not consistent with how they will perform prior. Hemiparetic movements will often appear more rigid and contain more discrete, smaller movements to correct their trajectory due to neuromotor noise [53].

Clustering techniques such as k–nearest neighbors have primarily been used to distinguish between different types of gestures using wearable sensor data. In previous studies, the application of KNN with accelerometer data has proven useful recognizing specific physical activities using lower extremities, differentiating between types of daily activities, and predicting neurological episodes [55, 56, 57].

5.3 Methods

KNN is a machine learning classifier that assesses data based on proximity to other data points and their relative classification. The intention of using a clustering technique is that while there are many ways an individual may have an impairment, individuals who are not impaired will generate fairly consistent results. Using statistical features from individuals who have experienced a hemiparetic stroke, it is believed that KNN may be capable to discern between gestures that are either paretic or non-paretic based on the values of these statistical features.

The APDM wearable sensor is capable of measuring both linear acceleration and angular velocity through tri-axial accelerometers and rate gyroscopes, respectively. While the costs of sensors have slowly decreased and allowed for this sort of technology to be applied in more commercial environments, a sensor that only measures one metric with minimal difference in efficacy will be more cost efficient. Any instance of a sensor that only uses a single modality will be more ideal from a financial perspective to finance and utilize for practical applications.

Part of the experiment is to determine if a binary KNN classifier can be applied to upper extremity gestures of individuals post stroke. A large quantity of samples from all relevant FMA gestures are utilized for the KNN algorithms implemented from both paretic and non-paretic upper extremities, along with a variety of parameters relating to modalities collected from the APDM wearable sensors.

5.3.1 Model Setup

To run the KNN analysis, a series of sets needed to be generated in order to create a model and test and validity or efficacy of the model. To run a KNN analysis, a portion of the collected data, usually between 50% to 75% of the total data, is documented as the training set and is used to create a classifier model. This classifier model is designed to classify and interpret data based on the samples and results from the training set with the assumption that the training set represents an average population with minimal bias.

The remaining portion of data is referred as the testing set and is used to measure the accuracy of the classifier model by comparing the output classifications predicted by the model against the output classifications documented in the samples. Normally the accuracy of a binary classifier can be measured through a confusion matrix, which measures the total instances that sample was classified as any number of times a sample was classified and whether that is the correct classification.

5.3.2 Software Implementation

Several custom and built-in MATLAB functions were used in this analysis. Using the .mat files generated prior from the `Setup.m` MATLAB script, extracted features belonging to either accelerometer or rate gyroscope modalities were stored into respective matrices using the written MATLAB script, `MatrixMaker.m`, along with a third matrix that classifies each sample as either paretic or non-paretic. The output matrices created by the function contained only accelerometer-based features, only rate gyroscope-based features, or a combination of both features.

The matrices were later split into two smaller matrices for training and testing sets. The written MATLAB script, `MatrixRandomizer.m`, uses a consistent random number generator to randomize all data sets such that the randomized orientation of all samples will match the randomized orientation of the classifier values. The script also duplicates each matrix as it is intended to compare the results of the same sets where one undergoes principle component analysis (PCA) and the alternative sets do not undergo PCA.

A simple MATLAB function was written, `confmat.m`, to calculate the accuracy of all KNN models using the measurements used in a confusion matrix. While a confusion matrix will have four possible outcomes (True Positive, True Negative, False Positive, False Negative), the MATLAB function combines the both true outputs and both false outputs. Despite the difference in categorization, the measurement of accuracy is the same as it measures the total number of accurate or true measurements over all attempted measurements.

The primary MATLAB script, `KNNAnalysis.m`, performed the KNN analysis on six unique combinations of values to determine which set of parameters may have the highest accuracy. For each unique set, the documented training set is implemented in the built-in function, `fitcknn.m`, which generates a KNN-based training model using the training set and associated output classification as input values. Using the testing set parameters, the built-in MATLAB function `predict.m` uses the testing set and prior KNN model developed by `fitcknn.m` to predict the output values for the testing set using the KNN training model. The overall accuracy of the model is assessed by comparing the classifier outputs generated by `predict.m` against the classifier outputs previously documented from the samples and calculated through `confmat.m`.

5.3.3 Comparing Models

The KNN analysis was conducted on six sets of data to determine the level of accuracy achieved with the provided data. The first three sets included a set containing only accelerometer-based features, a set containing only rate gyroscope-based features, and a set containing both accelerometer and rate gyroscope-based features. The first three sets only received z-standardization in terms of data pre-processing in order for each feature used in the study to have more influence due to larger values instead of variability. Z-standardization is a form of data pre-processing that standardizes all values such that they have a mean of zero and standard deviation of one. This standardization technique is used as it is common for various machine learning applications to ensure that a feature is not interpreted as more significant than another feature because of numeric value instead of a The second three sets include the same variety of modality-based features, but was also subject to principle component analysis, which modifies the data such that each column of data is linearly independent and organized based on variability.

5.4 Results

This section was originally published by Zachariah Nelson and Eric Wade: Relative Efficacy of Sensor Modalities for Estimating Post-Stroke Motor Impairment, IEEE Engineering in Medicine and Biology, Honolulu, HI, 2018

The class posterior probabilities of the six KNN analyses were utilized to determine model performance. For the $z-$normalized data without PCA, the combined data sets performed the best, with accelerometer–only showing reduced performance. Rate Gyroscope–only model performed the worst of the three. For the PCA data, the combination performed identically to the rate gyroscope–only model, and both better than the accelerometer–only. In both the accelerometer– and rate gyroscope–only datasets, the application of PCA improved model performance. However, the application of PCA slightly decreased the performance of the combination model. The results are presented in Table 5.1

Table 5.1: Results from KNN Analyses

Features	$z-$**norm**	$z-$**norm + PCA**
Accelerometer	70.15%	72.38%
Rate gyroscope	65.67%	69.40%
Accelerometer & Rate gyro	73.13%	72.38%

5.5 Discussion

This section was originally published by Zachariah Nelson and Eric Wade: Relative Efficacy of Sensor Modalities for Estimating Post-Stroke Motor Impairment, IEEE Engineering in Medicine and Biology, Honolulu, HI, 2018

5.5.1 Relative Performance of Sensor Modalities

The classification errors of the $z-$standardized data indicate that when taken alone, the accelerometer demonstrates better performance than the rate gyroscope. The use of both sensor modalities results in the best performance, suggesting they contain complementary information. The differences in individual sensor modality performance may be due to the properties of the movements. Since impairment can be described as joint level limitations, FMA task performance depends on joint range–of–motion. Thus, a more impaired individual may have reduced motion amplitude proportional to reduced joint capability. A translational sensor (e.g., the accelerometer) may therefore be more sensitive to the FMA tasks.

This can become relevant to choices in sensors used for therapy, as it is demonstrated that multiple modalities will be beneficial to performance. Limitations to the analysis were mentioned prior in chapter four regarding sample size. Further exploration of this concept may require the use of other assessments that measure disability at the impairment level of the ICF model.

5.5.2 Relative Performance of PCA

Given the number of features, we sought to investigate if the commonly used PCA feature selection method will alter the performance of the classifier as a reduced feature set will likely result in improved computational cost, relevant to the eventual deployment of this approach. Despite the improvement in accelerometer and rate gyroscope alone classifiers, the PCA transformed data resulted in no difference between rate gyroscope alone and the combined accelerometer an rate gyroscope data. Further, when compared to the $z-$standardized data, the accuracy of the combined classifier decreased. This may be due to the underlying nature of the complementary information of the accelerometer and rate gyroscope; specifically, if the variability in these data are uniformly aligned with one of the principal components, the other orthogonal components may cause the two classes to be indistinguishable [58].

The implication of this research can benefit stroke rehabilitation as the results imply that the PCA may reduce the accuracy of the classifier. Further analyses will explore dimensional reduction approaches (such as kernel PCA, which may mitigate this variability limitation) to determine the role, and relative importance, of the full feature set.

5.5.3 Overall Model Performance

While other systems using sensor data and functional assessments demonstrate higher classification accuracies, our results differ due to the overall purpose of the approach of the current study. Other research including our own, has addressed the problem of comparing or predicting assessment scores using sensor data [59, 24, 60, 61, 62]. The goal here is ultimately to use continuously monitored motion data to track longitudinal changes in impairment, as measured by motor activity. Though new segmenting techniques are being developed, the variability associated with upper extremity task performance renders it difficult to train recognition models for every type of activity. Therefore, features sensitive to impairment that may be taken, for instance, from sliding window data (e.g., frequency domain features, jerk, signal magnitude area) may be ideal for such continuous monitoring. Models capable of incorporating these metrics may prove useful for the eventual application domain.

Chapter 6

Discussion

6.1 Regression

6.1.1 Comparison of Linear and Nonlinear Models

Our findings suggest that nonlinear regression models generate more accurate prediction models than linear regression models when analyzing non-paretic Data, while only specific nonlinear regression models yielded more accurate predictions than linear with respect to paretic gestures. These results may advocate that nonlinear models may perform more efficiently or equivalently to linear models for non-paretic gestures. Nonlinear models are more complex and can allow for regression models to fit the data more accurately than linear [46]. The low error values may also recommend that total gesture time may be predicted through the use of frequency domain features.

These findings may be relevant to physical and occupational therapists as they demonstrate that a model can be developed that can compare frequency domain values to a continuous range of values, potentially allowing for therapists to better measure the rate of improvement an extremity may perform specific gestures.

The relevance of the findings are that a more efficient regression model could yield more accurate results from sensor data and assist in providing feedback to a physical or occupational therapist and improve quality of therapy sessions for individuals post-stroke.

The sample size of the participants was small with respect to the research conducted. The amount of some recorded gestures are less than other gestures, as the frequency of successful performances varied between participants. This can suggest that further study with larger samples may produce models for some gestures that are more accurate.

6.1.2 Comparison of Total Samples to FMA Scores

Results from the FMA-based regression models propose that linear models using only paretic gesture data to predict FMA scores may be the most efficient model. This is likely that the use of six variables instead of three may have over-defined the model [63]. This can be relevant to the topic of measuring gesture performance against a quantifiable values such as the FMA.

The sample size of the participants was small with respect to the research conducted. The amount of some recorded gestures are less than other gestures, as the frequency of successful performances varied between participants. It is also worth considering that FMA scores are meant to be discrete values and regression models are more designed for continuous values. The FMA only scores the optimal gesture in any session, so the sample size was severely diminished for all FMA-based regression models. This can propose that further study with larger samples may produce models for some gestures that are more accurate.

6.2 Clustering

6.2.1 Significance of Analyzed Modalities

Our findings propose that a binary KNN classifier performs optimally when features from both accelerometer and rate gyroscope sensors are utilized as compared to use individually. This suggests that the output values from both sensor modalities may provide complimentary information. These findings can also recommend that wearable sensors may not benefit from the use of PCA if both modalities are utilized, as the accuracy of the KNN model decreased.

Our binary classification of upper-extremity motion quality was less accurate than the analysis performed by Dolatabadi et al. [64]. It should be noted, however, that their analysis was towards differentiating the gait performance of healthy individuals and individuals post stroke, while our analysis has been focused on differentiating between paretic and non-paretic upper extremity gestures of individuals post stroke. It can also be noted that their analysis utilized features derived from the orientation of appendages and average velocity from motion-capture sensors, whereas our analysis utilized features derived from linear acceleration and angular velocity from IMU sensors.

The relevant importance of these results can suggest that wearable IMU sensors can recognize upper extremity gestures as either paretic or non-paretic. As mentioned prior, this could benefit individuals post-stroke undergoing therapy.

The sample size utilized in research was relatively small as some subjects can not be used due to missing data from faulty sensors. As mentioned in Chapter 4, the results are limited by a smaller sample size.

A subject that remains to be explored is if modality accuracies can improve if more than one IMU sensor is utilized when measuring each extremity. It can be relevant to determine if more accurate results may be obtained if features measured from sensors located along the upper arm or trunk are also assessed. Further research can be performed regarding the validity of the modality accuracies with a more robust sample or if both modalities are unnecessary when analyzing specific gestures. Another topic of research can be to determine if KNN can be applied to recognizing FMA scores, as these results may be more meaningful to occupational therapists providing feedback is sensors can assess that a particular gesture is showing improvement with respect to the FMA.

Stroke is a serious illness that requires extensive physical therapy to overcome. Use of sensors with modalities that can provide greater accuracy can allow for physical therapy to be supplemented with information collected from an individual's home environment.

There are a few results from the analysis to take away from the thesis. Analysis of KNN proposes that combinations of accelerometer and rate gyroscope modalities leads to more accurate measurement of paretic and non-paretic extremities as compared to individual modality use. The regression analysis suggests that nonlinear models are generally more accurate than linear models. These results may hold relevance in the concept of neuroscience and neurorehabilitation, as the KNN results submits the relevance of multiple modalities providing more accurate results when analyzing gestures post stroke and the regression results proposes that particular models for each gesture may result in optimal prediction of gesture time through methods such as ”window mapping”. These methods can be applicable to health sciences by demonstrating the promising concept of gesture recognition of sensors worn in external environments to supplement and provide feedback for physical and occupational therapy.

6.3 Contributions

Relevant contributions to the subject include the development of MATLAB programs that import H5 files, extracting meaningful motion data from those files, and performs a variety of signal processing and feature extraction functions. While this is not relevant to a large scientific community, this does hold significance for future researchers to have a more streamlined approach to processing IMU sensor data.

Contributions made with respect to regression methods include opening the possibility for further analysis of testing linear and nonlinear regression to measure to predict results based on frequency domain values. This work has also contributed to promote the concept that FMA scores could be measured through the use of frequency domain values.

Further contributions were made for KNN classification as this work demonstrates that gestures can be recognized between paretic and non-paretic. We have also contributed to the neurorehabilitation by demonstrating that more accurate results are achieved through the use of combining accelerometer and rate gyroscope modalities instead of only implementing a single modality.

Chapter 7

Conclusion

The purpose of this research study was to investigate the application of machine learning techniques to quantitative motion data for individuals post stroke. Through wearable sensors, therapists may receive more feedback regarding the activity levels of patients and where how to better focus future sessions. While more accurate results may currently be achieved from video capturing systems in a lab environment, wearable sensors can be utilized in external environments and provide recommendations to the participant in use of the sensors will act in a more natural fashion.

Analysis of linear and nonlinear regression models to predict total gesture time through the use of frequency domain values suggest that nonlinear models are equal or more accurate when compared to linear models.

Comparison of accuracy in gesture recognition with through KNN of various modalities propose that combining features from accelerometers and rate gyroscopes result in the most accurate results. The measured accuracy may advocate that KNN and the features utilized may warrant future research in classifying performance of upper extremities post stroke.

Bibliography

[1] E. J. Benjamin, M. J. Blaha, S. E. Chiuve, and M. Cushman, *Heart Disease and Stroke Statistics2017 Update*, 2017, vol. 135, no. 10. 4

[2] E. Taub, G. Uswatte, V. W. Mark, and D. M. Morris, "The learned nonuse phenomenon: implications for rehabilitation," *Eura Medicophys*, vol. 42, no. 3, pp. 241–256, Sep 2006. 4

[3] J. W. Krakauer, "Motor learning: its relevance to stroke recovery and neurorehabilitation," *Curr. Opin. Neurol.*, vol. 19, no. 1, pp. 84–90, Feb 2006. 5

[4] ——, "Arm function after stroke: from physiology to recovery," *Semin Neurol*, vol. 25, no. 4, pp. 384–395, Dec 2005. 5

[5] C. B. Berthelsen, T. Lindhardt, and K. Frederiksen, "A discussion of differences in preparation, performance and postreflections in participant observations within two grounded theory approaches," *Scand J Caring Sci*, vol. 31, no. 2, pp. 413–420, Jun 2017. 5

[6] M. Huijben-Schoenmakers, C. Gamel, and T. B. Hafsteinsdottir, "Filling up the hours: how do stroke patients on a rehabilitation nursing home spend the day?" *Clin Rehabil*, vol. 23, no. 12, pp. 1145–1150, Dec 2009. 5

[7] A. Althubaiti, "Information bias in health research: definition, pitfalls, and adjustment methods," *J Multidiscip Healthc*, vol. 9, pp. 211–217, 2016. 5

[8] D. R. Taber, J. Stevens, D. M. Murray, J. P. Elder, L. S. Webber, J. B. Jobe, and L. A. Lytle, "The effect of a physical activity intervention on bias in self-reported activity," *Ann Epidemiol*, vol. 19, no. 5, pp. 316–322, May 2009. 5

[9] APDM Wearable Technologies, "Image of apdm sensor," [Online; accessed March 20, 2018]. [Online]. Available: https://www.apdm.com/wp-content/uploads/2015/05/OpalFace-e1435608694957.jpg 5

[10] ——, "Image of individual wearing apdm sensor," [Online; accessed march 20, 2018]. [Online]. Available: https://www.apdm.com/wp-content/uploads/2015/05/apdm0037-e1431128218628.jpg 5

[11] Thalmic Labs, "Image of myo armband," [Online; accessed march 20, 2018]. [Online]. Available: https://static.thalmic.com/sapphire/techspecs/myo_image_black.png 6

[12] B. H. Dobkin and L. Angeles, "Physical Activities," vol. 26, no. 6, pp. 602–608, 2003. [Online]. Available: http://ec.europa.eu/public{_}opinion/archives/ebs/ebs{_}183{_}6{_}en.pdf 6

[13] F. Gemperle, C. Kasabach, J. Stivoric, M. Bauer, and R. Martin, "Design for wearability," in *Digest of Papers. Second International Symposium on Wearable Computers (Cat. No.98EX215)*, Oct 1998, pp. 116–122. 6

[14] M. K. O'Brien, N. Shawen, C. K. Mummidisetty, S. Kaur, X. Bo, C. Poellabauer, K. Kording, and A. Jayaraman, "Activity Recognition for Persons With Stroke Using Mobile Phone Technology: Toward Improved Performance in a Home Setting," *J. Med. Internet Res.*, vol. 19, no. 5, p. e184, May 2017. 7

[15] G. D. Fulk, S. R. Edgar, R. Bierwirth, P. Hart, P. Lopez-Meyer, and E. Sazonov, "Identifying activity levels and steps of people with stroke using a novel shoe-based sensor," *J Neurol Phys Ther*, vol. 36, no. 2, pp. 100–107, Jun 2012. 7

[16] D. J. Gladstone, C. J. Danells, and S. E. Black, "The Fugl-Meyer Assessment of Motor Recovery after Stroke: A Critical Review of Its Measurement Properties," *Neurorehabilitation and Neural Repair*, vol. 16, no. 3, pp. 232–240, 2002. [Online]. Available: http://nnr.sagepub.com/cgi/doi/10.1177/154596802401105171 7

[17] K. J. Sullivan, J. K. Tilson, S. Y. Cen, D. K. Rose, J. Hershberg, A. Correa, J. Gallichio, M. McLeod, C. Moore, S. S. Wu, and P. W. Duncan, "Fugl-Meyer assessment of sensorimotor function after stroke: standardized training procedure for clinical practice and clinical trials," *Stroke*, vol. 42, no. 2, pp. 427–432, Feb 2011. 7

[18] J. See, L. Dodakian, C. Chou, V. Chan, A. McKenzie, D. J. Reinkensmeyer, and S. C. Cramer, "A standardized approach to the Fugl-Meyer assessment and its implications for clinical trials," *Neurorehabil Neural Repair*, vol. 27, no. 8, pp. 732–741, Oct 2013. 7

[19] S. L. Wolf, P. A. Catlin, M. Ellis, A. Link Archer, B. Morgan, and A. Piacentino, "Stroke Assessing Wolf Motor Function Test as Outcome Measure for Research in Patients After Assessing Wolf Motor Function Test as Outcome Measure for Research in Patients After Stroke Subjects and Methods," *Stroke*, vol. 32, pp. 1635–1639, 2001. [Online]. Available: http://stroke.ahajournals.org/content/32/7/1635{%}0Ahttp://stroke.ahajournals.org//subscriptions/{%}0Ahttp://stroke.ahajournals.org/ 8

[20] T. M. Hodics, K. Nakatsuka, B. Upreti, A. Alex, P. S. Smith, and J. C. Pezzullo, "Wolf Motor Function Test for characterizing moderate to severe hemiparesis in stroke patients," *Arch Phys Med Rehabil*, vol. 93, no. 11, pp. 1963–1967, Nov 2012. 8

[21] S. V. Duff, J. He, M. A. Nelsen, C. J. Lane, V. T. Rowe, S. L. Wolf, A. W. Dromerick, and C. J. Winstein, "Interrater reliability of the Wolf Motor Function Test-Functional Ability Scale: why it matters," *Neurorehabil Neural Repair*, vol. 29, no. 5, pp. 436–443, Jun 2015. 8

[22] P. Otten, J. Kim, and S. H. Son, "A Framework to Automate Assessment of Upper-Limb Motor Function Impairment: A Feasibility Study," *Sensors (Basel)*, vol. 15, no. 8, pp. 20 097–20 114, Aug 2015. 8

[23] L. Yu, D. Xiong, L. Guo, and J. Wang, "A remote quantitative Fugl-Meyer assessment framework for stroke patients based on wearable sensor networks," *Comput Methods Programs Biomed*, vol. 128, pp. 100–110, May 2016. 8

[24] S. Del Din, S. Patel, C. Cobelli, and P. Bonato, "Estimating Fugl-Meyer clinical scores in stroke survivors using wearable sensors," *Conf Proc IEEE Eng Med Biol Soc*, vol. 2011, pp. 5839–5842, 2011. 8, 44

[25] S. Patel, R. Hughes, T. Hester, J. Stein, M. Akay, J. G. Dy, and P. Bonato, "A novel approach to monitor rehabilitation outcomes in stroke survivors using wearable technology," *Proceedings of the IEEE*, vol. 98, no. 3, pp. 450–461, March 2010. 8

[26] S. M. Michaelsen and M. F. Levin, "Short-term effects of practice with trunk restraint on reaching movements in patients with chronic stroke: a controlled trial," *Stroke*, vol. 35, no. 8, pp. 1914–1919, Aug 2004. 8

[27] T. A. Jones and S. C. Jefferson, "Reflections of experience-expectant development in repair of the adult damaged brain," *Dev Psychobiol*, vol. 53, no. 5, pp. 466–475, Jul 2011. 8

[28] G. Uswatte, W. H. Miltner, B. Foo, M. Varma, S. Moran, and E. Taub, "Objective measurement of functional upper-extremity movement using accelerometer recordings transformed with a threshold filter," *Stroke*, vol. 31, no. 3, pp. 662–667, 2000. 8

[29] A. Mansfield, J. S. Wong, J. Bryce, K. Brunton, E. L. Inness, S. Knorr, S. Jones, B. Taati, and W. E. McIlroy, "Use of Accelerometer-Based Feedback of Walking Activity for Appraising Progress With Walking-Related Goals in Inpatient Stroke Rehabilitation: A Randomized Controlled Trial," *Neurorehabil Neural Repair*, vol. 29, no. 9, pp. 847–857, Oct 2015. 8

[30] D. Kumar, J. Gubbi, B. Yan, and M. Palaniswami, "Motor recovery monitoring in post acute stroke patients using wireless accelerometer and cross-correlation," *Conf Proc IEEE Eng Med Biol Soc*, vol. 2013, pp. 6703–6706, 2013. 8

[31] K. Gordt, T. Gerhardy, B. Najafi, and M. Schwenk, "Effects of Wearable Sensor-Based Balance and Gait Training on Balance, Gait, and Functional Performance in Healthy and Patient Populations: A Systematic Review and Meta-Analysis of Randomized Controlled Trials," *Gerontology*, vol. 64, no. 1, pp. 74–89, 2018. 9

[32] E. Fortune, M. M. Morrow, and K. R. Kaufman, "Assessment of gait kinetics using triaxial accelerometers," *J Appl Biomech*, vol. 30, no. 5, pp. 668–674, Oct 2014. 9

[33] A. Godfrey, S. Del Din, G. Barry, J. C. Mathers, and L. Rochester, "Instrumenting gait with an accelerometer: a system and algorithm examination," *Med Eng Phys*, vol. 37, no. 4, pp. 400–407, Apr 2015. 9

[34] E. Taub, G. Uswatte, V. W. Mark, and D. M. Morris, "The learned nonuse phenomenon: implications for rehabilitation," *Eura Medicophys*, vol. 42, no. 3, pp. 241–256, Sep 2006. 9

[35] A. Sterr, S. Freivogel, and D. Schmalohr, "Neurobehavioral aspects of recovery: assessment of the learned nonuse phenomenon in hemiparetic adolescents," *Arch Phys Med Rehabil*, vol. 83, no. 12, pp. 1726–1731, Dec 2002. 9

[36] M. Osumi, M. Sumitani, Y. Otake, and S. Morioka, "A "matched" sensory reference can guide goal-directed movements of the affected hand in central post-stroke sensory ataxia," *Exp Brain Res*, Feb 2018. 9

[37] E. Wade, C. Chen, and C. J. Winstein, "Spectral analyses of wrist motion in individuals poststroke: the development of a performance measure with promise for unsupervised settings," *Neurorehabil Neural Repair*, vol. 28, no. 2, pp. 169–178, Feb 2014. 9, 10, 22

[38] R. Osu, K. Ota, T. Fujiwara, Y. Otaka, M. Kawato, and M. Liu, "Quantifying the quality of hand movement in stroke patients through three-dimensional curvature," *J Neuroeng Rehabil*, vol. 8, p. 62, Oct 2011. 9

[39] I. Carpinella, D. Cattaneo, and M. Ferrarin, "Quantitative assessment of upper limb motor function in Multiple Sclerosis using an instrumented Action Research Arm Test," *J Neuroeng Rehabil*, vol. 11, p. 67, Apr 2014. 9

[40] M. S. Totty and E. Wade, "Muscle Activation and Inertial Motion Data for Non-Invasive Classification of Activities of Daily Living," *IEEE Trans Biomed Eng*, Aug 2017. 10

[41] S. H. Roy, M. S. Cheng, S. S. Chang, J. Moore, G. De Luca, S. H. Nawab, and C. J. De Luca, "A combined sEMG and accelerometer system for monitoring functional activity in stroke," *IEEE Trans Neural Syst Rehabil Eng*, vol. 17, no. 6, pp. 585–594, Dec 2009. 10

[42] S. D. Bersch, D. Azzi, R. Khusainov, I. E. Achumba, and J. Ries, "Sensor data acquisition and processing parameters for human activity classification," *Sensors (Basel)*, vol. 14, no. 3, pp. 4239–4270, Mar 2014. 10, 21

[43] M. A. Urbin, K. J. Waddell, and C. E. Lang, "Acceleration metrics are responsive to change in upper extremity function of stroke survivors," *Arch Phys Med Rehabil*, vol. 96, no. 5, pp. 854–861, May 2015. 10

[44] M. A. Urbin, R. R. Bailey, and C. E. Lang, "Validity of body-worn sensor acceleration metrics to index upper extremity function in hemiparetic stroke," *J Neurol Phys Ther*, vol. 39, no. 2, pp. 111–118, Apr 2015. 10

[45] F. Attal, S. Mohammed, M. Dedabrishvili, F. Chamroukhi, L. Oukhellou, and Y. Amirat, "Physical Human Activity Recognition Using Wearable Sensors," *Sensors*, vol. 15, no. 12, pp. 31 314–31 338, 2015. [Online]. Available: http://www.mdpi.com/1424-8220/15/12/29858 11

[46] E. Alpaydin, *Introduction to Machine Learning*, 3rd ed. Cambridge, Massachusetts: The MIT Press, 2014. 11, 24, 36, 37, 45

[47] MATLAB, *version 7.10.0 (R2010a)*. Natick, Massachusetts: The MathWorks Inc., 2010. 14

[48] T. Nakamura, T. Nagai, D. Mochihashi, I. Kobayashi, H. Asoh, and M. Kaneko, "Segmenting Continuous Motions with Hidden Semi-markov Models and Gaussian Processes," *Front Neurorobot*, vol. 11, p. 67, 2017. 20

[49] J. Alon, V. Athitsos, Q. Yuan, and S. Sclaroff, "A unified framework for gesture recognition and spatiotemporal gesture segmentation," *IEEE Trans Pattern Anal Mach Intell*, vol. 31, no. 9, pp. 1685–1699, Sep 2009. 20

[50] J. Lopes, M. Simo, N. Mendes, M. Safeea, J. Afonso, and P. Neto, "Hand/arm gesture segmentation by motion using imu and emg sensing," *Procedia Manufacturing*, vol. 11, pp. 107 – 113, 2017, 27th International Conference on Flexible Automation and Intelligent Manufacturing, FAIM2017, 27-30 June 2017, Modena, Italy. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2351978917303645 20

[51] M. A. Simo, P. Neto, and O. Gibaru, "Unsupervised gesture segmentation of a real-time data stream in matlab," in *IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society*, Oct 2016, pp. 809–814. 21

[52] G. Kang and S. Guo, "Variable sliding window dtw speech identification algorithm," in *2009 Ninth International Conference on Hybrid Intelligent Systems*, vol. 1, Aug 2009, pp. 304–307. 21

[53] L. van Dokkum, I. Hauret, D. Mottet, J. Froger, J. Metrot, and I. Laffont, "The contribution of kinematics in the assessment of upper limb motor recovery early after stroke," *Neurorehabil Neural Repair*, vol. 28, no. 1, pp. 4–12, Jan 2014. 36, 37, 38

[54] R. Khusainov, D. Azzi, I. E. Achumba, and S. D. Bersch, "Real-time human ambulation, activity, and physiological monitoring: taxonomy of issues, techniques, applications, challenges and limitations," *Sensors (Basel)*, vol. 13, no. 10, pp. 12 852–12 902, Sep 2013. 38

[55] W. Song, C. Ade, R. Broxterman, T. Barstow, T. Nelson, and S. Warren, "Activity recognition in planetary navigation field tests using classification algorithms applied to accelerometer data," *Conf Proc IEEE Eng Med Biol Soc*, vol. 2012, pp. 1586–1589, 2012. 39

[56] A. T. Ozdemir and B. Barshan, "Detecting falls with wearable sensors using machine learning techniques," *Sensors (Basel)*, vol. 14, no. 6, pp. 10 691–10 708, Jun 2014. 39

[57] P. Ghaderyan, A. Abbasi, and M. H. Sedaaghi, "An efficient seizure prediction method using KNN-based undersampling and linear frequency measures," *J. Neurosci. Methods*, vol. 232, pp. 134–142, Jul 2014. 39

[58] S. Karamizadeh, S. Abdullah, A. Manaf, M. Zamani, and A. Hooman, "An Overview of Principal Component Analysis," *Journal of Signal and Information Processing*, vol. 4, pp. 173–175, 2013. 43

[59] A. Parnandi, E. Wade, and M. Mataric, "Motor function assessment using wearable inertial sensors," *Conf Proc IEEE Eng Med Biol Soc*, vol. 2010, pp. 86–89, 2010. 44

[60] Y. W. Hsieh, K. C. Lin, C. Y. Wu, H. Y. Lien, J. L. Chen, C. C. Chen, and W. H. Chang, "Predicting clinically significant changes in motor and functional outcomes after robot-assisted stroke rehabilitation," *Arch Phys Med Rehabil*, vol. 95, no. 2, pp. 316–321, Feb 2014. 44

[61] N. Gebruers, S. Truijen, S. Engelborghs, and P. P. De Deyn, "Prediction of upper limb recovery, general disability, and rehabilitation status by activity measurements assessed by accelerometers or the Fugl-Meyer score in acute stroke," *Am J Phys Med Rehabil*, vol. 93, no. 3, pp. 245–252, Mar 2014. 44

[62] H. L. Chen, K. C. Lin, Y. W. Hsieh, C. Y. Wu, R. J. Liing, and C. L. Chen, "A study of predictive validity, responsiveness, and minimal clinically important difference of arm accelerometer in real-world activity of patients with chronic stroke," *Clin Rehabil*, vol. 32, no. 1, pp. 75–83, Jan 2018. 44

[63] P. C. Austin and E. W. Steyerberg, "The number of subjects per variable required in linear regression analyses," *J Clin Epidemiol*, vol. 68, no. 6, pp. 627–636, Jun 2015. 46

[64] E. Dolatabadi, B. Taati, and A. Mihailidis, "Automated classification of pathological gait after stroke using ubiquitous sensing technology," *Conf Proc IEEE Eng Med Biol Soc*, vol. 2016, pp. 6150–6153, 08 2016. 46

Vita

Zachariah Nelson was born in Nashville, TN, in 1994. He graduated with a bachelor's degree in Mechanical Engineering at Lipscomb University in Nashville in 2016 under a strict four–year plan. He continued his education at the University of Tennessee under the tutelage of Dr. Eric Wade. During that time, he made contributions to the MABE department in the form of publications and involvement in outreach programs to promote STEM awareness. His research interests are in wearable sensing, therapeutic robotics, and medical technology.