MUSCLE LOAD PREDICTION MODEL USING WEARABLE SENSORS

by
İbrahim Kerem Tokdemir
B.S., Mechanical Engineering, Boğaziçi University, 2020

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Systems and Control Engineering
Boğaziçi University
2025

ACKNOWLEDGEMENTS

First of all, I would like to express my gratitude to my advisor Assoc. Prof. Sema Dumanlı Oktar for her support and guidance during this study. This study would not have been completed without her instructions. It was special for me to study with her. Also, I would like to thank all the members of BOUNTENNA Research Laboratory, with particular highlight to Burak Ferhat Özcan.

I appreciate Prof. Can Yücesoy for letting me work in the Biomechanics Lab. I must express a special thanks to Beste İmamoğlu Yıldırım, who has been patient and helpful during experiments. I also thank to Prof. Alev Taşkın, for taking her time and valuable comments.

I am grateful to my academic advisor, Prof. Yağmur Denizhan, who paved the way for me to start this thesis by introducing me to Assoc. Prof. Sema Dumanlı Oktar.

I also thank to Tolga Gök, who has been the key person to arrange necessary equipments used in this study. His patience and willingness could not have been gone unnoticed.

Last but definitely not least, I thank to my parents Sedat and Meral, my dear sister Merve, who supported me no matter how the outcome will be. I felt unconditional love from them everyday. No words can reveal how meaningful and important the presence of İlsu ,my loved one and life long friend, who pushed and backed me in the most difficult times during this thesis.

A very important thanks goes to my most loyal friend Daisy Luna. She made me smile in the most stressful days.

ABSTRACT

MUSCLE LOAD PREDICTION MODEL USING WEARABLE SENSORS

Assessment of workload intensity and muscle load measurement is important to understand the quality of the intended motion. There have been numerous attempts to quantify workload levels for different motion for different parts of the body using wearable sensors. This thesis investigates human locomotion, particularly walking and running, which are fundamental activities of daily life. Electromyography (EMG) is a technique to quantify the level of muscle activity. In this study, the aim is to lay foundation for feature engineering and to build a machine learning model to predict EMG sensor output with minimal number of sensors during walking. Two pressure insole sensors and a single EMG - IMU combined sensor have been used during the experiments. The process starts with building a machine learning model to predict EMG sensor output using only raw input signals, which provided a poor accuracy. In order to improve the model, lagged features, derivatives and window-based statistics like moving averages and standard deviations are added to feature set and accuracy is improved. Then the focus is put on hyperparameter tuning and regularization is introduced to decrease overfitting. Although these steps led to incremental improvements, the test performance eventually plateaued with overfitting. In order to overcome this issue, data is shuffled before splitting it into training and test sets. After shuffling during training and test data split, the model delivered its best results—both training and test accuracy improved, and overfitting was significantly reduced.

ÖZET

GİYİLEBİLİR SENSÖRLER İLE KAS YÜKÜ TAHMİNİ

Egzersiz yoğunluğunun ve kas aktivitesinin ölçülmesi, istenilen hareketin kalitesini anlamak açısından büyük önem taşımaktadır. Giyilebilir sensörler kullanılarak vücudun farklı bölgelerine yönelik, çeşitli hareketler için yük seviyelerini niceliksel olarak belirlemeye yönelik çok sayıda çalışma gerçekleştirilmiştir. Bu tezde günlük yaşamın en temel fiziksel aktiviteleri arasında yer alan yürüme ele alınmıştır. Elektromiyografi (EMG), kas aktivitesinin düzeyini ölçmek için kullanılan bir tekniktir. Bu çalışmada amaç, özellik mühendisliği için bir temel oluşturmak ve yürüyüş sırasında minimum sayıda sensör kullanarak EMG sensör çıktısını tahmin edebilen bir makine öğrenmesi modeli geliştirmektir. Deneylerde iki adet basınç ölçebilen ayak tabanı sensörü ile bir adet kombine EMG ve IMU sensörü kullanılmıştır. İlk aşamada, yalnızca ham giriş sinyalleri kullanılarak EMG çıktısını tahmin eden bir makine öğrenmesi modeli oluşturulmuş ve zayıf bir doğruluk elde edilmiştir. Modelin performansını artırmak amacıyla, gecikmeli özellikler, türevler ve hareketli ortalama ile standart sapma gibi pencere tabanlı istatistiksel özellikler veri setine eklenmiş ve doğruluk oranı iyileştirilmiştir. Sonrasında hiperparametre optimizasyonu ve düzenlileştirme teknikleri uygulanarak aşırı öğrenme azaltılmaya çalışılmıştır. Bu adımlar doğrulukta kademeli artışlar sağlasa da, test performansı bir noktada durağanlaşmıştır. Bu sorunun üstesinden gelmek için veri, eğitim ve test setlerine ayrılmadan önce rastgele karıştırılmıştır. Bu yöntemle model, hem eğitim hem de test verilerinde en iyi sonuçlarını vermiş; doğruluk artmış ve aşırı öğrenme önemli ölçüde azaltılmıştır.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

a, b	Butterworth filter coefficients
a_k	Denominator coefficient of digital filter
$b^{(2)}$	Bias term for MLP output layer
$b_j^{(1)}$	Bias term for hidden neuron j in MLP
b_k	Numerator coefficient of digital filter
f_c	Cutoff frequency of the low-pass filter [Hz]
f_N	Nyquist frequency, half the sampling frequency [Hz]
f_s	Sampling frequency of the signal [Hz]
$H(j\omega)$	Frequency response of the filter
$H(z)$	Transfer function of the digital filter
M	Number of trees in a random forest
n	Filter order (e.g., $n = 2$ for second-order)
N	Number of data samples
T	Number of leaves in a decision tree
t	Continuous time variable
w	Weight vector in linear or neural models
$w_j^{(2)}$	Weight from hidden neuron j to output in MLP
$w_{ji}^{(1)}$	Weight from input i to hidden neuron j in MLP
$x(t)$	Input signal
$x[n]$	Discrete input signal
x_i	Input vector for sample i
$x_{\text{rectified}}(t)$	Rectified EMG signal
$x_{\text{envelope}}(t)$	Envelope of the EMG signal after filtering
$y[n]$	Filtered discrete signal
y_i	Target value for sample i
$\hat{y}$	Model output (prediction)
$\hat{y}_i$	Predicted value for sample i

α	L2 regularization term for multi-layer perceptron model
ϵ	SVR margin of tolerance (epsilon-insensitive zone)
γ	RBF kernel width parameter in SVR
λ	Regularization strength (L1 or L2)
λ_1, λ_2	Elastic net L1 and L2 regularization weights
μ	Mean of the signal
ω_c	Normalized cutoff frequency (relative to Nyquist)
Ω	Regularization term in XGBoost loss function

LIST OF ACRONYMS/ABBREVIATIONS

EMG	Electromyography
FIFA	Fédération Internationale de Football Association
FSR	Force Sensing Resistor
GPS	Global Positioning System
Hb	Deoxygenated hemoglobin
HbO_2	Oxygenated hemoglobin
Hz	Hertz
IMU	Inertial Measurement Unit
MEMS	Micro Electro-Mechanical Systems
ML	Machine Learning
MLP	Multi-layer Perceptron
MRI	Magnetic Resonance Imaging
MSE	Mean Squared Error
MVC	Maximum Voluntary Contraction
NIRS	Near Infrared Spectrophotometry
RBF	Radial Basis Function
RFR	Random Forest Regressor
sEMG	Surface Electromyography
SmO_2	Muscle Oxygen Saturation
SpO_2	Oxygen Saturation
SVM	Support Vector Machine
T2	Transverse Relaxation Time
XGBoost	Extreme Gradient Boosting Regressor Model

1. INTRODUCTION

Walking and running, which are integral parts of human locomotion, yield a significant amount of daily life activities, especially for athletes. Comfort and ability of humans to move with desired agility depends on quality of walking and running. Quantitative insight on those activities require biomechanical analysis of muscles used to perform the intended movements. Biomechanical aspect of the human body is a very complex system that is a combination of mechanisms- muscles, joints, knees- and a control system induced by brain and center nervous system. Although people do not perceive the complexity during walking and running, the process includes various interactions between muscles, joints, nervous system, and ground. Therefore, study of movement has always been under scope of scientists and engineers.

This study aims to build a model that could be substitute for EMG sensor. The idea is to build a machine learning model that predicts output of EMG sensor. Inverse and forward biomechanical models include a lot of equations with many variables. Variables include orientation of the body, ankle-joint torques, velocities, angular acceleration, and ground reaction forces. Calculation and parametrization of biomechanical models are time and effort consuming processes. Inputs of the classical inverse and forward biomechanical models will be selected as features of machine learning model. Machine learning models, specifically deep learning models, are popular substitutes for complex physical problems. In theory, deep learning models are capable of capturing non-linear high dimensional behaviors. There are attempts to build machine learning models to replace or reduce dependence on physics based model in biomechanics [6] [7]. With the help of advancements in computing and cloud technologies, power for training of complex models is available to wider range of society.

1.1. History of Movement and Muscle Load Analysis

History of gait analysis could be started from 500-300 BCE in Greece. Aristotle (384–322 BCE) observed and claimed gait includes phases of lowering and rising of the body due to movement of knees [8]. The first notable experiment in gait analysis field is conducted by Giovanni Alfonso Borelli (1608-1679), a student of Galileo. Borelli has puplished the first study regarding mechanics of muscles. Borelli also stuied locomotion of animals, about which he has written book ”De Motu Animalum (1680)” [9]. Governing physical equations proposed by Borelli has been proven wrong with the book ”Philosophiæ Naturalis Principia Mathematica” written by Isaac Newton's(1642–1727). The first scientist applied Newton's study on human movement was Hermann Boerhaave (1668–1738) in Netherlands [10]. In time period between late 18th. century and early 19th. century, progress in the fiels was limited due to two reasons. The first is reason is the lack of experiments in the era. The other reason was scientists were not able to compose physics and physiology. Ernst Heinrich Weber (1795–1878) and Eduard Willhelm Weber (1806–1871) conducted experiments by using stop watch, measuring tape and a telescope. They have studied limb movement during different phases of gait. The next big step in the field was development of a shoe with three pressure sensors attached by Jules Etienne Marey (1830–1904) and his student his student Gaston Carlet (1849–1892). Marey also investigated movement of horses, which were main transportation animal. 20th. century marks the introduction of force plate utilization in experiments [11].

Modern era started at the Biomechanics Lab at the University of California at Berkeley. Verne T. Inman (1905–1980) led the team and laid foundation for presenting data for gait analysis and published book ”Human Walking” [12]. With the advancement of force plates, electromyography (EMG), and emerging inverse dynamics, deeper studies have been conducted to analyze muscle activation, joint torques, and forces in muscles [13] [14] [15].

1.2. Surface Electromyography

Electromyography (EMG) signals reflect the electrical activity linked to voluntary muscle contractions, which correspond to muscle tension. Each contraction originates from a motor unit, which is consisted of a single alpha motor neuron and the muscle fibers it controls. When a nerve impulse reaches the fibers and crosses the depolarization threshold, it triggers muscle action potentials, creating a measurable voltage. These action potentials generate an electromagnetic field as they spread along the muscle membrane. The combined electrical activity of all fibers in one motor unit forms the motor unit action potential. The EMG signal represents the total activity of all motor unit potentials detected by the electrode [16]. A representation of decomposition of an EMG signal together with the motor unit on which EMG is attached is given in Figure 1.1. Therefore, EMG is a technique to quantify the level of activity of the muscle on which an electrode is placed. The amplitude of the EMG signal refers to the extent of muscle activation and the duration of the EMG signal reflects the duration of muscle activity.

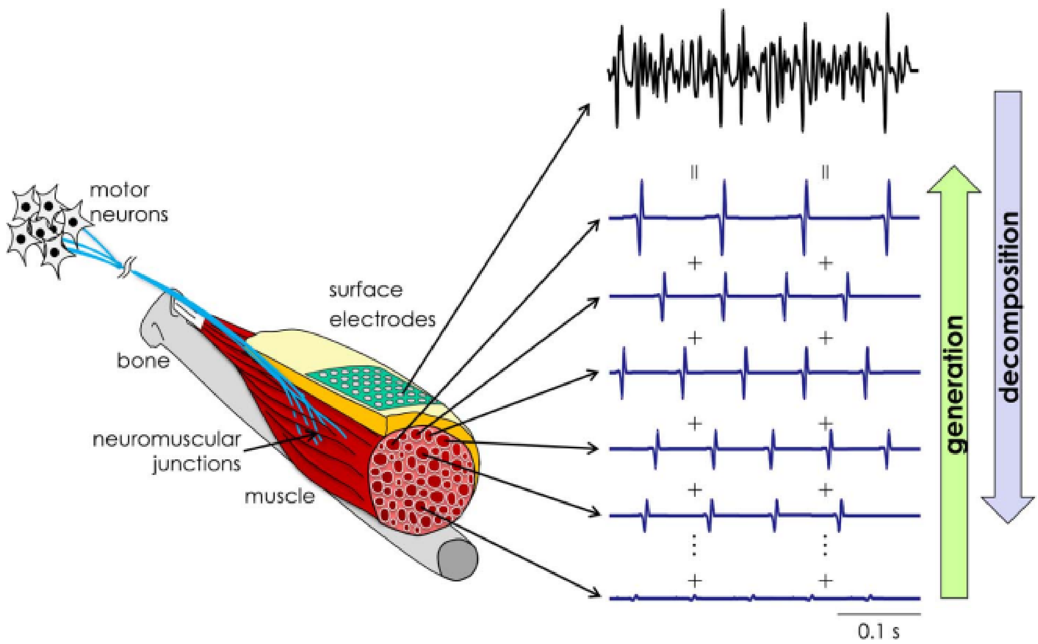

Figure 1.1. EMG signal is composed of all signals generated from fibers it is attached [1].

Surface Electromyography (sEMG) is a non-invasive method to generate EMG signals from the muscle [17]. Electrodes are placed on the skin above the muscle of interest. EMG signal depends on all the action potentials of the muscle fibers beneath the skin [18]. Another electromyography method is an invasive technique. A needle electrode is used for invasive EMG. Figure 1.2 demonstrates application of both EMG types.

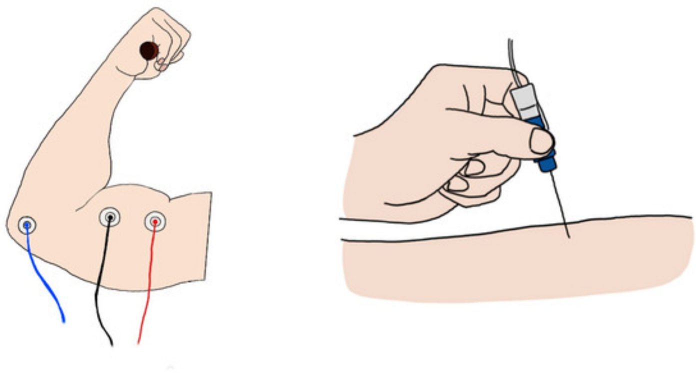

Figure 1.2. An illustration of (a) sEMG and (b) needle EMG [2].

Surface electromyography sensors are used in the fields of physiotherapy, biomechanics, and sports science. The applications of sEMG sensors vary in a broad range that include assessment of gait quality, posture disturbances, rehabilitation, kinesiology, and prosthetic development [19]. sEMG sensors are utilized to assess muscle synergy and body coordination quality, injury risk, recovery time in different sports [20].

Surface electromyography signals require processing for several reasons. One of the reason is that the sEMG sensor signals include noise. There are different noise sources. Inherent noise is typically the noise that is produced by all electronic components. Silver, silver-chloride electrodes provide sufficient signal-to-noise ratio [21]. Higher electrode size can improve signal-to-noise ratio, however it could lead to reduced specification and crosstalk which is another noise source. Crosstalk noise is the signal intervention from another muscle source which is not under investigation. Another source of noise is movement artifact that refers to noise due to shape and orientation change of electrodes during the movement. Electromagnetic noise could also be observed if a power source is nearby during data acquisition. Filtering, amplitude analysis, wavelet analysis are among the techniques used to process the sEMG signal before further the data analysis [22] [23].

1.3. Inertial Measurement Units

An Inertial Measurement Unit (IMU) is an electronic device that is composed of various sensors, including accelerometers, gyroscopes, and magnetometers as shown in Figure 1.3. Applications of IMU began with military guidance systems but was too costly for wider use. Advances in micro electro-mechanical systems (MEMS) and computing power have made inertial measurement units available for broader applications [24].

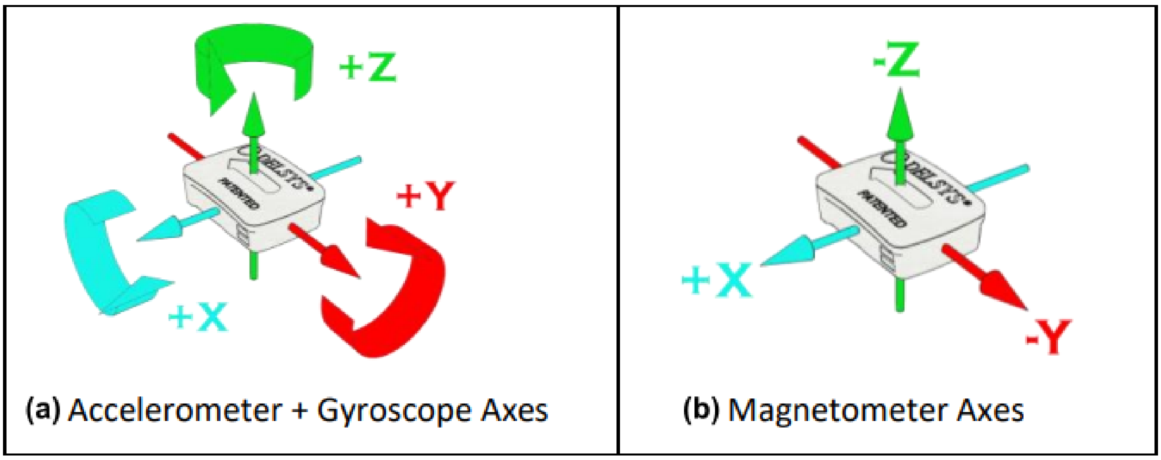

Figure 1.3. An internal Inertial Measurement Unit (IMU) is composed of (a) accelerometers, gyroscopes, and (b) magnetometers, outputs 3 DOF acceleration data and 3 DOF rotational data.

Gait analysis has been an area where IMU are used. In gait analysis, several methods are used to get required parameters to infer the orientation of the limbs. These parameters are position, velocity and acceleration of extremities. Image processing is a method to obtain those parameters, where markers are instrumented on extremities and by using special cameras the position is obtained. Another method is to instrument Inertial Measurement Units (IMU) to obtain acceleration, and then velocity and position by integration of acceleration values [25]. In [26], both methods have been compared, where a goniometer is used as baseline, then image processing and IMU approach are applied to measure angle between foot and calf. Image processing based approach yielded higher accuracy than IMU method. In [27], both techniques are combined to improve gait analysis where data from two different approaches are synchronized, and it is mentioned that IMU-based approach is more user-friendly and suitable for gait analysis than visual approach.

1.4. Motivation

The purpose of this research is to explore possibility of modelling an EMG sensor for lower extremity. A machine learning model is developed to capture EMG output. The model will output expected EMG signal given pressure insole input, body orientation and movement. This study will open path for anomaly detection through comparing EMG sensor and a model output. In [28], it is noted that the ground reaction forces alter during running when a Kinesio tape is applied to runner. The EMG model will be built with data obtained with no alteration on movement capability of athlete. Once the model is built, then its output will be compared with EMG sensor signals. Comparison will be done with respect to current movement of the athlete.

2. LITERATURE REVIEW

Using principles of biomechanics and several sensors, researchers have attempted to predict muscle activity and estimate workload of athletes under different conditions and movements.

2.1. Muscle Load Prediction

With the advancement of telemetric heart rate sensors, an attempt to measure intensity of a training by using heart rate of an athlete has been made. However, since heart rate could be affected by psychological reasons, utilization of heart rate alone does not give a full understanding of training intensity. Global Positioning Systems (GPS) have been used together with telemetric hear rate sensor to improve quality of training intensity analysis. GPS are used to track the movement of the athletes to characterize workload of an athlete. However, global positioning systems have drawbacks due to inability to track athletes for indoor sports and infeasibility to infer intensity for short movement sports [29]. Combination of telemetric heart rate sensors and GPS is the most common and commercialized application in sports. There are 49 wearable sensor systems approved by Fédération Internationale de Football Association (FIFA), which includes a combination of heart rate sensors and tracking systems.

Oxygen levels in tissue have also been used to assess muscle load. Oxygen saturation (SpO_2), shows the ratio of oxygen-saturated hemoglobin relative to total hemoglobin. During rest, oxygen saturation is between 95%-100% for healthy humans. Low to moderate intensity physical activity generally keeps oxygen saturation within normal range, but high-intensity training may result in a reduction in these levels [30]. Change in the levels of oxygenated hemoglobin (HbO_2), and de-oxygenated hemoglobin (Hb) can denote muscle activation too. During muscle contraction, oxygen consumption in the muscle increases. Oxygenated hemoglobin (HbO_2), and deoxygenated hemoglobin (Hb) are good indicators of oxygen consumption in tissues. Figure 2.1 shows the change of HbO_2 and Hb with changing maximal voluntary contraction (MVC). Oxygen muscle saturation (SmO_2), another compund with oxygen that changes with respect to workload intensity. Oxygen levels in a tissue can be measured by both invasive and non-invasive methods. One of the applicable non-invasive method during a training session is near-infrared spectrophotometry (NIRS), which uses optical properties of the tissue. Utilizing NIRS, levels of HbO_2 and SpO_2 are possible to be measured [3] [31].

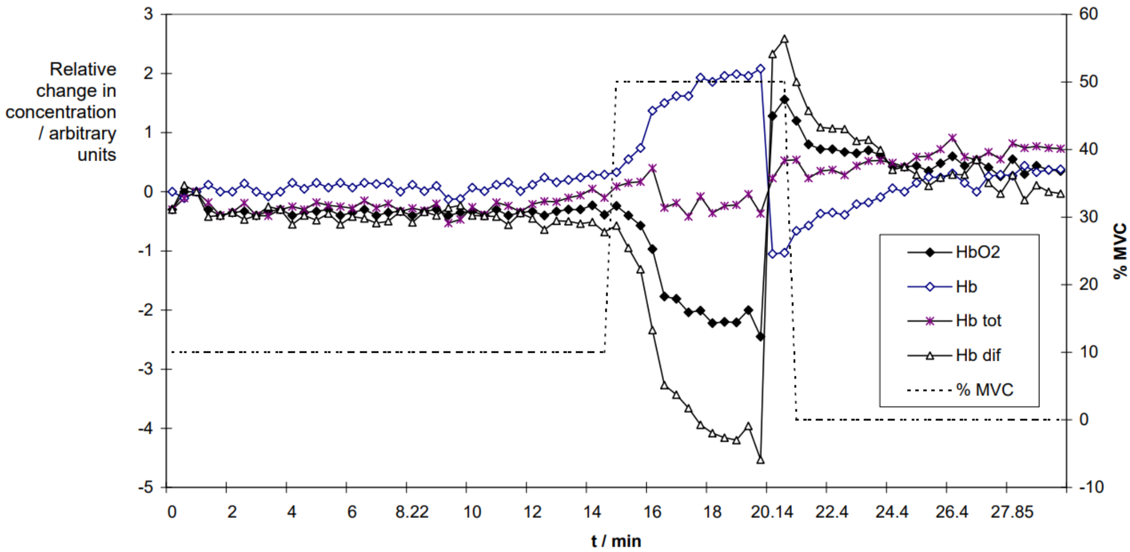

Figure 2.1. HbO_2, Hb (obtained by NIRS) with respect to MVC. At minute 16 the activation increase of MVF from 10% to 50% trigger changes in HbO_2, Hb [3].

Another method that is used frequently to assess muscle activity and training intensity relies on volume and shape change of muscles during activity. Force sensing resistors and strain gauges are devices used for that method. Resistance of force sensing resistor (FSR), changes as a force is applied on its surface. A strain gauge is a device whose resistance changes with respect to stretching or bending of its shape. Volume change on a muscle, which is a result of muscle load in terms of contraction or relaxation, can be interpreted through FSR and straing gauges. In [32], force sensing resistors are used to assess the activations of upper leg muscles. Combinations of force sensing sensors and strain gauges have also been used to evaluate muscle activity in different studies [33] [34] [35]. In [36], it has been shown that location of these sensors could influence the result of experiments.

Magnetic resonance imaging (MRI), is another non-invasie method used to get information about muscle activity. Transverse relaxation time (T2), is a reference for how quickly the transverse component of magnetization decays, and therefore a change in the resultant image. Training or physical exercise change the amount and distribution of water within muscle, therefore changes in T2. MRI signal intensity changes with respect to increases in the relaxation time of tissue water can be used to measure activity of muscles [37] [38]. Compared to other methods, MRI can access deeper muscles. Cross-talk issue observed in EMG sensors is not present with MRI approach. On the other hand, MRI has some other drawbacks compared to other approaches. MRI technique can only quantify the amplitude of activation but not timing.

There are also attempts to build models that can output muscle load given inputs of movement and other necessary conditions. Those models rely on inverse-biomechanics, forward-biomechanics, machine learning model and combination of them. Inverse-biomechanics needs insole pressure input, orientation and movement of body as shown in Figure 2.2. Forward-biomechanics begins with neural command and follows muscle activation mechanics to calculate muscle forces and joint moments [4].

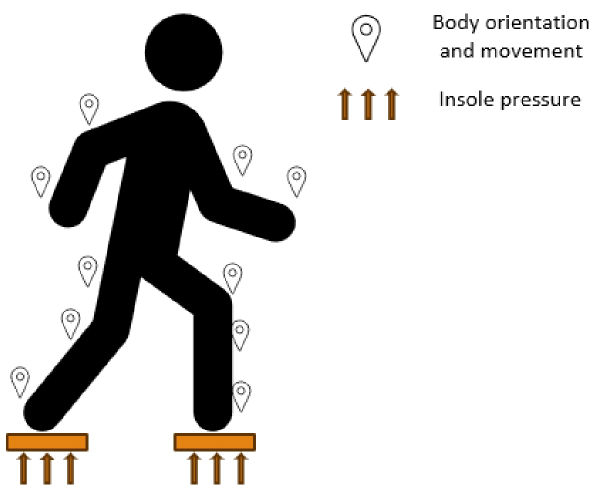

Figure 2.2. Necessary inputs for inverse-biomechanics modeling.

The flow of inverse dynamics and forward dynamics approaches are given in Figure 2.3. Machine learning models do not explicitly include any dynamics models.

Machine learning models may work with less inputs; however they are more difficult to interpret.

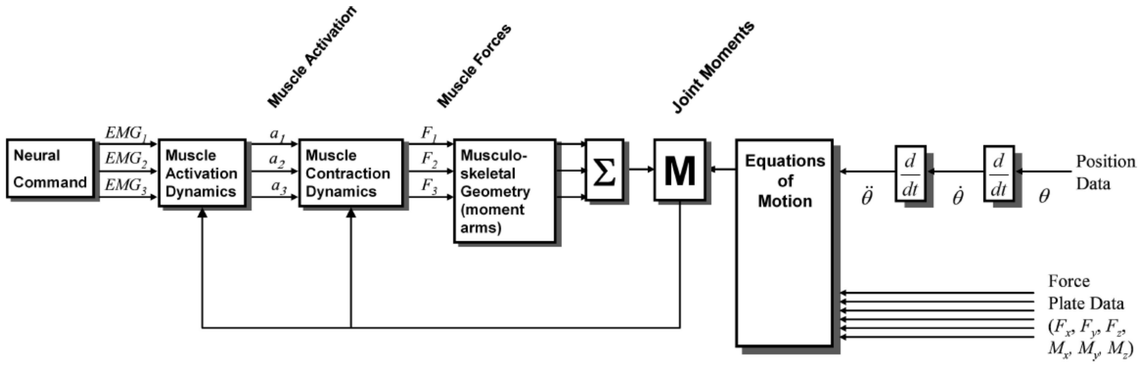

Figure 2.3. Flow of inverse-biomechanics and forward-biomechanics [4].

In [39] and [40], force on tibia is modeled using pressure insoles, accelerometers, and electromyography sensors using inverse dynamics. In [41], EMG signals are filtered with a low-pass filter and then used to predict knee moments which are calculated through inverse dynamics. In [42], it is shown how to predict joint moments by using EMG signals with a forward dynamics approach. In [4], the study uses both forward and inverse dynamics. EMG signals are used as inputs to model which predicts muscle loads and joint forces. Forward dynamics calculates motion of a system given forces and moments acting of system. Inverse dynamics calculates the necessary forces and moments given motion of the system.

In particular for running and walking, analysis could be made with different approaches. Motion analysis, dynamic electromyography, force plate recordings, energy cost measurements or energetics, and measurement of stride characteristics are among these approaches. Motion analysis refers to getting orientation and dynamics of body segment through markers and image processing or IMUs. Dynamic electromyography uses EMG sensors to understand muscle activity timings and amplitudes. Force plate recordings are used to denote peak loading, rate of loading, and vector analysis of foot plane. Energy cost measurement have been used to determine pathologic patterns that are related to neuromuscular diseases [43]. Stride characteristic refers to some parameters of gait such as cadence, speed, step length, and stride length [44]. In this study, EMG signal is predicted with a machine learning model using pressure insoles and IMU. In Figure 2.4, the difference between this study and other studies in terms of used sensor type is presented. Key difference is in terms of models used, where biomechanics models are utilized in [39] and [40], and this thesis studies a machine learning model.

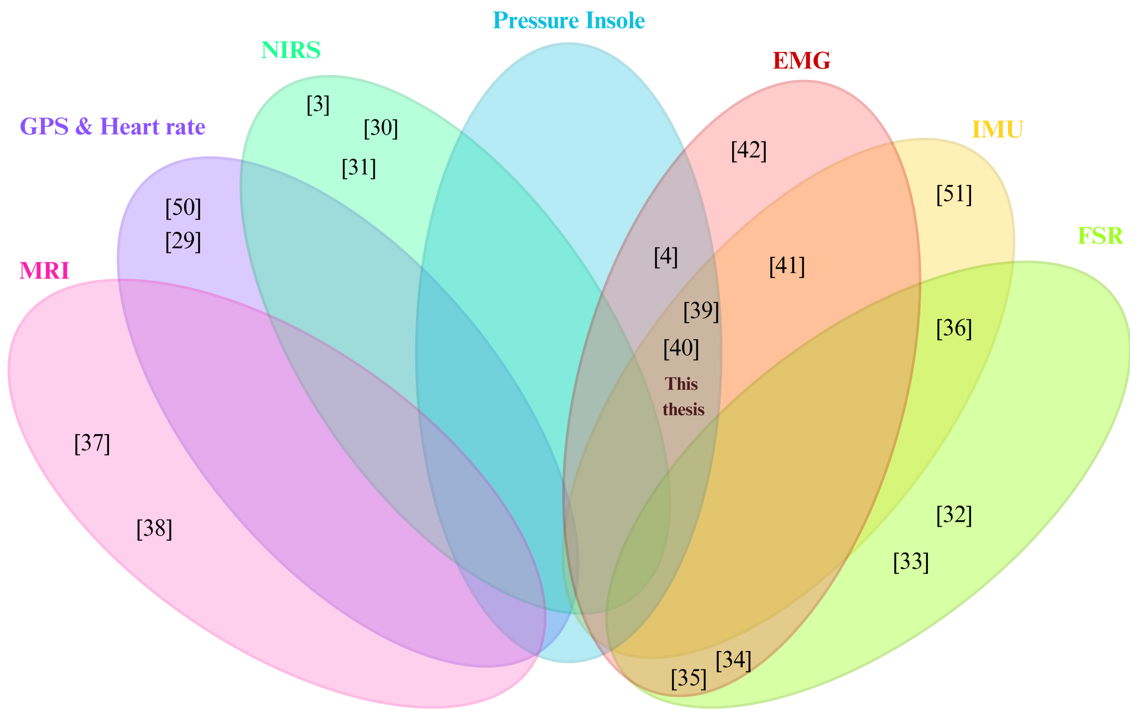

Figure 2.4. Lotus diagram to illustrate overall view of the sensor types used in different studies and this study.

2.2. Modern Applications

EMG is used to quantify loads in muscles and joints by integrating EMG outputs into musculoskeletal modes. EMG sensors are used in various fields such as prosthetics and exoskeleton control, sports analytics, rehabilitation and clinical assessment, and ergonomics.

Muscle load predictions and EMG signals are used in prosthetics and exoskeleton control. In [45], efforts to get information on central control of limb movements by using EMG signals together with limitation of those approaches are reviewed. Particular limitation is cross-talk between muscles. In [46], an EMG based neural network control method for upper-limb power-assist exoskeleton robot given intended user motion is proposed. A comprehensive review of EMG signal integration to prosthetic control is given in [47]. EMG signal quality and stability are highlighted as current challenges. Direct control, model based control, and machine learning based control are listed as current methods. Direct control utilizes EMG signals directly to activate muscle and knee joint actuators. Studies on direct control are evaluated by comparing distances, angles of target movement and realized movement. Model-based control uses biomechanical models of body segments and muscles to estimate joint movements or torques from EMG signals. Machine learning approaches use EMG signals to build gait phases, movement types, and joint actions like knee or ankle movement. These methods do not utilize biomechanical models but instead learn patterns from training data through signal processing, feature extraction, classification or regression.

One reason to study movement is to improve analytics in sports. The importance of close monitoring of physical status of athletes increases every day to get insight about game and injury risk of players. Studies have shown intensive training could lead to exhaustion and injury [48]. Real-time monitoring of muscle load requires high number of sensors which leads to excessive cost. Electromyography is a method to measure muscle activity and to assess muscle load. Also, it is possible evaluate the performance of an athlete by monitoring the muscle actions. Muscle activation amplitude and frequency give hint about the possibility to improve techniques of an athlete and analysis for injury risk. Action of athletes may require specific peak in certain muscles, or coordination of muscles groups. By recording the activities of muscles of an athlete, sports analysts may have an idea about performance of an athlete [49]. There have been numerous systems that monitors movement and health condition of players. Electronic performance tracking system, EPTS, approved by FIFA, tracks player movement and heart rate [50] . Inertial Measurement Units, IMU, are used to track quality of the shots made by tennis players. Combination of IMU sensors are also used to identify performance metrics in swimming, running and race-walking [51]. The market size of sports analytics is expected to rise to $8.6 billion in 2028 from $2.5 billion in 2021 [52].

Combination of EMG and IMU sensors are also used in rehabilitation and clinical assessment. In [53], it is concluded that inertial sensors can employed to classify rehabilitation exercise performance by developing a machine learning model which takes input from three inertial sensors. Studies with wearable technology involvement in addressing rehabilitation success are mentioned in [54], mostly focusing on accelerometers alone, or combination of accelerometers and EMG. Monitoring of functional activities in the event of strokes are also also studied with accelerometers and EMG during daily activities in [55], [56].

Ergonomics is another field where EMG and accelerometers are used for analysis. Ergonomics is the discipline that researches quality of interactions of humans with working environment. EMG sensors are utilized with IMU to assess fatigue, muscle activation and coordination for different works. In [57], fatigue due to load lifting is studied by utilizing sEMG sensors. Pain could be a result of repetitive and long working hours. Influence of pain on muscle activation is studied in [58], where pain is induced by a special chemical injection into muscle and sEMG signals are recorded for before and after injection.

A reliable EMG model, the goal of this study, would replace an actual EMG sensor and therefore it would decrease the cost of the applications mentioned before and make it more widely available to adjust for daily life. The EMG sensor is also sensitive to noise. The EMG sensor needs physical contact to skin, and a DC-offset can be present in signal produced due to high susceptibility to displacement on the skin to which it is applied. Sensors also needs a power source, which is a limiting factor for its long and continuous measurement ability. An EMG model would overcome the mentioned disadvantages.

3. THEORETICAL BACKGROUND

3.1. Biomechanics

3.1.1. Human Movement Mechanics

There are two methods to describe motion. One method to get information about human movement is to analyze kinematics of the motion. Kinematics tries to explain and analyze motion by the movement itself, and not interested in forces or torques that cause motion. Another methods is kinetics, which deals with forces and torques, that is to say cause of movement to explain motion.

A comprehensive kinematics analysis of the human motion is complex. During a single stride, analysis of lower limbs may require up to 50 variables. Specific analysis may require lower number of variables. One can calculate forces and torques built in joints and muscles if full kinematic variables and external forces are available. This approach is called inverse solution [59].

Absolute spatial reference system, shown in 3.1, is useful to denote direction of movements in gait analysis. However, it is also possible use relative motions of limbs. Terms proximal, flexion and anterior are used to describe relative motion of one limb to another in anatomical literature [59]. IMU sensor acceleration output a relative acceleration values as it also moves with the limb on which it is attached. One can derive absolute acceleration values by using compass output of the IMU sensor. In this thesis, raw acceleration outputs are used.

Forward solution is another type of technique used to analyze motion. Forward solution requires forces and reactions built on muscle and a full model of the body to get resultant motion given initial positions and velocities of body segments. This approach yields important potential to answer theoretical questions such as "What would have happened if tendon had undergone a specific treatment?". Answers to such theoretical questions would explore opportunities to solve physiological issues, e.g. spasticity, or performance improvements in sports [59].

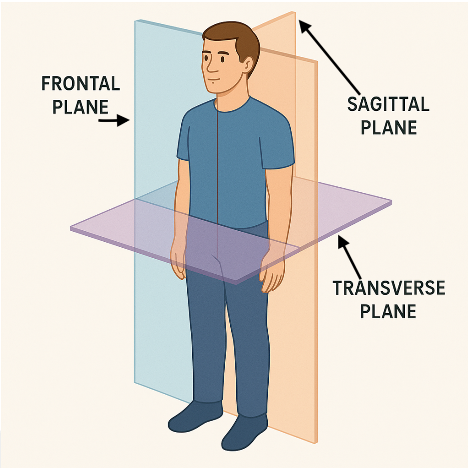

Figure 3.1. Frontal, sagittal and transverse planes are shown [5] (generated by ChatGPT, OpenAI).

3.1.2. Processing of the EMG

EMG signals need processing to be meaningful to represent muscle tension. Raw EMG signal fluctuates around zero and it is highly frequent as can be seen in Figure 3.2. There are different types of steps during EMG signal processing.

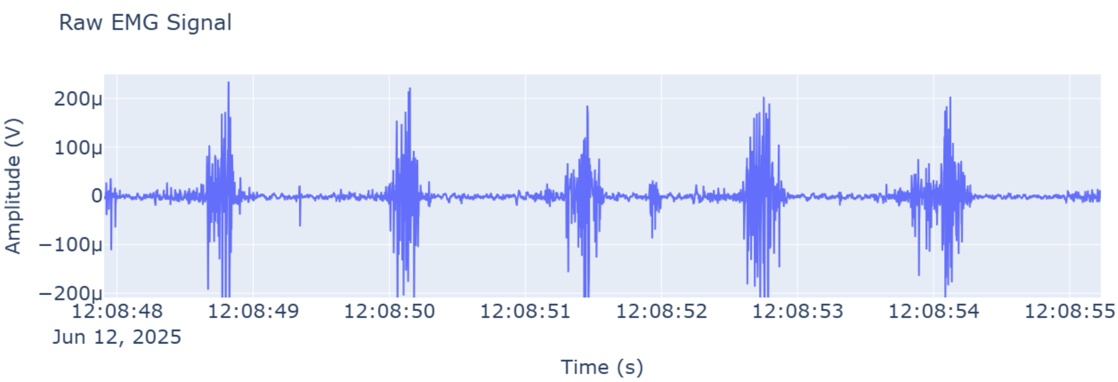

Figure 3.2. A raw sEMG signal sample taken from forearm. Signal fluctuates around zero.

- Half-wave or full-wave rectification.

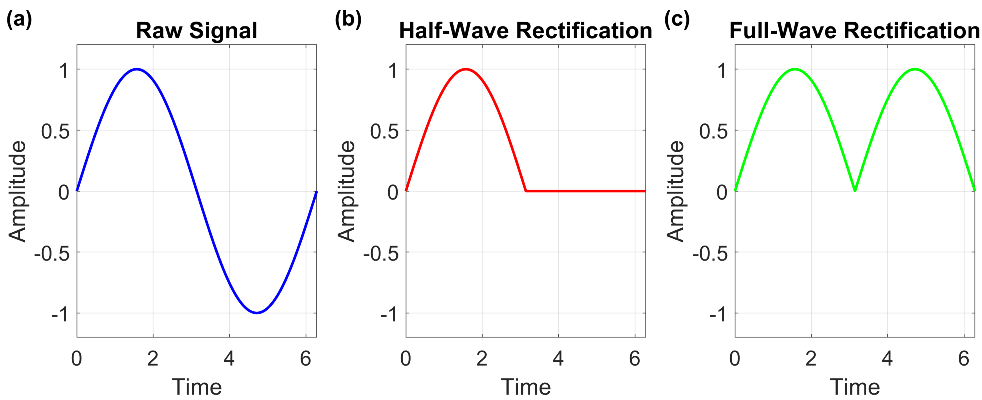

Figure 3.3. An example of (a) raw signal and processed outputs by (b) half-wave and (c) full-wave rectification.

- Linear envelope.
- Integration of rectified signal over during muscle contraction
- Integration of rectified signal for a specific time, resetting to zero once integration ends.
- Integration of rectified signal up to a certain value, resetting to zero once the value is reached.

Half-wave rectification refers to ignoring negative part of the signal, and full-wave rectification refers to generation of absolute values of the signal. The raw EMG signal has a mean value of 0. If the signal includes some offset, then DC-offset removal should be applied. Full-wave rectified signal has a mean value that depends on the extent of muscle contraction. Linear envelope processing is the output of a low-pass filter given a full-wave rectified signal. The cut-off frequency should be selected between 3-6 Hz and the filter should be second order type to represent muscle tension behavior. Integration is also applied after a full-wave rectification of the signal. Integrators provide extent and frequency of muscle contractions [59].

This thesis studies the prediction of processed EMG signals. The methods to process EMG signals used in this thesis are DC-offset removal, full wave rectification, and envelope creation in the respective order. The cut-off frequency value is a crucial point for trade-off between capturing gait cycle behavior and best prediction performance.

3.2. Machine Learning

Machine learning is a wide range of collection of techniques to predict classes or continuous values of desired behavior. The type of problems where the goal is predict a set of class values is called classification problem. Another type of problem is the prediction of continuous values, such as time-series data, is called regression [60]. The problem dealt in this thesis is to predict EMG values, which are continuous values. Therefore, models suitable for regression are tested in this thesis. Another method to divide machine learning is to seperate them based on feedback coming from data. Supervised learning is type of learning using labeled data, unsupervised learning utilizes unlabeled data and reinforcement learning interacts with data by reward and penalty. In supervised learning, linear regression, logistic regression, support vector machines (SVMs), decision trees, and random forests are used to model target value using inputs. K-means clustering, principal component analysis (PCA), and hierarchical clustering, reveals patterns in unlabeled data, therefore they are used in unsupervised learning. Reinforcement learning model examples are Q-learning and policy gradient methods, which trains model by interacting with environment, e.g. simulation, and rewarding or penalizing model. A deep neural network is a class of machine learning models. Deep neural networks can be used for both classification and regression tasks. Training of these models is called deep learning [61]. Developing a machine learning model includes steps such as data collection, data processing, model training and model validation. Typical machine learning workflow is given in Figure 3.4.

Inputs of a machine learning model are called features. Feature engineering refers to selection of the correct inputs, deriving and processing them to have a better representation of the target value. Feature engineering is a critical process, since a poorly managed feature engineering phase may direct a research into wrong direction especially regarding model selection. A model that would be sufficient for a problem would be considered insufficient if features are not selected and engineered correctly [62]. There are various methods to process raw features and create new features. Some methods to process and create new features are statistical transformations. The notable ones are creating polynomial features; multiplying each features with each other, logarith-

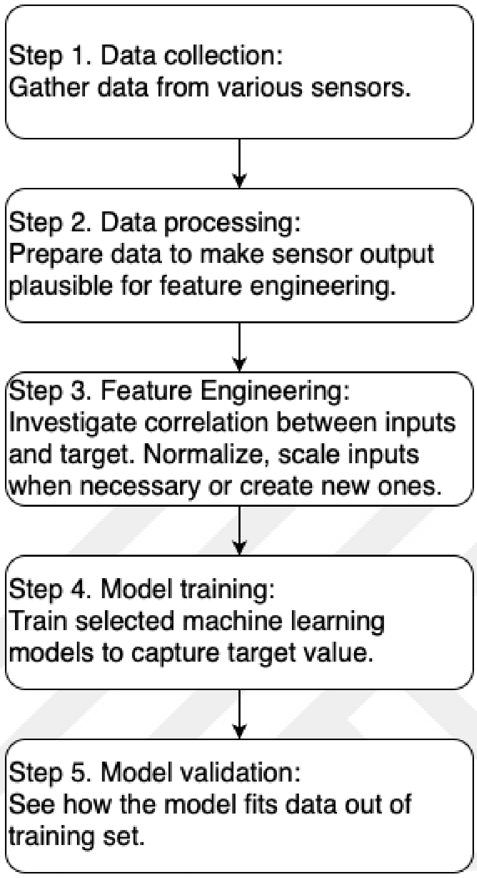

Figure 3.4. Steps for building a machine learning model.

mic transformation, standardization or scaling of data. Those methods are useful to capture non-linear relationship and helpful for distance based or gradient based algorithms. Another set of methods are arithmetic combinations, basically subtraction, addition, divison or multiplication of features. Interesting method in arithmetic combination family is rolling function of mean calculation, standard deviation calculation, or minimum-maximum calculation over a window. A powerful method to predict time-series data, particularly when history of the signals is very influential, is time-series manipulation, which includes lagging features and taking derivatives and integrating features over a period. For non-numerical data, encoding methods can be applied. One-hot encoding, ordinal encoding, target encoding, frequency encoding are among encoding methods.

There are two phenomenon that an engineer should be aware of. One of them is called overfitting, which is observed when model predicts training set, observed data, very well but does poor on test set, unobserved data. The reasons of overfitting could be using few data, or picking too complex models. Simplifying model, adding more data, regularization or feature elimination can solve overfitting issue. The other phenomenon is underfitting, which means model performs too bad both on training data and test data. The reason could be using too simple model, low training time or insufficient features to capture target value trends. Using more complex models, adding or deriving more features, longer training time, and removing or decreasing regularization can solve underfitting issue. Regularization is a method to decrease complexity of the model by adding a penalty term in cost function that punishes high number and high value coefficients of the model [60].

Feature engineering is an important part of this thesis, since the raw signals include limited temporal information. Creation and processing of features are necessary to capture temporal values. This thesis uses several complex models to predict processed EMG signal. The dataset size and complexity of the models suggest that a regularization term in the cost function is beneficial to decrease overfitting extent and increase accuracy of the model on the test set.

4. EXPERIMENTAL SETUP AND RESULTS

4.1. Data Collection

In the experiment, a person has been instrumented with two insole pressure sensors beneath his feet and a combined EMG/IMU sensor on the calf. The person has repeated 4 cycles of walking, each cycle lasting for two minutes. Insole presure sensor is able to measure the pressure distribution and dynamic changes of the insole. Each insole pressure sensor is consisted of 12 force sensing resistor cells, which can collect the force applied positions and pressure changes information. Location of each force sensing resistor cells on the insole pressure sensor can be seen in Figure 4.1. In Figure 4.1, the insole pressure sensor can be seen. Specification of the insole pressure sensor is provided in Table 4.1. The sensor characteristic curve (resistance vs. mass) of a force sensing resistance is shown in Figure 4.2.

Table 4.1. Insole Pressure Sensor Specifications.

Item	**Data**
Sensor size (single point sensor)	20.5×11mm
Sensor quantity	12
Force measure range (single point sensor)	50G∼20KG
Force measure range (1 piece insole sensor)	10KG∼100KG
FSR Thickness	<0.3mm
Force repeatable (part to part)	±10%
Off resistance	>2 MΩ
Force resolution	Continuous
Response time	<1ms
Operating temperature	-20°C∼60°C

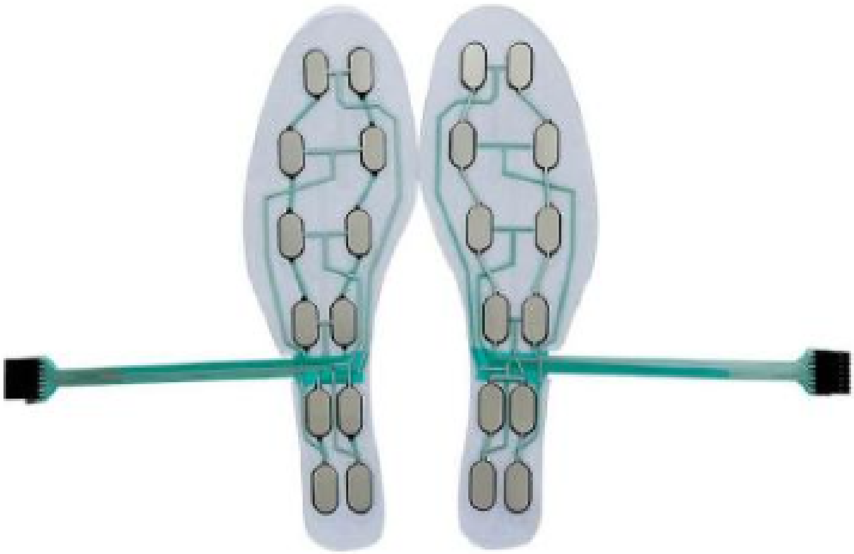

Figure 4.1. Insole pressure sensors used in experiment.

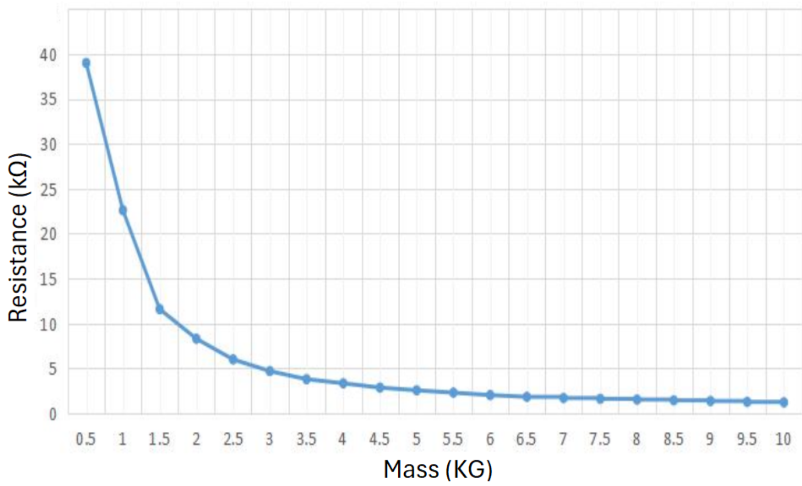

Figure 4.2. Resistance decreases as the mass/force applied on an FSR increases.

Trigno Avanti sensors are used for EMG/IMU measurements. The sensor has a built-in nine DOF inertial measurement unit together with an onboard EMG Sensor. Sensor placement is made as shown in Figure 4.3. Before the placement of EMG, the skin should be cleaned to remove oils and dead skin. Shaving may also be required to improve signal quality.

Two different devices are used to collect data from sensors. EMG and IMU sensor data are collected through DELSYS software. Insole pressure sensor transmits data

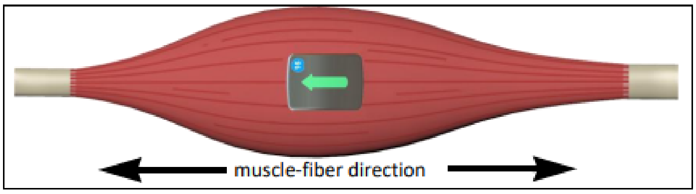

Figure 4.3. EMG sensors should be placed to align muscle fibers with direction of electrodes.

through bluetooth. Insole pressure data is collected with bleak Python library [63]. Synchronization of data is achieved by logging time-stamps of each data sample. Snips from data is given in Appendix B. The longest time interval common for both data sources is used. Then resampling is done by aligning time-stamps with hold last value technique. Interpolation is not used, since EMG data would be harmed by interpolation as EMG signals fluctuates around zero frequently. Resampling is done at 20 Hz.

An adhesive tape is applied to secure the EMG sensor to the skin and to prevent it from falling off during walking. The tape brought some noise onto EMG sensor. Therefore, the first step of pre-process of EMG data is DC-offset removal to ensure zero mean signal. [64]. DC-offset removal ensures signal to fluctuate around zero, as expected for EMG data as shown in Figure 4.4. Rectification is done by

$$x_{\text{zero\_mean}}(t) = x(t) - \mu \tag{4.1}$$

where the mean μ is computed

$$\mu = \frac{1}{N}\sum_{i=1}^{N} x(i). \tag{4.2}$$

After the DC-offset removal, zero-mean is signal rectified by

$$x_{\text{rectified}}(t) = |x_{\text{zero\_mean}}(t)| \tag{4.3}$$

and shown in Figure 4.5. To extract the envelope of the rectified EMG signal, a second-order Butterworth low-pass filter is applied. The sampling rate of the signal, f_s, is 20 Hz, and the cutoff frequency, f_c, for the filter is set to 4 Hz. First, the Nyquist frequency is computed as

$$f_N = \frac{f_s}{2} \tag{4.4}$$

then the normalized cutoff frequency is calculated as

$$\omega_c = \frac{f_c}{f_N}. \tag{4.5}$$

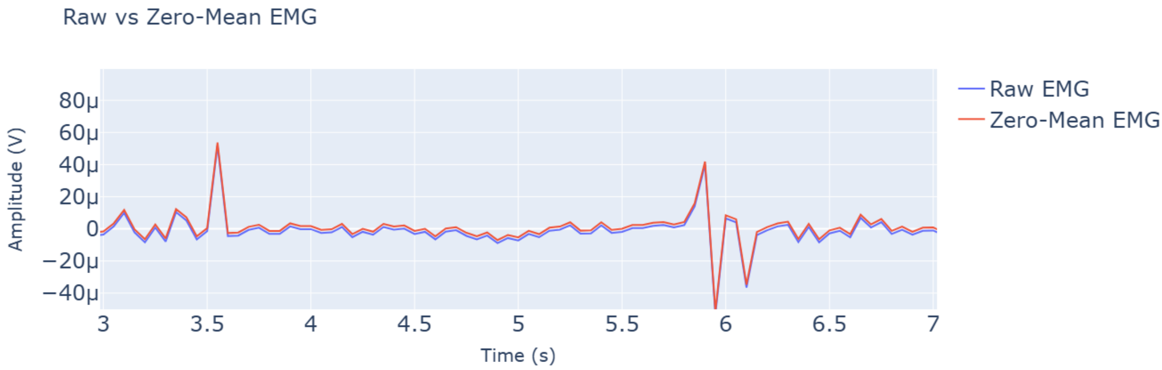

Figure 4.4. Raw EMG signal fluctates around a negative value. After DC-offset removal the signal fluctuates around zero.

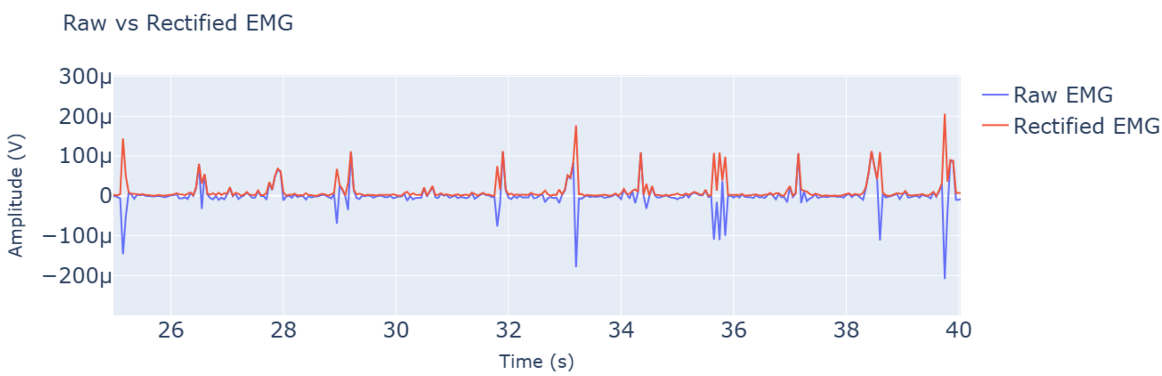

Figure 4.5. DC-offset removed signal is rectified.

The filter coefficients are obtained using

$$b, a = \text{butter}(2, \omega_c, \text{btype='low'}) \tag{4.6}$$

and the envelope signal is computed through zero-phase filtering as

$$x_{\text{envelope}}(t) = \text{filtfilt}(b, a, x_{\text{rectified}}(t)). \tag{4.7}$$

The purpose of the filtering is to smoothen the EMG signal, which fluctuates rapidly. The filtering is done using the filtfilt method to avoid phase distortion by applying the filter forward and backward. This process allows us to capture the low-frequency envelope of the EMG signal without shifting the signal in time, as seen in Figure 4.6.

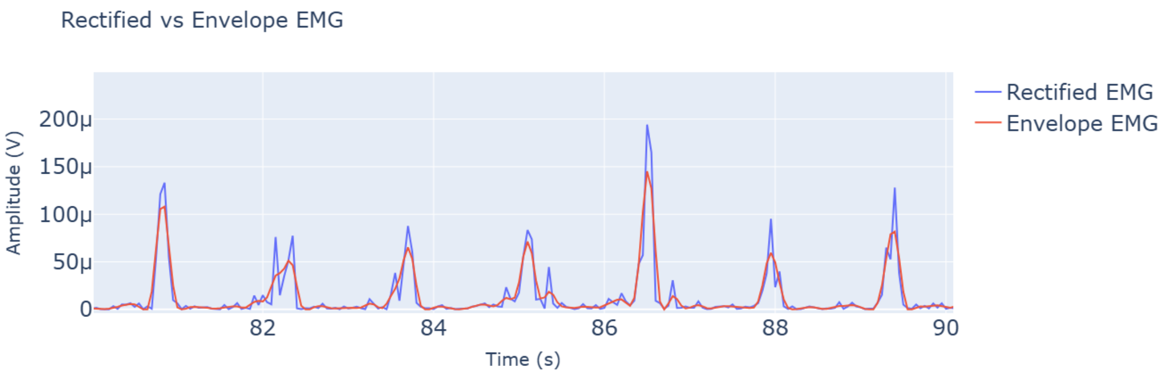

Figure 4.6. Rectified signal is passed through low-pass filter and EMG envelope is obtained.

4.2. Modeling

Modeling phase starts with feature engineering. The first step is visual exploration of data. Target signal, envelope of the EMG sensor attached to right calf, is plotted together with insole pressure sensors in Figure 4.7. In this plot, it is seen that a peak in the EMG occurs when right insole pressure values starts to increase from lowest levels, at the same time left insole pressure values decreases. This behavior suggests that processing insole pressure sensor data to create new feature to capture temporal patterns is necessary.

The first trained model is XGBoost regressor (XGBRegressor) only with a single cycle to save time for initial model exploration. Training time may grow rapidly as the sample number increases. XGBoost is an ensemble learning method based on gradient-boosted decision trees. The model was configured with 100 estimators, a maximum tree depth of 4, and a learning rate of 0.1. Target value, EMG envelope, is converted to milivolt unit to avoid training phase stop prematurely due low error accompanied by low accuracy. 80% of data is used for training and rest 20% is reserved as test set throughout the study. Initial features are selected as raw insole pressure sensor signals, which provided low accuracy. Then the feature set is expanded to capture temporal patterns. Lagged features, the first and the second order derivates, and the mean and standard deviation over a 1-second moving window are created as new

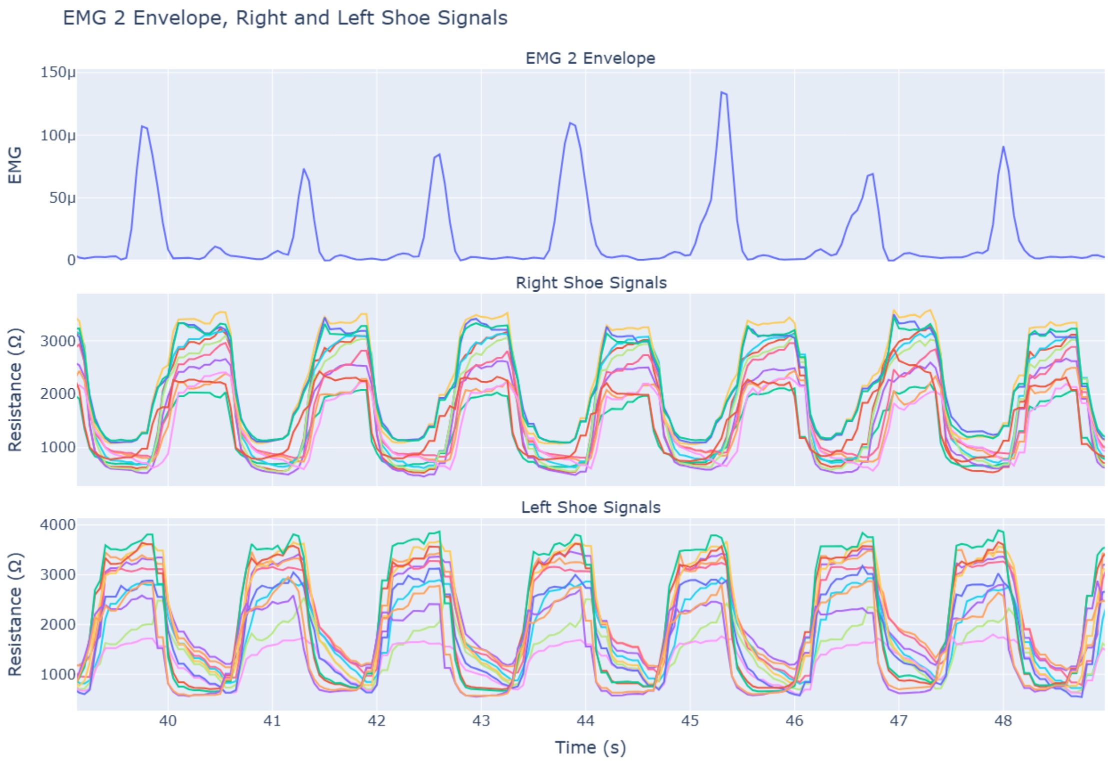

Figure 4.7. DC-offset removed signal is rectified.

features. New feature set provided a better accuracy. Also a grid search is conducted for hyperparameter tuning, tree count, depth, learning rate, and sampling ratios are swept. All combinations using cross-validation is performed and the best set based on chosen scoring metric is selected as best model through cross-validation. The results of the iterative phase XGBoost model attempts have been given in Table 4.2. Accuracy of the XGBoost model does not increase significantly even after hyperparameter, which suggests improving the feature set, increasing samples for training phase or testing new machine learning models.

Table 4.2. Performance metrics for models predicting EMG envelope.

Model Type	**Feature Set**	**R^2**	**RMSE (μV)**
XGBoost	Raw insole pressure signals	0.3119	21.34
XGBoost	Time-windowed features + derivatives	0.3974	20.02
XGBoost	Time-windowed features + derivatives (best hyperparameters)	0.4069	19.86

It is intuitive to work further on feature set since the previous model utilized only 2 sensors, insole pressure sensors for both feet. Acceleration and angular velocity signals on all three axis are filtered using the same filter to create EMG envelope, and then they are inserted into feature set. Similar to first phase, selection of features begins with initial visual reasoning. EMG envelope is plotted with acceleration and angular velocity signals in Figure 4.8. It is seen that peak of the EMG envelope is related with increasing phase of angular velocity in Z-axis. It is also clear that cyclic behavior is present in each gait cycle and repetitive. Therefore, the acceleration and gyroscope signals are added to feature set.

The addition of acceleration and gyroscope signals have been useful to improve accuracy. Initially, the feature set is not processed and fed into training phase. Then in order to capture temporal behavior, same feature creation processes, creation of lagged features, derivatives and window parameters, in previous step are repeated. The results of the addition of acceleration and angular velocity values have been presented in Table 4.3.

In order to check if there is an overfitting issue, the accuracy is tested on training data with expanded feature set with acceleration and angular velocity data. The result is presented in Table 4.4.

Table 4.3. Performance metrics for models predicting EMG envelope after acceleration and gyroscope signals are included in feature set.

Model Type	**Feature Set**	**R^2**	**RMSE (µV)**
XGBoost	Raw insole pressure signals (filtered acceleration and angular velocity)	0.5224	17.77
XGBoost	Raw insole pressure signals filtered acceleration and angular velocity + time windowed features + derivatives	0.5693	16.93

Figure 4.8. Trend of EMG envelope is in relation with the trend of acceleration and gyroscope signals.

Table 4.4. Performance metrics for models predicting EMG envelope after acceleration and gyroscope signals are included in feature set on training set.

Model Type	**Feature Set**	**R^2**	**RMSE (µV)**
XGBoost	Raw insole pressure signals (filtered acceleration and angular velocity)	0.8999	7.66
XGBoost	Raw insole pressure signals filtered acceleration and angular velocity + time windowed features + derivatives	0.9980	1.09

It is clear that overfitting is present. The reason for the overfit may be that we have a too complex model. In order to make a slight improvement initial model maximum depth is lowered. The influence of change in maximum depth is given in Table 4.5.

Table 4.5. Comparison of less complex models where complexity is changed by lowering maximum depth of the model.

Model Type	**Max depth**	**R^2**	**RMSE (µV)**
XGBoost	6	0.5693	16.93
XGBoost	5	0.5778	16.76
XGBoost	4	0.5979	16.36
XGBoost	3	0.6090	16.13
XGBoost	2	0.5885	16.54

There is still low accuracy on test set which shows overfitting is still present. In order to elimate overfitting issue and improve accuracy test set, regularization is applied. L1 and L2 regularization is applied here and the results have been provided in Table 4.6. Mathematical foundation of this regularization is provided in Appendix D.

The best R^2 value is 0.6314, which tells the model is still suffers from overfitting issue. Applying regularization and decreasing model complexity still leave overfitting issue behind. One reason of overfitting could be the noise in the target value. One more iteration is done, where the target envelope created by picking the cutoff frequency, f_c as 3 Hz. instead of 4 Hz. Also since the filtering effect is more dominant now, the lagging feature window is shortened. This changes have provided the best result with preserving maximum depth of the model to 6, regularization parameter L1 is set to 0.5 and regularization parameter L2 is set to 0.1. Results have of those changes is provided in Table 4.7. Accuracy of corresponding to this R^2 value is plotted in Figure 4.9

It is useful to test different models to check if there is a better candidate to model EMG signals. Support Vector Regressor (SVR), Random Forest Regressor (RFG) and Multi-layer Perceptron (MLP) models have been tested to check if there is a room for improvement by utilizing different models.

Table 4.6. Performance metrics for XGBoost with different L1 and L2 regularization values.

Model Type	**Hyperparameters**	$\mathbf{R^2}$	**RMSE (µV)**
XGBoost	reg_alpha=0.1, reg_lambda=0.1	0.6244	15.81
XGBoost	reg_alpha=0.1, reg_lambda=0.5	0.6256	15.78
XGBoost	reg_alpha=0.1, reg_lambda=1	0.6091	16.13
XGBoost	reg_alpha=0.1, reg_lambda=2	0.5823	16.67
XGBoost	reg_alpha=0.5, reg_lambda=0.1	0.6272	15.75
XGBoost	reg_alpha=0.5, reg_lambda=0.5	0.6314	15.66
XGBoost	reg_alpha=0.5, reg_lambda=1	0.6057	16.19
XGBoost	reg_alpha=0.5, reg_lambda=2	0.5723	16.87
XGBoost	reg_alpha=1, reg_lambda=0.1	0.6238	15.82
XGBoost	reg_alpha=1, reg_lambda=0.5	0.6293	15.70
XGBoost	reg_alpha=1, reg_lambda=1	0.6208	15.88
XGBoost	reg_alpha=1, reg_lambda=2	0.5681	16.95
XGBoost	reg_alpha=2, reg_lambda=0.1	0.6264	15.76
XGBoost	reg_alpha=2, reg_lambda=0.5	0.6154	16.00
XGBoost	reg_alpha=2, reg_lambda=1	0.6030	16.25
XGBoost	reg_alpha=2, reg_lambda=2	0.5906	16.50

Table 4.7. Performance metrics when cut-off frequency is set to 3 Hz.

Model Type	**Lag Window**	**Maximum Depth**	$\mathbf{R^2}$	**RMSE (µV)**
XGBoost	10 sample (0.5 seconds)	6	0.6952	13.11

Support vector regressor model is tested for different hyperparameter by changing epsilon, error insensitive range, and regularization coefficient. It is seen that the epsilon value of 1 or smaller performs similar, while regularization term has significant effect. In support vector regression, as regularization term increases, strength of

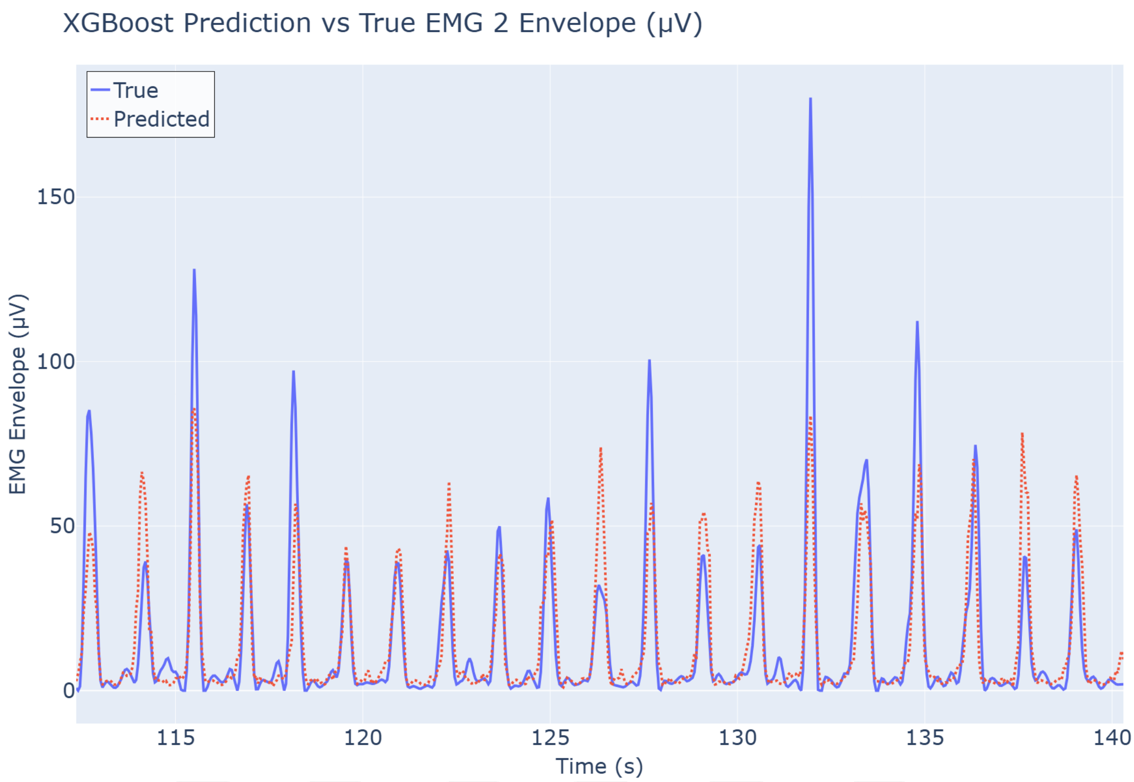

Figure 4.9. Predicted envelope vs. actual envelope.

regularization decreases. Hyperparameter tuning results have been provided in Table 4.8. As the regularization effect is increased, model complexity increases and therefore training accuracy always increases. However, there is an optimal intermediate level of regularization which strikes a balance between test and training accuracy. Therefore increasing regularization coefficient, decreasing strength of regularization, increases accuracy on both sets. Overfitting issue is also observed for support vector regression model.

Random forest regressor model is another model type tested for the task. Similar to XGBoost model, random forest is also a tree-based model, trains many trees. Number of estimators, maximum mepth, minimum split and minimum leaf number parameters are swept and results are provided in Table 4.9. Overfitting issue is still present with random forest models.

Another set of models trained to model EMG signal is multi-layer perceptron model. Number of hidden layer units, L2 regularization term -α-, and learning rate

Table 4.8. Performances of SVR with different settings.

Regularization Coefficient	**Epsilon**	**R^2 Test**	**R^2 Training**
1	1	0.3939	0.3958
1	0.01	0.3934	0.3955
10	0.01	0.5417	0.5766
100	0.01	0.6151	0.7403
1000	0.01	0.5713	0.9278
2000	0.01	0.5086	0.9707
2000	0.0001	0.5086	0.9707

parameters are tested and results are presented in Table 4.10. The results show close accuracy values for both test and training sets. The extent of overfitting is decreased but the accuracy maintained at similar levels with previous model attempts , which suggests expanding data pool would be helpful.

The summary of the models trained so far and best performances of each model is provided in Table 4.11. Each of the best performing examples from different model types yields similar accuracy levels and overfitting issue.

Table 4.9. Performance of Random Forest models with different hyperparameters.

n_estimators	**max_depth**	**min_split**	**min_leaf**	**R^2 Test**	**R^2 Train**
100.0	10.0	2.0	1.0	0.6717	0.9399
200.0	15.0	5.0	2.0	0.6822	0.9466
300.0	20.0	10.0	4.0	0.6803	0.9150
500.0	30.0	5.0	1.0	0.6743	0.9518

Table 4.10. R^2 performance of MLP models with different hyperparameters.

Hidden Layers	α	**LR**	$\mathbf{R^2}$ **Test**	$\mathbf{R^2}$ **Train**
(256, 128)	1e-05	0.01	0.6854	0.8361
(64, 32)	0.001	0.001	0.6779	0.8196
(64, 32)	0.001	0.01	0.6708	0.8601
(64,)	1e-05	0.001	0.6688	0.7952
(256,)	0.001	0.001	0.6670	0.8346
(256,)	1e-05	0.001	0.6640	0.8350
(256, 128)	0.001	0.01	0.6633	0.8951
(64,)	0.001	0.001	0.6624	0.7948
(256,)	0.001	0.01	0.6617	0.8193
(256,)	1e-05	0.01	0.6615	0.8189
(128, 64)	1e-05	0.001	0.6540	0.7779
(128, 64)	0.001	0.001	0.6526	0.7789
(64,)	0.001	0.01	0.6512	0.7626
(64, 32)	1e-05	0.001	0.6505	0.7580
(128, 64)	0.001	0.01	0.6454	0.8333
(64,)	1e-05	0.01	0.6386	0.8217
(128, 64)	1e-05	0.01	0.6344	0.7712
(256, 128)	0.001	0.001	0.6341	0.8850
(256, 128)	1e-05	0.001	0.6208	0.8850
(64, 32)	1e-05	0.01	0.6088	0.8795

So far, training and test sets are split without distortion of the order of data, which means test data always correspond to last parts of data. Since our data is time ordered data, and the walking speed varies from initial phases of experiments to later stages, distribution of the data may not be the best practice for model build up. Training and test data should be independently and identically distributed from the same distribution for supervised learning [65]. In [66], it is highlighted that if the data is sorted in anyway, random separation of data as training and test set is necessary to

Table 4.11. Comparison of best-performing models and their performance metrics.

Model	R^2 Test	R^2 Train
XGBoost	0.6952	0.9960
SVR	0.6930	0.9599
Random Forest	0.6822	0.9466
MLP	0.6854	0.8361

avoid evaluation bias. There are some pitfalls to split data pool randomly. In [67], it is emphasized that shuffling during test and training data split would harm temporal dependencies. Since the feature engineering step is this study includes creation of lagged features prior to training set creation, in our case shuffling is not an issue. Random shuffling of time-series data would disable utilization of models that has temporal memory, such as recurrent neural networks. Since the models utilized in this study, XGBoost - SVR - RFR - MLP, do not consider time axis and has no memory, shuffling is applicable in this case.

The results with best forming configuration in terms of training accuracy is tested for `Shuffling=True` is tested for XGBoost regressor, SVR, RFR, and MLP models. The reason to pick the models with best training set accuracy instead of test set accuracy is to check effects of shuffling, which can solve overfitting issue to some extent on its own. Results are provided in Table 4.12. Since shuffling of training and test sets include randomness, training and test results are not same between different model trial runs given same configuration. In Table 4.12 approximate accuracy values for each model type is given.

All models result in higher test accuracies. Among the all models, XGBoost accuracy is significantly increased, after the data pool is shuffled. Since the test set data pool order is random, not time-ordered, it would be more explanatory to plot all data with target and predict value to have a clear understanding in time domain. A snip from the test, ordered in time is provided in Figure 4.10.

Table 4.12. Comparison of best-performing models and their performance metrics.

Model	**R^2 Test**	**R^2 Train**
XGBoost	0.7546	0.9962
SVR	0.6930	0.9599
Random Forest	0.7230	0.9435
MLP	0.6910	0.8620

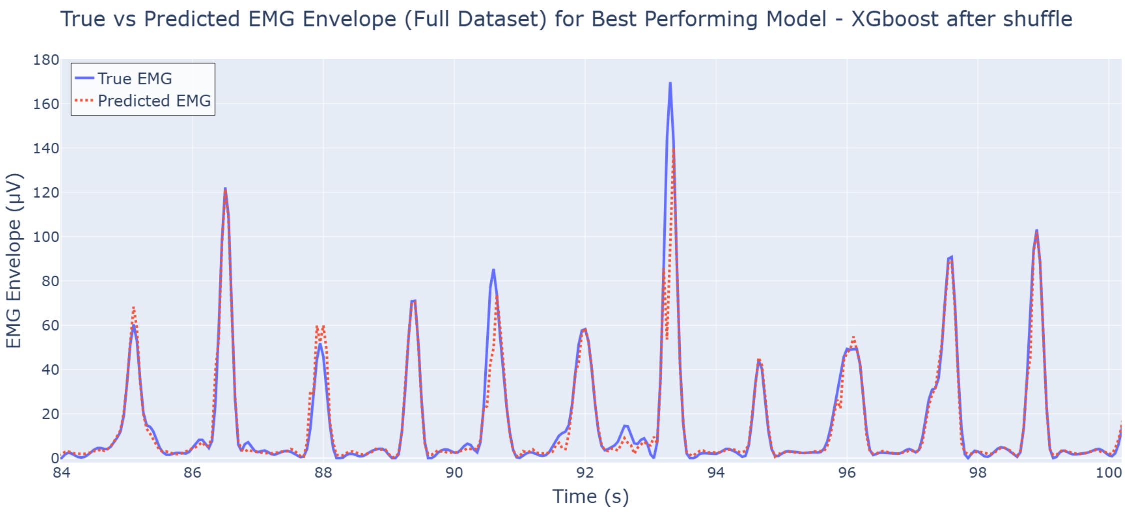

Figure 4.10. Predicted envelope vs. actual envelope for best performing XGBoost model after shuffle.

The trend of predicted EMG envelope is almost identical with real EMG envelope. Model is able to capture even slight and sudden increases in real envelope. There are cases where high peaks of EMG envelopes are slightly underestimated. It is beneficial to check if some portion of the error is specified to some certain levels of EMG values. Such analysis are useful to pinpoint possible improvement areas for the model. One way to make this analysis is to plot scatter of true and prediction values. In Figure 4.11, it is seen that underestimation tendency is dominant for high EMG values. It is also seen that the portion of total data for high EMG signal values are low, which means model is not trained with enough data for high end values.

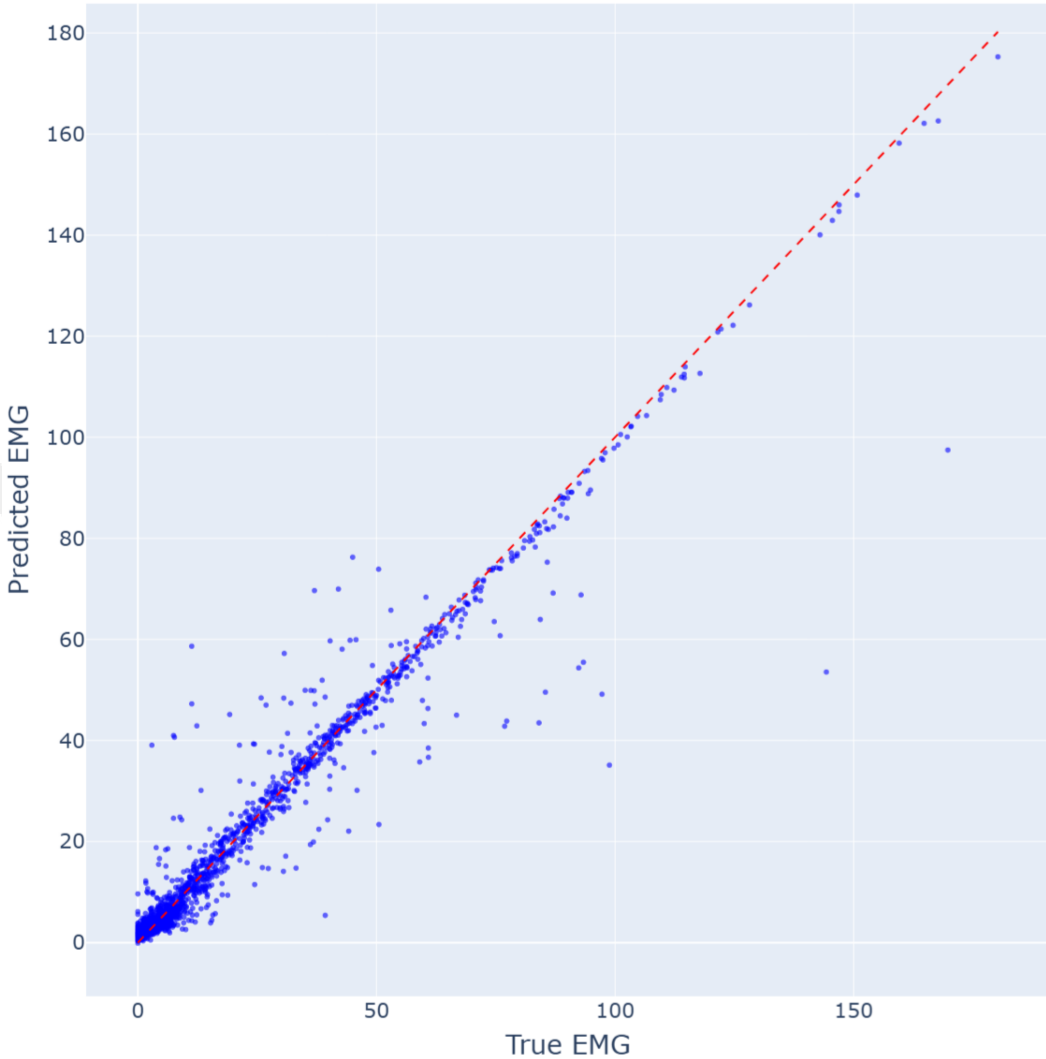

Figure 4.11. Predicted envelope vs. actual envelope for best performing XGBoost model after shuffle.

5. CONCLUSION AND FUTURE WORKS

5.1. Conclusion

There have been different attempts measure muscle load or workload intensity. There are different methods that utilize different sensors or devices to predict muscled load. Telemetric heart rate sensor combined with Global Positioning System (GPS) has been widely used in sports to track athletes and compute workload associated with movement. Near-infrared spectrophotometry is used to measure oxygen levels in the blood to relate it with workload level. Force sensing resistors and strain gauges are used to measure volumetric changes which is associated with muscle activity. Magnetic resonance imaging is used to capture change in water content in muscles to quantify level of muscle activity. Combining EMG sensors with devices that can capture the movement of athletes have been deployed to model muscle load. Cameras and inertial measurement units have been used to derive movement of the athletes. Force plates and pressure insoles also have been used in muscle load model development. Those models include combination of forward dynamics, inverse dynamics and machine learning models. Some of those models predict muscle load where some models are used to categorize the movement.

This study aims to utilize machine learning methods to model an EMG sensor attached to calf during walking with lowest number of sensors. EMG sensors directly output the activity of a muscle in terms of voltage signal which is the sum of motor unit potentials beneath the muscle on which EMG is attached. Two pressure insoles and eight EMG-IMU combined sensors are used in the experiment.

EMG sensor envelope creation is followed by feature selection and model training. It is seen that cutoff frequency used during envelope creation can be influential on model accuracy. Different feature sets suggested that insole pressure sensors could be limited to interpret the change in EMG signals. Feature is set expanded with addition of acceleration and angular velocity of the calf. Current configuration suffered from overfitting issue. Feature engineering and hyperparameter tuning have been helpful in reducing the extent of overfitting issue. Different models, XGBoost Regressor, support vector regression , multi-layer perceptron and random forest regressor, have been tested. The best result is provided with XGBoost model with moderate regularization and trained with shuffled data pool, and the average R^2 value for test set accompanied with aforementioned model is 0.75.

5.2. Future Works

This study has explored the possibility of modeling an EMG sensor. It has been seen that there is a great potential in developing a machine learning regression model that can predict EMG sensor output. Utilization of insole pressure sensors with higher resolution accompanied with higher sampling rate would be useful to improve the accuracy of the model. Increasing the resolution of the insole pressure sensors would let machine learning model to distinguish a wider range of pressure beneath the feet, which is particularly useful for models that do not need scaling. Increasing the sample rate would make higher resampling rate possible as well.

One way to overcome overfitting issue is also to expand the data pool. Walking phases should be kept longer to produce more data. In order to achieve longer walking phases, a special attention should be paid to ensure EMG sensor sticks to the muscle for the sake of experiment. Running phases can also be included in the data pool once aforementioned issue has been solved. Once a model with high accuracy is developed, it can be used to detect anomalies by comparing model output with actual EMG sensor output. Detection of anomalies would require matching same movement patterns and keep the track of the difference between the model output and the EMG output.

REFERENCES

1. Farina, D. and A. Holobar, "Characterization of Human Motor Units From Surface EMG Decomposition", *Proceedings of the IEEE*, Vol. 104, No. 2, pp. 353–373, 2016.

2. Nam, D., J. M. Cha and K. S. Park, "Next-Generation Wearable Biosensors Developed with Flexible Bio-Chips", *Micromachines*, Vol. 12, No. 1, p. 64, 2021.

3. Gersak, G., Vesna; Geršak, "NIRS: Measuring Changes in Muscle Oxygenation and the Detection of Muscle Activity", *19th IMEKO World Congress 2009*, 2009.

4. Buchanan, T. S., D. G. Lloyd, K. Manal and T. F. Besier, "Estimation of Muscle Forces and Joint Moments Using a Forward-Inverse Dynamics Model", *Medicine & Science in Sports & Exercise*, Vol. 37, No. 11, pp. 1911–1916, 2005.

5. OpenAI, "ChatGPT (June 2025 version)", `https://chat.openai.com`, 2025, figure generated using ChatGPT based on author instructions, accessed on June 3, 2025.

6. Arzani, A., J.-X. Wang, M. S. Sacks, S. C. Shadden, S. Baek and J. D. Humphrey, "Machine Learning for Cardiovascular Biomechanics Modeling: Challenges and Beyond", *Annals of Biomedical Engineering*, Vol. 50, No. 6, pp. 615–627, 2022.

7. Halilaj, E., A. Rajagopal, M. Fiterau, J. L. Hicks, T. J. Hastie and S. L. Delp, "Machine Learning in Human Movement Biomechanics: Best Practices, Common Pitfalls, and New Opportunities", *Journal of Biomechanics*, Vol. 81, No. 1, pp. 1–11, 2018.

8. Andriacchi, T. P. and E. J. Alexander, "Studies of Human Locomotion: Past, Present and Future", *Journal of Biomechanics*, Vol. 33, No. 10, pp. 1217–1224, 2000.

9. Borelli, G. A., *De Motu Animalium*, Rome, 1680.

10. Boerhaave, H., "De usu ratiocinii mechanici in medicina: oratio habita in auditorio magno", *Boerhaave's Orations*, Brill Academic Publishing, Leiden, 1983.

11. Baker, R., "The History of Gait Analysis before the Advent of Modern Computers", *Gait Posture*, Vol. 26, No. 3, pp. 331–42, 2007.

12. Inman, V. T., H. J. Ralston and F. Todd, *Human Walking*, Williams & Wilkins, Baltimore, 1981, includes illustrations and index.

13. Sutherland, D. H., "The Evolution of Clinical Gait Analysis Part I: Kinesiological EMG", *Gait & Posture*, Vol. 14, No. 1, pp. 61–70, 2001.

14. Sutherland, D., "The Evolution of Clinical Gait Analysis: Part II Kinematics", *Gait & Posture*, Vol. 16, No. 2, pp. 159–179, 2002.

15. Sutherland, D. H., "The Evolution of Clinical Gait Analysis Part III – Kinetics and Energy Assessment", *Gait & Posture*, Vol. 21, No. 4, pp. 447–461, 2005.

16. Rash, G. and Edd, "Electromyography Fundamentals", *International Encyclopedia of Ergonomics and Human Factors*, 2002.

17. Amorim, C. and R. Marson, *Application of Surface Electromyography in the Dynamics of Human Movement*, 2012.

18. Reaz, M. B. I., M. S. Hussain and F. Mohd-Yasin, "Techniques of EMG Signal Analysis: Detection, Processing, Classification and Applications", *Biological Procedures Online*, Vol. 8, No. 1, pp. 11–35, 2006.

19. Al-Ayyad, M., H. Owida, R. De fazio, B. Al-Naami and P. Visconti, "Electromyography Monitoring Systems in Rehabilitation: A Review of Clinical Applications, Wearable Devices and Signal Acquisition Methodologies", *Electronics*, Vol. 12, No. 7, 2023.

20. Taborri, J., J. Keogh, A. Kos, A. Santuz, A. Umek, C. Urbanczyk, E. van der Kruk and S. Rossi, "Sport Biomechanics Applications Using Inertial, Force, and EMG Sensors: A Literature Overview", *Applied Bionics and Biomechanics*, Vol. 2020, p. 2041549, 2020.

21. Navarro, X., T. B. Krueger, N. Lago, S. Micera, T. Stieglitz and P. Dario, "A Critical Review of Interfaces with the Peripheral Nervous System for the Control of Neuroprostheses and Hybrid Bionic Systems", *Journal of the Peripheral Nervous System: JPNS*, Vol. 10, No. 3, pp. 229–258, 2005.

22. Chowdhury, R. H., M. B. I. Reaz, M. A. B. M. Ali, A. A. A. Bakar, K. Chellappan and T. G. Chang, "Surface Electromyography Signal Processing and Classification Techniques", *Sensors*, Vol. 13, No. 9, pp. 12431–12466, 2013.

23. Norali, A. N., M. H. Mat Som and J. Kangar-arau, "Surface Electromyography Signal Processing and Application: A Review", pp. 1A4–1, Proceedings of the International Conference on Man-Machine Systems (ICoMMS), Batu Ferringhi, Malaysia, 2009.

24. Benbasat, A. Y., *An Inertial Measurement Unit for User Interfaces*, Master's thesis, Massachusetts Institute of Technology, Program in Media Arts and Sciences, 2000.

25. Ferrete Ribeiro, N. and C. P. Santos, "Inertial Measurement Units: A Brief State of the Art on Gait Analysis", *2017 IEEE 5th Portuguese Meeting on Bioengineering (ENBENG)*, pp. 1–4, Coimbra, Portugal, 2017.

26. Gao, G., M. Kyrarini, M. Razavi, X. Wang and A. Gräser, "Comparison of Dynamic Vision Sensor-Based and IMU-based Systems for Ankle Joint Angle Gait Analysis", *2016 2nd International Conference on Frontiers of Signal Processing (ICFSP)*, pp. 93–98, Warsaw, 2016.

27. Ayvaz, B., H. İlikçi, F. Bilgili and A. Ergenç, "IoT Networks and Online Image Pro-

cessing in IMU-Based Gait Analysis", A. Souri and S. Bendak (Editors), *Artificial Intelligence for Internet of Things (IoT) and Health Systems Operability*, Vol. 8 of *Engineering Cyber-Physical Systems and Critical Infrastructures*, Springer, Cham, 2024.

28. Tahmasbi, A., A. Shadmehr, B. Attarbashi Moghadam and S. Fereydounnia, "Does Kinesio Taping of Tibialis Posterior or Peroneus Longus Have An Immediate Effect on Improving Foot Posture, Dynamic Balance, and Biomechanical Variables in Young Women with Flexible Flatfoot?", *The Foot*, Vol. 56, p. 102032, 2023.

29. Lambert, J., Mike; Borresen, "Measuring Training Load in Sports", *International Journal of Sports Physiology and Performance*, Vol. 5, No. 3, 2010.

30. Yunusa, H., "The Impact of Physical Activity on Oxygen Saturation Levels", *International Journal of Physiology, Health and Physical Education*, Vol. 5, No. 1, pp. 21–23, 2023.

31. Michalik, K., M. Smolarek, M. Nowak, B. Pueo and P. Żmijewski, "Intra-Reliability of a Wearable Near-Infrared Sensor for Monitoring the Intensity of Exercise", *Applied Sciences*, Vol. 14, No. 13, p. 5856, 2024.

32. Lukowicz, P., F. Hanser, C. Szubski and W. Schobersberger, "Detecting and Interpreting Muscle Activity with Wearable Force Sensors", K. P. Fishkin, B. Schiele, P. Nixon and A. Quigley (Editors), *Pervasive Computing. Pervasive 2006*, Vol. 3968 of *Lecture Notes in Computer Science*, pp. 101–116, Springer, Berlin, Heidelberg, 2006.

33. Sakr, M. and C. Menon, "Study on the Force Myography Sensors Placement for Robust Hand Force Estimation", *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 1387–1392, Banff, AB, Canada, 2017.

34. Zizoua, C., M. Raison, S. Boukhenous, M. Attari and S. Achiche, "Detecting Muscle Contractions Using Strain Gauges", *Electronics Letters*, Vol. 52, No. 22, pp.

1836–1838, 2016.

35. Alvarez, J. T., L. F. Gerez, O. A. Araromi, J. G. Hunter, D. K. Choe, C. J. Payne, R. J. Wood and C. J. Walsh, “Towards Soft Wearable Strain Sensors for Muscle Activity Monitoring”, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, Vol. 30, pp. 2198–2206, 2022.

36. Manna, M. A. H. B. G. A., Soumya K.; Azhar, “Optimal Locations and Computational frameworks of FSR and IMU Sensors for Measuring Gait Abnormalities”, *Heliyon*, Vol. 9, No. 4, 2023.

37. Cagnie, B., J. Elliott, S. O’Leary, R. D’Hooge, N. Dickx and L. Danneels, “Muscle Functional MRI as an Imaging Tool to Evaluate Muscle Activity”, *Journal of Orthopaedic & Sports Physical Therapy*, Vol. 41, No. 11, pp. 896–903, 2011.

38. Kinugasa, R., Y. Kawakami and T. Fukunaga, “Quantitative Assessment of Skeletal Muscle Activation Using Muscle Functional MRI”, *Magnetic Resonance Imaging*, Vol. 24, No. 5, pp. 639–644, 2006.

39. Branscombe, L. M., *Evaluating a Wearable Sensor-Based Tibia Force Estimation Algorithm for Applications in Stress Fracture Reduction in Runners*, Master’s Thesis, Vanderbilt University, 2018.

40. Matijevich, E. S., L. R. Scott, P. Volgyesi, K. H. Derry and K. E. Zelik, “Combining Wearable Sensor Signals, Machine Learning and Biomechanics to Estimate Tibial Bone Force and Damage During Running”, *Human Movement Science*, Vol. 74, p. 102690, 2020.

41. Lloyd, D. G. and T. F. Besier, “An EMG-driven Musculoskeletal Model to Estimate Muscle Forces and Knee Joint Moments in vivo”, *Journal of Biomechanics*, Vol. 36, No. 6, pp. 765–776, 2003.

42. Buchanan, T. S., D. G. Lloyd, K. Manal and T. F. Besier, “Neuromusculoskeletal

Modeling: Estimation of Muscle Forces and Joint Moments and Movements from Measurements of Neural Command", *Journal of Applied Biomechanics*, Vol. 20, No. 4, pp. 367–395, 2004.

43. Waters, R. L. and S. Mulroy, "The Energy Expenditure of Normal and Pathologic Gait", *Gait & Posture*, Vol. 9, No. 3, pp. 207–231, 1999.

44. Dugan, S. A. and K. P. Bhat, "Biomechanics and Analysis of Running Gait", *Physical Medicine and Rehabilitation Clinics of North America*, Vol. 16, No. 3, pp. 603–621, 2005.

45. Farina, D., R. Merletti and R. M. Enoka, "The Extraction of Neural Strategies from the Surface EMG", *Journal of Applied Physiology*, Vol. 96, No. 4, pp. 1486–1495, 2004.

46. Su, H., Z. Li, P. Li and C. Yang, "EMG-Based Neural Network Control of an Upper-Limb Power-Assist Exoskeleton Robot", Vol. 7952, pp. 204–211, 2013.

47. Ahkami, B., K. Ahmed, A. Thesleff, L. Hargrove and M. Ortiz-Catalan, "Electromyography-Based Control of Lower Limb Prostheses: A Systematic Review", *IEEE Transactions on Medical Robotics and Bionics*, Vol. 5, No. 3, pp. 547–562, 2023.

48. Vetter, R. E. and M. L. Symonds, "Correlations Between Injury, Training Intensity, and Physical and Mental Exhaustion Among College Athletes", *Journal of Strength and Conditioning Research*, Vol. 24, No. 3, pp. 587–596, 2010.

49. Camomilla, V., E. Bergamini, S. Fantozzi and G. Vannozzi, "Trends Supporting the In-Field Use of Wearable Inertial Sensors for Sport Performance Evaluation: A Systematic Review", *Sensors*, Vol. 18, No. 3, p. 873, 2018.

50. *Digitalizing the Football Experience: A Study on Electronic Performance and Tracking Systems (EPTS) from the Perspective of Football Athletes and Training*

Staff, Proceedings of the New Trends in HCI and Sports Workshop (NTSPORT 2022), Vancouver, Canada, 2022.

51. Rana, M. and V. Mittal, "Wearable Sensors for Real-Time Kinematics Analysis in Sports: A Review", *IEEE SENSORS JOURNAL*, Vol. 21, No. 2, 2021.

52. Mansurali, A. and A. B. Mahmoud, *Sports Analytics, Data-Driven Sports and Decision Intelligence*, chap. The Future of People Analytics in Sports, pp. 82–83, Springer, 2024.

53. Giggins, O. M., K. T. Sweeney and B. Caulfield, "Rehabilitation Exercise Assessment Using Inertial Sensors: A Cross-sectional Analytical Study", *Journal of NeuroEngineering and Rehabilitation*, Vol. 11, p. 158, 2014.

54. Bonato, P., "Advances in Wearable Technology and Applications in Physical Medicine and Rehabilitation", *Journal of NeuroEngineering and Rehabilitation*, Vol. 2, No. 1, p. 2, 2005.

55. Roy, S., M. S. Cheng, S.-S. Chang, J. Moore, G. Luca, S. Nawab and C. Luca, "A Combined sEMG and Accelerometer System for Monitoring Functional Activity in Stroke", *IEEE Transactions on Neural Systems and Rehabilitation Engineering : A Publication of the IEEE Engineering in Medicine and Biology Society*, Vol. 17, No. 6, 2010.

56. Hester, T., R. Hughes, D. Sherrill, B. Knorr, M. Akay, J. Stein and P. Bonato, "Using Wearable Sensors to Measure Motor Abilities Following Stroke", *International Workshop on Wearable and Implantable Body Sensor Networks (BSN'06)*, pp. 4–8, Cambridge, MA, USA, 2006.

57. Spyropoulos, E., E. Chroni and G. Athanassiou, "Muscle Fatigue Estimation in Repetitive Lifting Task Using Surface Electromyography-Based Analysis", *Journal of Ergonomics*, Vol. 05, 2015.

58. Falla, D., D. Farina and T. Graven-Nielsen, “Experimental Muscle Pain Results in Reorganization of Coordination among Trapezius Muscle Subdivisions During Repetitive Shoulder Flexion”, *Experimental Brain Research. Experimentelle Hirnforschung. Expérimentation Cérébrale*, Vol. 178, No. 3, pp. 385–93, 2007.

59. Winter, D. A., *Biomechanics and Motor Control of Human Movement*, Wiley, New York, 2nd edition, 1990.

60. Abu-Mostafa, Y. S., M. Magdon-Ismail and H.-T. Lin, *Learning from Data*, AMLBook, New York, 2012.

61. Prince, S. J., *Understanding Deep Learning*, The MIT Press, 2023.

62. Kuhn, M. and K. Johnson, *Feature Engineering and Selection: A Practical Approach for Predictive Models*, CRC Press, Boca Raton, FL, 2019.

63. Blidh, H. and contributors, “Bleak: Bluetooth Low Energy platform Agnostic Klient”, `https://github.com/hbldh/bleak`, 2023, accessed on June 3, 2025.

64. Chamberland, F., Buteau, S. Tam, E. Campbell, A. Mortazavi, E. Scheme, P. Fortier, M. Boukadoum, A. Campeau-Lecours and B. Gosselin, “Novel Wearable HD-EMG Sensor With Shift-Robust Gesture Recognition Using Deep Learning”, *IEEE Transactions on Biomedical Circuits and Systems*, Vol. 17, No. 5, pp. 968–984, 2023.

65. Bishop, C. M., *Pattern Recognition and Machine Learning*, Springer, New York, 2006.

66. Géron, A., *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*, O’Reilly Media, Sebastopol, CA, 2nd edition, 2019.

67. Hyndman, R. J. and G. Athanasopoulos, *Forecasting: Principles and Practice*,

OTexts, Melbourne, Australia, 2nd edition, 2018.

68. Virtanen, P., R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. Carey, I. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. o. H. Ribeiro, F. Pedregosa, P. van Mulbregt and S. . Contributors, “SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python”, *Nature Methods*, Vol. 17, No. 3, pp. 261–272, 2020.

APPENDIX A: BUTTERWORTH FILTER

The Butterworth filter is a type of low-pass filter used to remove high-frequency noise from a signal. To create the envelope of the EMG signal, a second-order Butterworth filter is applied, beginning by defining the Nyquist frequency as

$$f_N = \frac{f_s}{2} \tag{A.1}$$

where f_s is the sampling frequency. The normalized cutoff frequency is then given by

$$\omega_c = \frac{f_c}{f_N} \tag{A.2}$$

with f_c denoting the desired cutoff frequency.

The digital low-pass filter coefficients are obtained in Python's `scipy.signal` module using

$$b, a = \text{butter}(2, \omega_c, \text{btype='low'}) \tag{A.3}$$

which returns the numerator b and denominator a coefficients of the transfer function [68]. The inputs of the `butter()` function are degree of filter, critical frequency and filter type. The zero-phase filtered envelope is then computed as

$$x_{\text{envelope}}(t) = \text{filtfilt}(b, a, x_{\text{rectified}}(t)). \tag{A.4}$$

A Butterworth filter of order n is characterized by a maximally flat passband, with squared magnitude response

$$|H(j\omega)|^2 = \frac{1}{1 + \left(\frac{\omega}{\omega_c}\right)^{2n}} \tag{A.5}$$

where ω_c is the cutoff angular frequency and ω is the angular frequency variable.

The analog filter poles lie on a unit circle in the left-half s-plane, given by

$$s_k = \omega_c \cdot e^{j\left(\frac{\pi}{2} + \frac{(2k+1)\pi}{2n}\right)}, \quad k = 0, 1, \ldots, n-1 \tag{A.6}$$

with only those having negative real parts retained for stability.

To convert the analog filter to a digital one, the bilinear transform

$$s = \frac{2}{T} \cdot \frac{1 - z^{-1}}{1 + z^{-1}} \tag{A.7}$$

is applied. This transformation warps the frequency axis, so the analog cutoff frequency is pre-warped using

$$\omega_c' = \tan\left(\frac{\pi f_c}{f_s}\right). \tag{A.8}$$

After transformation to the z-domain, the transfer function of the digital filter takes the form

$$H(z) = \frac{b_0 + b_1 z^{-1} + \cdots + b_n z^{-n}}{1 + a_1 z^{-1} + \cdots + a_n z^{-n}} \tag{A.9}$$

where the sets $\{b_k\}$ and $\{a_k\}$ are obtained by expanding the polynomials derived from the zeros, poles, and gain using standard filter design formulas. The `butter()` function returns:

- $\mathbf{b} = [b_0, b_1, \ldots, b_n]$: numerator coefficients
- $\mathbf{a} = [1, a_1, \ldots, a_n]$: denominator coefficients

Finally, the `filtfilt()` function from the same library applies the filter forward and backward. The forward pass is computed as

$$y_{\text{fwd}}[n] = b_0 x[n] + b_1 x[n-1] + b_2 x[n-2] - a_1 y_{\text{fwd}}[n-1] - a_2 y_{\text{fwd}}[n-2] \tag{A.10}$$

and the reverse pass as

$$y[n] = b_0 y_{\text{fwd}}[N-n] + b_1 y_{\text{fwd}}[N-n-1] + b_2 y_{\text{fwd}}[N-n-2] - a_1 y[n-1] - a_2 y[n-2] \tag{A.11}$$

resulting in zero phase distortion of the processed signal.

APPENDIX B: DATA SYNCHRONIZATION

The following code provides the synchronization for two data sources collected from different modules. The timestamps and channels from each data collection sources are shown in Table B.1 and Table B.2.

Table B.1. First few rows and columns of Left Shoe Sensor Data.

Timestamp	LeftShoe Channel 1	LeftShoe Channel2
2025-06-12 12:08:24.471000	3442	3337
2025-06-12 12:08:24.503000	3469	3335
2025-06-12 12:08:24.563000	3416	3307

Table B.2. First few rows and columns of EMG Data.

Timestamp	Avanti sensor 1: EMG 1 [V]
2025-06-12 12:08:24.900000	0.0
2025-06-12 12:08:24.950000	0.0
2025-06-12 12:08:25	-9.32646729032258e-06

The first step is to find the longest common time interval for each dataframe. The code is provided in Figure B.1.

```
start_common = max(LeftShoeDf.index.min(),mRightShoeDf.index.min(),
        processed_emg_df.index.min(), processed_imu_df.index.min())
end_common = min(LeftShoeDf.index.max(),RightShoeDf.index.max(),
        processed_emg_df.index.max(),processed_imu_df.index.max())
```

Figure B.1. Find the largest common interval

Then the EMG and IMU data frames are resampled by 20 Hz. which is the same sampling frequency with insole pressure sensor. The code that resamples the EMG and IMU data frames are provided in Figure B.2. The method of interpolation here is holding last value method, since EMG and IMU channels are sampled with higher frequency and linear interpolation would harm the behavior of raw signals.

Step 1: Extract the index as float seconds

```
original_times = processed_emg_df_absolute.index.to_numpy()
```

Step 2: Define 20 Hz time grid (0.05s steps)

```
start_time = original_times[0]
end_time = original_times[-1]
target_times = np.arange(start_time, end_time, 0.05)
```

Step 3: Find closest index for each target time

```
closest_indices = np.searchsorted(original_times, target_times)
closest_indices = np.clip(closest_indices, 0, len(original_times) - 1)
```

Step 4: Select and assign rows

```
downsampled_emg_df = processed_emg_df_absolute.iloc[closest_indices].copy()
downsampled_emg_df.index = target_times
```

Figure B.2. Downsampling the EMG signal to 20 Hz using nearest time-point indexing.

APPENDIX C: MACHINE LEARNING MODELS

This appendix briefly introduces the machine learning regression models used in this study: XGBoost, Support Vector Regression (SVR) with a Radial Basis Function (RBF) kernel, Random Forest Regressor, and Multi-Layer Perceptron (MLP). Each method is described in plain language and includes basic mathematical equations to support understanding.

C.1. XGBoost (Extreme Gradient Boosting)

XGBoost is a fast and efficient implementation of gradient boosting for decision trees. It builds an ensemble of weak learners (usually decision trees) in a stage-wise manner, where at each stage a new tree is trained to reduce the residual error of the existing ensemble. Given a dataset with input features $\mathbf{x}_i \in \mathbb{R}^d$ and target values y_i, the model prediction after t trees is

$$\hat{y}_i^{(t)} = \sum_{k=1}^{t} f_k(\mathbf{x}_i), \quad f_k \in \mathcal{F} \tag{C.1}$$

where $\mathcal{F}$ is the space of regression trees. The model is trained to minimize the regularized objective function

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l\big(y_i, \hat{y}_i^{(t-1)} + f_t(\mathbf{x}_i)\big) + \Omega(f_t) \tag{C.2}$$

with l denoting a loss function (e.g., squared error) and

$$\Omega(f_t) = \gamma T + \frac{1}{2}\lambda\|w\|^2 \tag{C.3}$$

as the regularization term, where T is the number of leaves in the tree and w the vector of leaf weights.

C.2. Support Vector Regression (SVR) with RBF Kernel

Support Vector Regression tries to find a function that approximates the data within a tolerance margin ϵ while keeping the model as flat as possible. In the linear case, the function has the form

$$f(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + b \tag{C.4}$$

and is obtained by solving the optimization problem

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2 \quad \text{subject to} \quad |y_i - f(\mathbf{x}_i)| \leq \epsilon. \tag{C.5}$$

For nonlinear cases, the input is mapped to a higher-dimensional space via a kernel function such as the radial basis function (RBF) kernel

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma\|\mathbf{x}_i - \mathbf{x}_j\|^2\right) \tag{C.6}$$

which enables SVR to capture nonlinear relationships in the data.

C.3. Random Forest Regressor

A Random Forest is an ensemble of decision trees, each trained on a random subset of the data and features. The final prediction is the average of the predictions of all trees:

$$\hat{y}_i = \frac{1}{M}\sum_{m=1}^{M} f_m(\mathbf{x}_i) \tag{C.7}$$

where M is the total number of trees and f_m is the prediction from the m-th tree. This approach reduces overfitting by combining bootstrap aggregation (bagging) with random feature selection at each split.

C.4. Multi-Layer Perceptron (MLP)

An MLP is a feedforward neural network composed of multiple layers of nodes (neurons), where each neuron performs a weighted sum of its inputs followed by a non-linear activation function. For a single-hidden-layer MLP, the output can be expressed as

$$\hat{y} = f_{\text{out}}\left(\sum_{j=1}^{H} w_j^{(2)} \cdot f_{\text{act}}\left(\sum_{i=1}^{d} w_{ji}^{(1)} x_i + b_j^{(1)}\right) + b^{(2)}\right) \tag{C.8}$$

where f_{act} is the activation function (e.g., ReLU), f_{out} is the output activation (e.g., identity for regression), H is the number of hidden units, $w_{ji}^{(1)}$ and $b_j^{(1)}$ are the weights and biases for the hidden layer, and $w_j^{(2)}$ and $b^{(2)}$ are the weights and biases for the output layer. MLPs are typically trained via gradient descent to minimize a loss function such as the mean squared error.

APPENDIX D: REGULARIZATION

Regularization is a technique used to prevent overfitting in machine learning models by penalizing large model weights.

D.1. L2 Regularization (Ridge)

L2 regularization adds a penalty based on the squared values of the weights as

$$\text{Loss} = \text{MSE} + \lambda \sum_j w_j^2. \tag{D.1}$$

This penalty discourages the model from using large weights and helps keep the model simpler.

D.2. L1 Regularization (Lasso)

L1 regularization adds a penalty based on the absolute values of the weights as

$$\text{Loss} = \text{MSE} + \lambda \sum_j |w_j|. \tag{D.2}$$

This regularization can shrink some weights exactly to zero and is often used for feature selection.

D.2.1. Elastic Net

Elastic Net combines L1 and L2 penalties as

$$\text{Loss} = \text{MSE} + \lambda_1 \sum_j |w_j| + \lambda_2 \sum_j w_j^2. \tag{D.3}$$

APPENDIX E: FIGURE REFERENCES

E.1. EMG Signal Decomposition Figure

Figure 1.1 , © 2016 IEEE, is reprinted, with permission, from D. Farina and A. Holobar, "Characterization of Human Motor Units From Surface EMG Decomposition".

E.2. Illustration of sEMG and needle EMG

Figure 1.2, under an open access Creative Common CC BY license, is taken from Nam, D., J. M. Cha and K. S. Park, "Next-Generation Wearable Biosensors Developed with Flexible Bio-Chips", Micromachines, Vol. 12, No. 1, p. 64, 2021, https://www.mdpi.com/2072-666X/12/1/64. *Proceedings of the IEEE*, vol. 104, no. 2, pp. 353–373, 2016.

E.3. Oxygenated Hemoglobin and Deoxygenated Hemoglobin Trend Figure

Figure 2.1 is obtained from Gersak, G., Vesna; Gersak, "NIRS: Measuring changes in muscle oxygenation and the detection of muscle activity", 19th IMEKO World Congress 2009, which is an open access article.

E.4. Inverse and Forward Biomechanics Flow Figure

Figure 2.3 is reused with agreement between İbrahim Kerem Tokdemir and Wolters Kluwer Health, Inc. ("Wolters Kluwer Health, Inc.") consists of license details and the terms and conditions provided by Wolters Kluwer Health, Inc. and Copyright Clearance Center. License Number 6090830894043.

E.5. Frontal, Sagittal and Transverse Plane Figure

Figure 3.1 is generated by OpenAI, “ChatGPT (June 2025 version)”, figure generated using ChatGPT based on author instructions.

APPENDIX F: ETHICAL APPROVAL

The experiments conducted in this thesis were approved by the Boğaziçi University Human Research Ethics Committee for Science and Engineering Disciplines, under the decision of application number 202-05, on June 19, 2025.