Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Computer Science

Sigbjørn Nøst Skauge

Vigorous Activity Detection in Human Activity Recognition

Master's thesis in Computer Science
Supervisor: Kerstin Bach
June 2021

NTNU
Norwegian University of
Science and Technology

Sigbjørn Nøst Skauge

Vigorous Activity Detection in Human Activity Recognition

Master's thesis in Computer Science
Supervisor: Kerstin Bach
June 2021

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Computer Science

NTNU
Norwegian University of
Science and Technology

Abstract

Human Activity Recognition is a field of study focusing on the detection of human movements in particular situations (exercise, labor, etc) or in daily life. Recently, the field has received more attention from the machine learning community since there are more datasets openly available and the data collection with Internet of Things devices has become easier to implement. This study focuses on building a machine learning model to better understand peoples health through vigorous physical activity detection. In previous studies at the Department of Computer Science at the Norwegian University of Science and Technology, activity recognition with vigorous data have not yet been performed.

HUNT4 is the fourth recurrence of the largest population based health study in Norway. The study is based upon the collection of data mainly through surveys and clinical measurements. In addition to the surveys, participants were invited to participate in another data collection by wearing body-worn sensors for one week. The participants wore two Axivity AX3 sensors, one on their lower back and the other on their thigh. This created a large dataset which would be preferable to analyse by using machine learning methods.

This thesis focuses on detection of vigorous physical activity in a subset of the HUNT4 dataset, namely the UngHUNT data. The UngHUNT data contains accelerometer data from adolescents wearing the previously mentioned body-worn sensors. This thesis uses machine learning to classify vigorous activity in this data. The machine learning algorithm used in this study is Extreme Gradient Boosting. The algorithm was selected by recognizing it's missing coverage in previous work on vigorous activity through a review of relevant literature and it's well-known strong performance on inbalanced, real-world datasets. To optimize the machine learning model built, training datasets were created and cross validation was performed to avoid overfitting. To further improve the model, feature selection, mix-in classification and different window sizes for the data were tested. To train the model, curated datasets are used containing both in-lab data and out-of-lab data. This thesis's results show that the machine learning model using a static sliding 3-second window is able to separate vigorous from non-vigorous activity with a precision, recall and F1-score of 95.56%, 95.38% and 95.40% respectively.

Sammendrag

Human Activity Recognition er et studiefelt hvor menneskets bevegelse er i fokus. Dette gjelder under spesielle situasjoner, som trening, arbeid eller lignende, eller under hverdagslige aktiviteter. Feltet har nylig fått mer oppmerksomhet fra folk som driver med maksinlæring ettersom datasett har blitt lettere tilgjengelig. Datainnsamling har også tatt store steg i forskningsfeltet hvor enheter i Tingenes internett har blitt enklere å implementere. Denne masteroppgaven fokuserer på bygge en maskinlæringsmodell for å forstå menneskelig bevegelse i krevende fysisk aktivitet bedre. I tidligere studier ved Norges teknisk-naturfaglige universitet har ikke krevende fysisk aktivitet vært undersøkt.

HUNT4 er den fjerde iterasjonen av Norges største befolkningsbaserte helseundersøkelse. Undersøkelsen er basert på innsamling av data gjennom spørre-undersøkelser og kliniske målinger. I tillegg be deltagere spurt om å delta i en annen datainnsamling ved å feste to aksellerometere på kroppen for en uke. Deltagerne fikk påsatt to Axivity AX3 sensorer. Den ene sensoren ble plassert ved korsryggen, mens den andre ble plassert midt på låret. Dette skapte et stort datasett som man prefererer å analysere med metoder fra maskinlæring.

Denne oppgaven fokuserer på klassifisering av krevende fysisk aktivitet i en spesifikk del av HUNT4 datasettet. Denne delen, kalt UngHUNT, inneholder alle deltagere som var ungdommer under undersøkelsen. For å gjøre dette brukes metoder fra maskinlæring og algoritmen brukt i denne oppgaven er Extreme Gradient Boosting. Denne algoritmen ble valgt på grunn av dens manglende tilstedeværelse i relatert arbeid i studiefeltet. Dette ble funnet gjennom et literatursøk. For å optimalisere algoritmen for krevende fysisk aktivitet ble nye datasett som inneholder slik aktivitet laget. Kryssvalidering ble brukt for å unngå overtilpasning (overfitting). I tillegg ble feature selection, mix-in klassifisering og forskjellige vindusstørrelser i data testet for å forbedre maskinlæringsmodellens resultater. Hovedresultatene i denne oppgaven viser en maskinlæringsmodell som bruker data med tresekunders vindu. Denne modellen klarte å oppnå presisjon, recall og F1-score på henholdsvis 95.56%, 95.38% og 95.40%.

Preface

This thesis was conducted at the Data and Artificial Intelligence Group of the Department of Computer Science at NTNU. The scope of the project was decided in cooperation with our supervisor Kerstin Bach.

Firstly, we would like express our thanks and gratitude to Kerstin for her guidance during the whole project. We would also like to give a special thank Aleksej Logacjov for the help with implementing the Vigorous Human Activity Recognition Pipeline in the HAR framework and giving input on the writing of the thesis. In addition, we would like to thank Paul Jarle Mork, Atle Kongsvold, and Ellen Marie Bardal for answering questions regarding the datasets and previous health studies at NTNU.

Sigbjørn Nøst Skauge
Trondheim, June 15, 2021

Contents

List of Figures

List of Tables

Abbreviations

ARC Activity Recognition Chain

CNN Convolutional neural network

CPUs Central Processing Units

DFT discrete Fourier transform

FFT fast Fourier transform

FN false negative

FP false positive

FS-MODEL Feature Selection Model

HAR Human Activity Recognition

IDI The Department of Computer Science

K-NN K-Nearest Neighbors

LOC Lundamo Obstacle Course

LOOCV leave one out cross validation

NTNU Norwegian University of Science and Technology

RBF Running Backward and Forward

SVM Support-Vector Machine

TAH Trondheim Adolescents Handball

TN true negative

TP true positive

XGBoost Extreme Gradient Boosting

Chapter 1

Introduction

Human Activity Recognition (HAR) is a broad research field in the scope of health studies focusing on recognizing human activity based on sensor data. The field tries to detect human movements in particular situations (exercise, labor, etc) or daily life. Recently, the field has received more attention from the machine learning community since there are more datasets openly available. Analyzing large datasets with accelerometer data manually is a daunting task. Machine learning algorithms have the benefit of only needing a smaller dataset as training data to explore a much larger dataset. There are different health benefits connected to such research. An example of a HAR system in action, is a surgical skill rating system as proposed in Hung et al. [2018]. This study uses sensor data to rate surgical skills by using machine learning classifiers. For our study, the focus is set on leisure physical activity.

The classification of human activities using machine learning have become useful over the years, since machine learning techniques have become more accessible Trost et al. [2011]. In the context of this study, machine learning models are trained on datasets similar to the data collected through the HUNT4 study, which had over 35 000 participants wearing accelerometer sensors for a week[1]. The training data is labelled with known activities and this study demonstrates how to develop a machine learning model to classify these activities in new, unseen data. By creating a model that is able to classify activities found in accelerometer data, one helps cohort health studies by providing a tool that allows for a more detailed data analysis, which in turn is crucial for a healthy population.

Higher levels of physical activity are associated with a lower risk of cardiovascular disease [Ramakrishnan et al., 2021]. This project focuses on HAR tasks using machine learning to classify vigorous physical activities, giving cohort health studies a tool to detect vigorous activity in accelerometer data. Optimizing a HAR system to classify

[1] https://www.ntnu.no/hunt/forskning, last accessed 08.12.2020

vigorous activities is a rather new part of the field, where improvements to already existing solutions using everyday living activities are needed. The data used in this thesis is time series data, which uses time as an index for the acceleration signal. The training data consists of recorded data with a length up to 120 minutes for a single subject. These data include a various range of activities, where some of them are vigorous activities. Vigorous activity is defined and further described in section 2.9. The overall aim of the study is to build a model that is able to analyse accelerometer data and give statistics for vigorous activities present in the UngHUNT dataset from the UngHUNT study. To do this three main goals have been set and will be described in the next section.

1.1 Goals and Research Questions

This section introduces the research goals for this project and the relevant research questions related to each individual goal.

The HUNT4 study produced a large dataset, which would be preferable to explore using machine learning models. The overall aim of this thesis is to create a machine learning model that can recognize periods of vigorous activity.

Aim of the thesis *To create a machine learning model that can recognize periods of vigorous activity in the HUNT4 dataset.*

This model would be helpful for further health studies on the HUNT4 data or similar data from accelerometers placed on the lower back and thigh. To achieve this aim, we formulate goals and define research questions that are addressed in this thesis:

Goal 1 *To research and describe existing machine learning approaches for vigorous activity recognition in HAR datasets.*

HAR is an active research field and has lately gotten more attention, also from the machine learning community. However, for the particular task of vigorous physical activity we need to do a literature search to understand the state of the art and identify possible gaps in the field.

Research question 1.1 *What is the state of the art in research when detecting periods of vigorous activity in accelerometer data using machine learning?*

Goal 2 *To find a suitable window size for vigorous activity detection in long-term HAR datasets.*

Detecting vigorous activities is a task that differs from detection of every day living activities, since vigorous activities are shorter in general. This can be seen in chapter 4 and turns out to be a difficult problem. Shorter activities could mean a need for shorter windows in the training data. Window sizes are explained in more detail in section 5.1. This goal is set to evaluate the impact of shorter window sizes.

Setting this window size for the detection tool is a challenging optimization problem, since too large windows could ignore sections of the window containing vigorous activity, while too small windows are generally slow to process and increase the data to be managed in large datasets. This window does also need to be reasonable for public health research.

Research question 2.1 *What is a suitable window size for vigorous activity detection in long-term HAR datasets?*

Goal 3 *To create and evaluate a machine learning model for vigorous activity.*

The third goal of this project is to train and optimize a machine learning model to classify vigorous data. The model has to classify 50Hz data since that is the frequency used in the HUNT4 dataset. To reach this goal, a new training dataset is needed, which includes relevant vigorous training data. This is crucial to be able to create a new machine learning model for vigorous activity classification. Three research questions were created to evaluate the new vigorous machine learning model.

Research question 3.1 *How well does the vigorous machine learning model classify activities?*

Research question 3.2 *Which features are needed to obtain the best results for classification of vigorous activities?*

Research question 3.3 *How well does the vigorous machine learning model separate vigorous activity from non-vigorous activity?*

1.2 Research Methods

This thesis uses different research methods to discover and research in the field of HAR and vigorous activities. Firstly, a literature review is conducted to obtain insight into the research field, before experiments are performed to test how previous general HAR studies adapts to vigorous data.

To get insight into machine learning methods used in related work, a Structured Literature Review is performed. The Structured Literature review in this project is performed to gain background information in the field of HAR and to describe the state-of-the-art on how far the field has come regarding recognizing vigorous activity.

Experiments are performed to both reuse information and work done in the HAR field by previous students and professors at Norwegian University of Science and Technology (NTNU). This thesis aims to reproduce previous work and applies it in a new field by focusing on vigorous activity. The scientific method is applied to reproduce previous work in a new setting, using the same hypotheses as in previous work and customizing them to the environment of vigorous activity by creating new machine learning models.

1.3 Thesis Structure

Chapter 2: Background An explanation of the machine learning concepts, previous work and other theory relevant for this study.

Chapter 3: Related Work A look into related work in the HAR field with a focus towards vigorous activity and machine learning.

Chapter 4: Datasets An overview of the dataset used to train and evaluate the machine learning model.

Chapter 5: Methods An explanation of the methods used in experiments to improve the classifier.

Chapter 6: Experiments and Results The experimental setup and results.

Chapter 7: Evaluation and Discussion An evaluation and discussion of the results from the previous section.

Chapter 8: Conclusion and Future Work A conclusion upon the evaluation of results from the previous chapter and a look into future work in the field of vigorous physical activity HAR.

Chapter 2

Background

In this chapter machine learning methods relevant to the thesis are explained, together with other relevant theory and background information needed to better understand the study.

2.1 The HAR Framework

The HAR Framework[1] is a framework created by the NTNU AI Lab[2] to make previous HAR studies from the Department of Computer Science The Department of Computer Science (IDI)[3] easily reproducible. The framework hosts a variety of functions, such as functionality to train machine learning models, use existing models and other services to process accelerometer data. This includes the extraction of the raw data coming from accerlerometers, synchronizing the data if more than one sensor is used and pre-processing the signals to run machine learning classifiers on the data. The framework can be configured to create various features and has a user interface to monitor the data analysis.

2.2 HUNT4 Study

The HUNT study is Norway's largest populated health study. The first data gathered was in 1984[4]. Later, there have been four studies in total where the most recent one called HUNT4 happened between 2017 and 2019[5].

[1] https://github.com/ntnu-ai-lab/hunt4-har-framework

[2] https://www.ntnu.edu/ailab

[3] https://www.ntnu.no/idi

[4] https://ntnu.no/hunt/om, last accessed: 2020-11-03

[5] https://ntnu.no/hunt/hunt4, last accessed: 2020-11-03

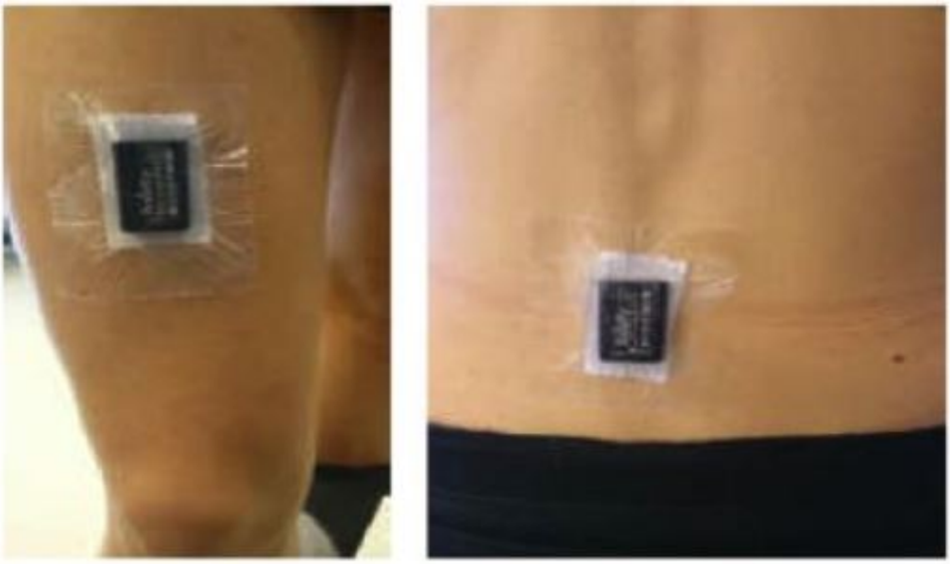


Figure 2.1: The placement of the accelerometer sensors. The first image shows a sensor placed on a person's thigh. The second image shows sensor placement on the lower back.

More than 56 000 people from Trøndelag participated in the HUNT4 study. Previously, these studies only consisted of people from the northern part of Trøndelag, but after the counties got merged in 2019, people from the southern part of Trøndelag have also participated through surveys. The study consists mainly of people older than 20 years. People younger than this also had the chance to participate, but their data was collected in a sub study called UngHUNT4[6]. As a part of the HUNT4, study participants were offered to wear a set of accelerometers over a period of one week. One of these was placed at the thigh and the other at the lower back of the participant. Images of the sensors' placement can be seen in figure 2.1. This is also illustrated in figure 2.2 with a more detailed view of the sensor orientations. These sensors provide data on the users movements while worn.

The sensor used in the study is named Axivity AX3 and is a data logger ideal for collecting longitudinal movement data. The sensor collects data in three dimensions and has a lifespan of 14 days when collecting data before needing to be recharged. The sensors were calibrated to measure values between -8 to +8 G.

[6] https://ntnu.no/hunt/unghunt, last accessed: 2020-11-03

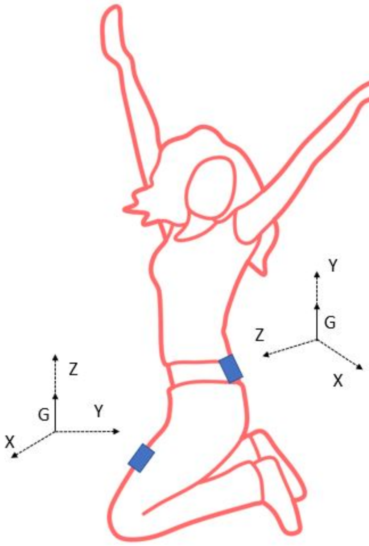


Figure 2.2: Illustration of the sensor placements of a person jumping. The image also shows directional axis for the data captured by the sensors.
Source: The base figure used is from dimensions https://www.dimensions.com/.

2.3 The Activity Recognition Chain

The Activity Recognition Chain (ARC), as presented in Bulling et al. [2013], is a typical process for creating a HAR system. Bulling et al. proposes that the process is separated into five distinct parts:

1. **Data Collection**: Collecting data from subjects wearing sensors.
2. **Data Pre-Processing**: Aligning and labelling the raw data. Data from multiple sensors do also need to be synchronized, if multiple sensors were used. Also one might need to remove noise from the data or resample data to a specific frequency.
3. **Data Segmentation**: Data windowing to classify upon segments of data instead of single datapoints.
4. **Feature Generation and Selection**: Extraction of relevant information in the data, called features, for every segment from the previous step.
5. **Classification**: Machine learning models take the produced features as input to make decisions, and in the case of this thesis classifications.

Hessen and Tessem [2016] made an illustration showing the ARC and can be seen in figure 2.3. For this thesis the most relevant parts of the ARC are the parts after Data Collection. However, an understanding of the Data Collection process and the importance of demographics is still important in HAR work, since movement patterns differ between different age groups and genders [Bartlett, 2007].

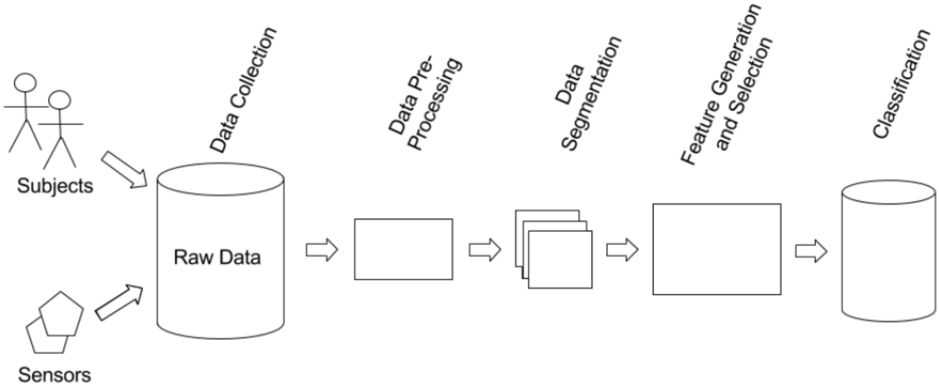


Figure 2.3: The process of creating a HAR system. Called the Activity Recognition Chain by Bulling et al. [2013]

2.4 Machine Learning

The goal of machine learning is to make a computer able to learn from experience with respect to some class of tasks and performance measures. To do this one needs a well posed learning problem. A well posed learning problem consist of three elements, namely a task to perform, a way to measure performance and a way to gain experience through training. The machine learning method then tries to model a function $\hat{F}(\boldsymbol{x})$ of the target function $F(\boldsymbol{x})$, which correctly maps the inputs to it's appropriate values [Mitchell, 1997].

There are two main different types of machine learning tasks, namely classification and regression tasks. In classification tasks the computer's goal is to classify an atomic result from the input, whereas regression focuses on predicting continuous values. This project focuses on supervised classification problems, using labelled data for classification.

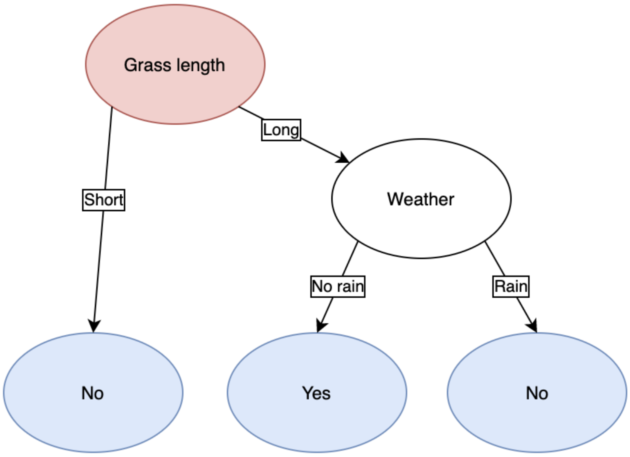


Figure 2.4: An example of a decision tree deciding whether to mow the lawn or not. The red node represents a root node, the white node an internal node and the blue nodes terminal nodes.

2.4.1 Decision Trees

The decision tree algorithm is a classification method that creates a map, a tree structure, that can take a variety of variables as input and compute an output based on the input values. The mapping in a decision tree consist of three different types of nodes:

- Root node: The initial node with zero incoming edges and zero or more outgoing edges.
- Internal node: A node containing one incoming edge and at least two outgoing edges.
- Terminal or leaf node: A node with one incoming edge and zero outgoing edges.

The decision tree learner computes an output by taking a vector as an input, runs tests for selected values in each node iterating down a path through internal nodes in the tree. These tests do checks for certain values contained by the input vector and then transcend the direction the test result provides, and does this for enough values to eventually end up at a leaf node with a decision (classification) [Mitchell, 1997].

During creation and training of the decision tree classifier the algorithm chooses the input attribute that gives the highest information gain as root node. Information gain is defined in Mitchell [1997] as the expected reduction in entropy. Mitchell's definition of entropy is defined in Equation 2.1, where c is the amount of classes and p_i is the portion of S belonging to class i. The definition of information gain can be seen in Equation 2.2, where the information gain is given from an attribute A relative to a collection S. Here $Values(A)$ is the set of all possible values for attribute A and S_v is the subset of S where attribute A has value v.

$$Entropy(S) = \sum_{i=1}^{c} -p_i log_2 p_i \tag{2.1}$$

$$Gain(S,A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v) \tag{2.2}$$

After the algorithm has chosen its root node it follows the same process for internal nodes, selecting the next attribute to split on by the decisions information gain. This process is performed until the tree structure is complete, which means that each path in the tree ends up at a terminal node. Decision tree learners on their own are weak learners, which means that decision trees will not work well for more complex problems. Decision trees are however found useful when combined with each other in ensemble methods.

2.4.2 Ensemble Learners

Ensemble learners use a number of weak learners to increase accuracy in predictions. The decision tree learner discussed in the previous section is an example of such a weak learner used in ensemble methods. Ensemble trees allow for an extraction of overall feature importances over the decision trees, which is useful during feature selection. The importance of features can be calculated in multiple ways, but this thesis uses the total amount the feature appears in the tree, which is the amount of times the feature is used split either a root node or internal node.

Popular techniques for constructing ensemble learners are bootstrap aggregation [Breiman, 1996a], also called bagging, and boosting algorithms [Freund and Schapire, 1996]. The bagging method creates additional training data which replicates the original training data. This improves the ensemble learner's stability and accuracy. Boosting in ensemble methods uses an additive approach when creating new weak classifiers for the ensemble learner. This method tries to fix previous weak classifier's mistakes by making the new weak classifiers focus on the misclassified input data. Successful performance boost by the usage of ensemble methods, are demonstrated in various papers, where Breiman [1996b] and, Kohavi and Kunz [1997] were some of the earlier ones in the field. These methods are often used, since they have turned out to perform well in real-world scenarios.

Random Forest

The random forest classifier is an example of an ensemble learner. Random forest uses decision tree classifiers as its weak classifiers where each tree is trained on a random subset of the input data Breiman [2001]. For the final prediction in classification, majority voting is used to select a single most probable class from the different classifications made by the trees in the model.

Extreme Gradient Boosting

Extreme Gradient Boosting (XGBoost) is an ensemble tree learner. The algorithm has shown that it often has good perform in real world scenarios, both in terms of accuracy and speed through system optimization[7]. Gradient boosted trees have been used in machine learning for some time, and some of the earliest applications of these methods were documented in Friedman [2001]. XGBoost, as other machine learning algorithms, tries to make an estimation over the domains target function by minimizing the model's loss function as described in Friedman [2001]. Friedman's definition of the function estimation can be seen in equation 2.3. Here Friedman is restricting F to be a parameterized class of functions. For gradient boosted trees, these functions $h(\boldsymbol{x},\boldsymbol{a}_m)$ resemble decision trees and the main differences from the random forest implementation is this additive approach in equation 2.3. The task then becomes optimizing the parameters $\boldsymbol{a}_m$ and the weight β_m, where this in the case of gradient boosted trees becomes the optimisation of trees by choosing split on parameters, split locations and terminal node.

$$F(\boldsymbol{x};\boldsymbol{P}) = F(\boldsymbol{x};\{\beta_m,\boldsymbol{a}_m\}_1^M) = \sum_{m=1}^{M} \beta_m h(\boldsymbol{x};\boldsymbol{a}_m) \tag{2.3}$$

The XGBoost framework defines an objective function for the algorithm to optimize as in Equation 2.4. Here the first term, l, is the loss function with y_i as the correct classification and $\hat{y}_i^{(t)}$ as the predicted class from decision tree t. The second term is the regularization term, where $\Omega(f_i)$ is the complexity of tree f_i. Regularization will be explained in section 2.7.2. The task for XGBoost then becomes minimising its overall loss and complexity through additive learning. For this thesis it is enough to

[7] https://xgboost.readthedocs.io/en/latest/tutorials/model.html, last accessed 26.11.2020

understand the additive approach of XGBoost, adding new trees to learn from previous tree's mistakes and the basic ideas of optimizing a machine learning algorithm.

$$obj = \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t)}) + \sum_{i=1}^{t} \Omega(f_i) \tag{2.4}$$

The XGBoost framework also has functionality to extract feature importances from a trained XGBoost model. This is useful when considering a cut in the number of features for the model and for explaining which features are the most important in the model.

2.5 Frequency Domain Transforms

Section 4.5 applies frequency domain transformation on the signal stream used as training data for the machine learning models. This section contains the theoretical basis needed to understand this frequency transformation.

Frequency domain transforms are mathematical operators that transform functions from the time domain to the frequency domain. This is done by applying the concept of Fourier analysis, which states that any real valued function can be expressed as a sum of sinusoidal functions.

2.5.1 Discrete Fourier Transform

Discrete Fourier transform (DFT) takes a finite sequence of equally spaced samples and returns a set of amplitudes contained within the sequence. The transformation can be seen in Equation 2.5 and the inverse transform in 2.6.

$$y(k) = \sum_{n=0}^{N-1} e^{-2\pi j \frac{kn}{N}} x(n) \tag{2.5}$$

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} e^{2\pi j \frac{kn}{N}} y(k) \tag{2.6}$$

Applying DFT on a real valued sequence of length n would result in an array of complex numbers $y = [y_0, y_1, ..., y_{k-1}, y_k]$, where the absolute value of each number in the array represents the amplitude, a_k, of a specific frequency in the spectrum. The frequency f_k which a_k corresponds to is found by using Equation 2.7. Here d represents the sample spacing and n the number of the samples within the window.

$$f_k = \frac{k}{d \times n} \tag{2.7}$$

Fast Fourier Transform

The DFT has a runtime of $\mathcal{O}(n^2)$, which would be time consuming when applied to large datasets. Computer algorithms implementing DFT, such as the fast Fourier transform (FFT), have achieved a runtime of $\mathcal{O}(n \log n)$ by using complex polynomial symmetry in the transformation[8].

2.6 Evaluation of Methods

Using methods from statistics is a common approach to measure a machine learning model's performance. Statistical methods can also be used to optimize and assess the performance of machine learning models. Throughout this section accuracy, recall, precision and F1-score will be defined by using the theory found in Sammut and Webb [2017].

To assess the machine learning models with the previously mentioned methods, a set of statistical terms is needed. These terms are true positive (TP), true negative (TN), false positive (FP) and false negative (FN). True positives are correctly classified positive samples. True negatives are correctly classified negative samples. For the false terms, the model fails to predict correctly. A false positive is a sample that was wrongly classified as positive. A false negative is a sample that was wrongly classified as negative.

Accuracy is the percent of correctly classified instances in the population.

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

Recall is the amount of true positives predicted by the model with respect to every positive sample in the data. This is also known as sensitivity.

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

Precision is the amount of correct positive classifications made by the model. This is found by dividing the true positives count by the total amount of positives predicted by

[8] `https://docs.scipy.org/doc/scipy/reference/tutorial/fft.html`, Last accessed: 02.06.2021.

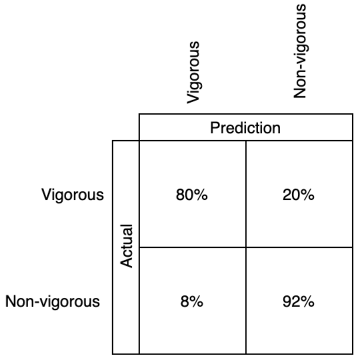
the model.

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

F1-score is the harmonic mean of recall and precision.

$$\text{F1-score} = 2 \times \frac{Recall \times Precision}{Recall + Precision}$$

All of these metrics can be computed from a **confusion matrix**. A confusion matrix shows where the machine learning model makes mistakes, by listing predictions against actual values in a grid. An example of a simple two by two confusion matrix can be seen in figure 2.5. In this example the two classes are vigorous and non-vigorous with the ground truth represented by row and the prediction by column. Then the optimal result would be for every entry to align at the diagonal, having every prediction be the same as the ground truth.



		Prediction	
		Vigorous	Non-vigorous
Actual	Vigorous	80%	20%
	Non-vigorous	8%	92%

Figure 2.5: An example of a confusion matrix containing two classes, *vigorous* and *non-vigorous*

2.7 Model Training

This section explains the process of training a machine learning model to make correct assumptions without any form of explicit programming.

2.7.1 The HAR Learning Problem

The field of HAR consist of activity recognition on a given population in a set time period. This is done by analyzing the populations individual movements. A definition of the HAR classification problem is as stated in Lara and Labrador [2013], which is also expressed in definition 2.7.1. This definition states that the goal is to find a temporal partition to classify in the data, making the HAR problem a classification problem.

Definition 2.7.1. ***HAR problem:*** *Given a set* $S = S_0, ..., S_{k-1}$ *of k time series, each one from a particular measured attribute, and all defined within time interval* $I = [t_\alpha, t_\omega]$*, the goal is to find a temporal partition* $(I_0, ..., I_{r-1})$ *of I, based on the data in S, and a set of labels representing the activity performed during each interval* I_j *(e.g., sitting, walking, etc.). This implies that time intervals* I_j *are consecutive, non-empty, non-overlapping, and such that* $\bigcup_{j=0}^{r-1} I_j = I$

2.7.2 Overfitting

Overfitting occurs when the machine learning algorithm is being trained to fit training data too much, making it harder for the algorithm to make correct choices when encountering unseen, new cases in the data. This subsection explains useful methods to avoid this issue.

Splitting data for training

A common way to handle data when training a machine learning model is to split the training data into two subsets. The first subset, normally about 80 percent of the set's size, is used as training data for the model. The second part is then being used to test the data after the model training is completed. The reason why this is such a common approach is the lack of actual test data in the field. Training data is often more accessible, if not the only accessible data for both training and validation. This is a common but basic method.

Cross Validation

Cross validation is a commonly used method to avoid overfitting in machine learning [Mitchell, 1997]. Cross validation splits the data by leaving a different part of the training data as test data for every iteration. This makes training data and validation data differ for every iteration, and is useful for giving an indication for how the machine learning model will function in practice by giving split-wise performance measures. Cross validation is one of the main methods used for machine learning model evaluation in this thesis, since this is what creates the confusion matrices presented later on. The two cross validation methods important for this thesis is k-fold cross validation and leave one out cross validation (LOOCV).

K-fold cross validation introduces folds to the cross validation. This method shuffles the data randomly before splitting the data into k folds. Starting out, one of the folds are being used for testing, and for every iteration in the training, another fold is added to the testing data and removed from the training data. A simplified version of Mitchell's k-fold cross validation algorithm can be seen in algorithm 1. This algorithm returns the mean error made by the model, while it can also be modified to return different interesting results from the for loop.

Algorithm 1: K-fold cross validation.

```
Partition the available data $D_0$ into $k$ disjoint subsets $T_1, T_2, ..., T_k$ of equal size,
  where this size is at least 30.
for i from 1 to k do
    use $T_i$ for the test set, and the remaining data for training set $S_i$
        1. $S_i \leftarrow D_0 - T_i$
        2. $model \leftarrow learn(S_i)$
        3. $\delta_i \leftarrow error_{T_i}(model)$
end
return $\frac{1}{k}\sum_{i=1}^{k} \delta_i$
```

In LOOCV, the data is split into k folds, just like in k-fold cross validation. This method is the same as the previously explained k-fold cross validation just with k set to one, leaving a single fold for test data every iteration. The training algorithm is then running k times over the data, leaving a different partition from the data as test data for every iteration. For example, if a dataset is divided into eight folds, each fold is selected as test data once and the rest as training data. This creates eight models, one for each fold, which get evaluated against the iteration's particular test fold. This is useful in HAR use cases, since one often handles datasets containing subjects. This gives the opportunity to create subject-wise statistics for cross-validation.

Regularization

Regulatization is a method used to avoid overfitting in ensemble methods and is a core idea in avoiding overfitting in XGBoost. The regularization term controls the complexity of the model, which avoids overfitting. This is described in the documentation of the

XGBoost framework[9] and also mentioned as a suggested improvement to the decision tree algorithm in Mitchell [1997]. In short terms, regularization keeps the trees simple and thereof the complexity of the model low.

For XGBoost, the machine learning model used in this thesis, the regularization is defined as in Equation 2.8. In this equation the complexity of the tree f is given by the number of leaves in the tree (T), the vector w containing the score for every leaf and the constants γ and λ. XGBoost uses this definition of tree complexity to minimize the overall complexity together with the loss, as described in 2.4.2.

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \tag{2.8}$$

2.8 Previous Work

The first project experimenting with machine learning to solve HAR problems at IDI was the work of Hessen and Tessem [2016]. Their data was collected in-lab and for classification they used a Convolutional neural network (CNN) combined with a Hidden Markov model. They also experimented with dynamic windowing of data in addition to combining machine learning models with different training data to a type of voting classifier. This classifier was then selecting a prediction from the pool of classifier based upon the classifiers confidentiality in the prediction. At the same time Kongsvold [2016] and Bårdstu [2016] wrote their own reports about data collection using accelerometer sensors and how the data could be used in HAR work.

Vågeskår [2017] performed experiments with different window sizes on data from stroke patients. A year later, Reinsve [2018] performed experiments with transitions between activities, while in 2019, experiments with sampling rate to better sensor lifetime was performed by Garcia [2019]. The same year, Hay [2019] experimented with body-worn sensors for sleep-wake classification. All the work done previously in the HAR field at NTNU was using everyday living activities.

2.9 Vigorous Data

All training data during this thesis contain periods of physical activity with vigorous intensity. In short we call this vigorous activity. Stamatakis et al. [2021] defined vigorous

[9] https://xgboost.readthedocs.io/en/latest/tutorials/model.html, last accessed 26.11.2020

activity in daily living as in definition 2.9.1. Vigorous activity is therefore expanded upon and defined a bit broader in this thesis, and can be seen in definition 2.9.2.

Definition 2.9.1. ***Vigorous Intermittent Lifestyle Physical Activity**: Brief bouts of incidental physical activity that are done during activities of daily living.*

Definition 2.9.2. ***Vigorous Activity**: Brief bouts of physical activity that are done during activities of daily living or vigorous sessions of physical activity.*

As for actual vigorous activities in this project the following activities will be classified as vigorous:

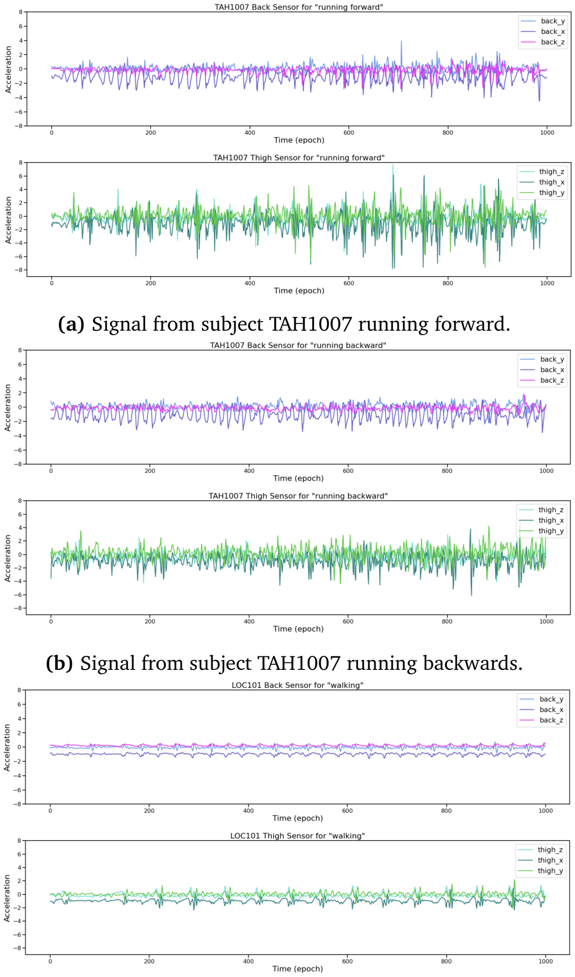
- Crabwalk
- Jumping
- All sorts of running
- All sorts of skipping

Where *crabwalk* is rapid sideways movement when playing handball. This movement is usually performed when the players are defending the goal. Examples of data streams from the accelerometer sensors can be seen in figure 2.6. The first data stream shows *running forward*, the second *running backward* and the last *walking*. The first two data streams contain vigorous activity, while *walking* is non-vigorous. To be able to classify a time period as vigorous a certain percentage of the period needs to be vigorous activity. For example, if a person is running forwards for a second and standing still for a minute, the whole period will not count as vigorous activity. However, if a period over a minute contains a majority of vigorous activities, that whole period should count as vigorous activity. This is done by finding the majority class present in portions of data called windows. Windowing of data and majority class selection will be explained in section 5.1.

There are also non-vigorous labels in the training data for the machine learning classifier, having labels such as *walking* and *standing*. A table displaying every label and whether the activity is vigorous or not can be seen in table 2.1.

Label	**Vigorous**
Walking	Non-vigorous
Crabwalk	Vigorous
Running	Vigorous
Running forward	Vigorous
Running backward	Vigorous
Skipping sideways	Vigorous
Shuffling	Non-vigorous
Stairs (ascending)	Non-vigorous
Stairs (descending)	Non-vigorous
Standing	Non-vigorous
Sitting	Non-vigorous
Transitions	Non-vigorous
Bending	Non-vigorous
Undefined	Vigorous
Jumping	Vigorous

Table 2.1: The full list of labels and whether or not the labels are vigorous.



(a) Signal from subject TAH1007 running forward.

(b) Signal from subject TAH1007 running backwards.

(c) Signal from subject LOC101 walking.

Figure 2.6: Three data streams for different labels. Running forward and running backwards are vigorous activities, while walking is non-vigorous.

Chapter 3

Related Work

This section presents related research to the work presented in this thesis. Most of the papers presented in this section were found through a structured literature review performed to get an understanding of the state of the art in the HAR field concerning vigorous activities. Some previously known papers are also added upon the papers found in the literature review to present the state-of-the-art research relevant for this thesis.

To find relevant papers Google Scholar[1] was used as the primary search engine. The literature review was performed in the early autumn 2020 and the search words included were *human activity recognition*, *machine learning*, *accelerometer*, *vigorous*, *sports* and *Axivity*. At the time of the literature review these search terms resulted in a total of about 120 papers. This was shortened to a total of six papers, were one was cut of as a result of a quality assessment later on. The criteria used to score the different studies was gathered from NTNU AILab's definition for a good research paper and can be seen in table B.1 in the appendix. All papers included were using the same sensors as the HUNT4 study, while most of the papers either included vigorous activity or mentioned vigorous activity as future work in the HAR field. Later on three papers were added because of their interesting sensor placement for data collection.

This chapter is separated in three sections. The first section gives a perspective into the different sensor placements and activity types during data collection. The second section focuses on the different machine learning methods used in the experiments presented in the papers. Lastly, a summary of the finds are discussed at the end of this chapter.

[1] https://scholar.google.com/, last accessed 03.12.2020

3.1 Sensors

This section focuses on sensor placement and the number of the sensors used for data collection. Most papers used a few body worn Axivity AX3 sensors, usually two to three, for their data collection.

Steels et al. [2020] classifies moves made in the sport of badminton. They collected data by having two subjects repeatedly perform common badminton moves. They placed a single Axivity AX6 sensor on the bottom of the racket's grip, the wrist or the upper arm of the subject.

Widianto et al. [2019] used five AX3 sensors from Axivity, placed on lower back, sternum, ankle of dominant foot and on both wrists, to measure the intensity of activities performed by 12 individuals. Their labels consisted of *sedentary*, *light*, *moderate* and *vigorous*. This study did not include any activity containing vigorous activity, but is included since adding vigorous activities was mentioned as a natural next step for future work. Their most vigorous activity in the training data was jogging, which was measured as *moderate*.

Hedayatrad et al. [2021] compared an older and more sensor ActiGraph GT3X+ with the newer sensor Axivity AX3 to ensure consistency with older devices. The participants in their study wore both sensors concurrently while performing prescribed activities. Both accelerometers obtained a balanced accuracy of 74%-96%, with the Axivity AX3 sensor outperforming the older sensor slightly for detection of posture and physical activity intensity.

Narayanan et al. [2020] evaluated different dual-accelerometer systems' accuracy classifying a broad range of behaviours in an free-living environment. Their participants wore three Axivity AX3 accelerometers for two hours. The sensors were placed on the thigh, back and wrist to eventually do comparisons for the combinations thigh–back, thigh–wrist and back–wrist, using machine learning to classify. The best performing accelerometer combination was the thigh-back with an overall accuracy of 95.6%. The other sensor combinations had an accuracy drop of at least 11%.

Small et al. [2020] experimented with lowering the sampling rate for accelerometer sensors to increase study monitoring periods. Their study try to assess the effect of such a reduction in sampling rate by looking having sensors collect data sampled at 25Hz and 100Hz. The sensor placement used in this study was a wrist-hip combination. The study concluded with the different sampled accelerometer data having predictable differences, which can be accounted for in inter-study comparisons. They also state that sampling rate should be reported in any physical activity study, tailored in study design and tailored to the outcome of interest.

3.2 Machine Learning Methods

The following section focuses on the machine learning methods used in the relevant papers. The studies presented achieve promising results for their specific areas of research. The papers use machine learning methods to classify sport specific and everyday living activities.

Steels et al. [2020] used a CNN for classification, were the activities classified were different kinds of badminton moves. The CNN had a precision of about 86% when only using accelerometer data and improved to 99% when combing the accelerometer data with a gyroscope. The paper also included a weight based neural network approach which could indicate clear mistakes made by the model. This weight based system used action length in classification, giving the probabilities for each label a weight decided by the length of the activity performed.

A different sport performance measurement system was developed in Khan et al. [2017]. They used five different models based on Support-Vector Machine (SVM), decision trees and K-Nearest Neighbors (K-NN) to find the best approach. Their research goal was to make a system that predicted shot direction and performance based on data gathered from sensors placed on subjects playing cricket. Having 20 different classes, they managed to achieve an average F1-score of above 88% for the models.

Sani et al. [2018] used matching networks, which applies K-NN by reusing a label in the most similar instances in a provided support set. In addition to this, they compared their approach with normal K-NN, SVM and feed-forward neural networks. Their final F1-scores ranged between 68% to 78%. The study used nine activities, including *jogging* and different paces of walking. They state that variety in training data is of importance, since there is a clear difference in model performance in personalized HAR systems and general HAR systems.

Sani et al. [2017] used both deep and shallow learning when comparing models trained on different sensor data. The data for the study was collected from 34 subjects between the range of 18 to 54 years. Their study focused on comparing the performance in models trained on two different sensors, namely wrist and thigh. For the training they used a SVM to learn the shallow features, and a CNN to learn deep features. For the results, thigh had the best score, outperforming the accuracy of the wrist data prediction with 11%. The best scoring algorithm was a hybrid solution between the SVM and the CNN. Their only vigorous activity was *jogging*, where the subject was jogging on a treadmill at moderate speed.

Widianto et al. [2019] trained a CNN for classification on everyday living activities. They concluded that the next step for their study would be to include *running* in the data to also classify vigorous activities.

3.3 Summary

There clearly is a lack of studies experimenting with general vigorous activities inside the HAR field. Only a few of the studies in the scope of this literature search actually dealt with vigorous data in particular. Also, almost every search result were released after 2015 which indicates that this field has not been addressed a lot previously.

Some studies targeted specific vigorous activities connected to sports, such as badminton or cricket. The badminton paper by Steels et al. [2020] provided a interesting approach combining a neural network with weights from the matching of length in the activity performed. A general vigorous activity classification model however, was not present in any of the papers. Some of these papers also had creative sensor placements, which is also an important topic in HAR work.

The most used algorithms were CNNs. Many seem to use deep learning algorithms for these problems, which have also given promising results. An interesting and different approach was the hybrid model that combined a CNN and a SVM to classify activities [Sani et al., 2017]. This type of approach was not mentioned in any of the other papers. In total these were the most common algorithms mentioned in the papers:

- CNN
- SVM
- K-NN

A lot of work has been put into this research field by professors and previous students at NTNU, but vigorous data is currently an undiscovered field. This also seems to be the case for the general state of art in this field. Even though neural networks, SVM and K-NN were the most popular machine learning methods used in the papers found in this section, XGBoost using decision trees were not discussed in these papers but have shown promising results in previous HAR studies at NTNU on every day living activities as shown in Hessen and Tessem [2016] and Reinsve [2018].

The reason for the lack of machine learning studies on general vigorous activities could be the lack of good datasets. Hedayatrad et al. [2021] shows that the Axivity

AX3 sensor outperforms the older sensor ActiGraph GT3X+ in detection of posture and physical activity intensity. The best performing sensor placement in Narayanan et al. [2020] was the thigh-back combination, which is the baseline for data collection in this thesis. Their study shows the importance of good training data, where the other sensor combinations had an accuracy drop of at least 11%.

Chapter 4

Datasets

Two datasets were produced using data from participants playing handball and traversing an obstacle course. These datasets were then used to create the training dataset used for the machine learning algorithm. The age group of the participants for both datasets are adolescents. This chapter explains the process of creating these datasets from the raw data signal and manual annotations, and presents the characteristics and differences between the datasets.

Both datasets contain accelerometer data recorded by the Axivity AX3 from adolescents' movements where one of the datasets contains data from two handball training sessions, while the other contains data from participants exercising in an obstacle course. Both datasets were recorded at 100Hz. The handball data was collected in an out-of-lab environment where the subjects performed a normal handball training session. The obstacle course data however, had a strict setup the adolescents had to follow. Therefore this data will be counted as in-lab data. The two datasets were combined into a single dataset which will be referred to as the training data in this thesis.

Additionally, during the course of this thesis, an additional dataset of young adults running backwards and forwards was created. This was done to both evaluate the machine learning algorithm on data gathered from young adults and to see if the machine learning algorithm was able to improve its precision on selected labels.

4.1 Trondheim Adolescents Handball

The Trondheim Adolescents Handball (TAH) dataset contains recordings of five handball players practicing handball twice in the span of 24 hours. The data collected for the TAH dataset was gathered in 2019. The subjects were all male with an average height of 182.50cm and an average weight of 77.17kg. The subjects had an accelerometer placed on the lower back and at the thigh. The adolescents wore the sensors for about

24 hours, which included two sessions of playing handball for about one and a half hour. Both handball sessions were video recorded for the manual labeling and it was from these sessions we built our dataset.

For this study the dataset was engineered to only include the handball sessions by removing areas containing low amounts of vigorous activity, such as sleep during the night. One of the players did only participate in one of the sessions, which makes one of the entries a bit shorter as only one handball session was included in the data collection. A example of the raw data can be seen in figure 4.1, where the handball sessions are marked with red squares. The start of the recording happened with the first handball session at afternoon the first day. The handball session was recorded with a video camera, so each participant's activities could be labelled manually after the training session.

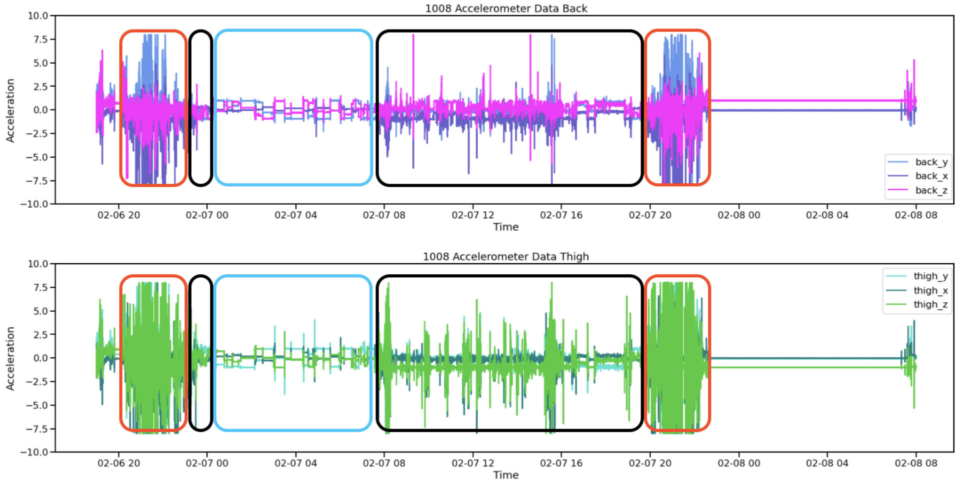


Figure 4.1: An example of handball subject's raw data from both back and thigh sensor. Red squares indicate handball sessions, blue sleep and black everyday living activities. The parts before and after these squares represent recorded data with the sensors not attached to the subject.

The time in between the two training sessions was also recorded with periods of sleep, school and free time displayed as blue and black. Here blue is the estimated period of sleep and black periods estimated periods of free time. It should be noted that these squares are estimated periods found in the visualisation of the data. The recording ended after the second handball session the second day.

In figure 4.2 one can see an example of the finalized training data, where the two handball sessions from the same subject as in figure 4.1 are annotated and combined into a single file. The ground truth for this example is shown as a black scatter plot and the split between the two days in the figure is in the long sitting session in the middle. An overview of the activity distribution in the dataset can be seen in table 4.1 *Running forward* is the most present vigorous activity in the dataset. The total amount of vigorous activity in the dataset is about one hour and 30 minutes, which is about 14% of the dataset.

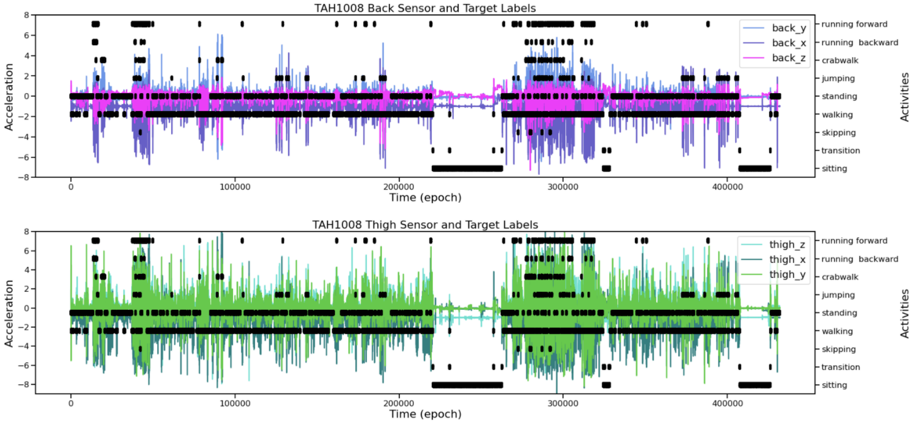


Figure 4.2: Labelled training data from subject TAH1008 playing handball. Here labels are shown as a black scatter plot together with the accelerometer data for each sensor.

Label	Distribution	Total time
Walking	47.4%	4 hours and 56 minutes
Standing	25.7%	2 hours and 41 minutes
Sitting	11.7%	1 hour and 13 minutes
Running forward	10.0%	1 hour
Crabwalk	2.9%	18 minutes
Running backward	1.1%	7 minutes
Jumping	0.1%	6 minutes
Transition	<0.1%	< 1 minute
Skipping	<0.1%	< 1 minute

Table 4.1: The label distribution in the TAH dataset.

4.2 Lundamo Obstacle Course

The Lundamo Obstacle Course (LOC) dataset was collected in an in-lab like environment. This data collection included a total of 18 participants, where 10 of the participants' data could be synchronised using the HAR Framework and hence are included in this new training dataset. Of these, seven participants were boys, while the remaining three were girls. The 10 subjects were all adolescents having an average height of 169.60cm and an average weight of 59.33kg. The subjects had an accelerometer placed on the lower back and at the thigh, collecting data of the subjects movements during the traversion of the obstacle course.

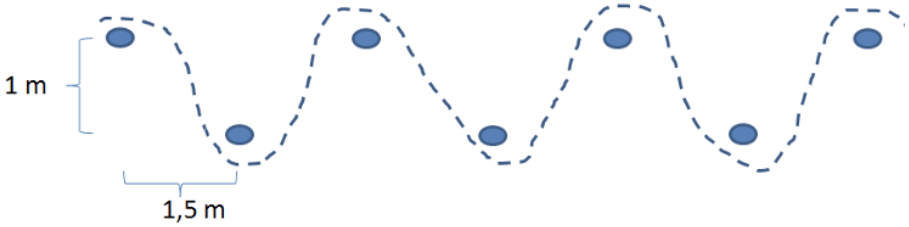


Figure 4.3: An obstacle course having the participant running forward with sideways motions to move around the cones.

The dataset recorded contained a variety of different movements, where our main focus is on the running parts since this is counted as a vigorous activity. The running parts were recorded in an obstacle course closely resembling the one in figure 4.3. The label distribution in the dataset can be seen in table 4.2. It should be noted that *running* in this dataset is mostly forwards running, but with a systematic sideways pattern created by the subjects traversing the obstacle course. This must not be confused with the labels *running forward* and *running backward* which is introduced in the TAH dataset. This will be kept as *running* for the classification model later on, to explore if the model actually separates these specific movements created in a more in-lab environment.

4.3 The Running Backward and Forward Dataset

During the evaluation of the machine learning model in this project, a certain mistake made by the model needed further experimentation. The model confused *running backward* with *running forward*, which can be seen in section 6.2. An additional dataset was created, to both evaluate the machine learning algorithm on data gathered from

Label	Distribution	Total time
Walking	40.3%	2 hours and 45 minutes
Standing	24.0%	1 hour and 55 minutes
Running	22.6%	1 hour and 32 minutes
Sitting	10.0%	41 minutes
Shuffling	2.5%	12 minutes
Undefined	0.2%	1 minute
Transition	0.1%	1 minute
Stairs (descending)	<0.1%	< 1 minute
Stairs (ascending)	<0.1%	< 1 minute
Bending	<0.1%	< 1 minute
Non-vigorous activity	<0.1%	< 1 minute

Table 4.2: The label distribution in the LOC dataset.

young adults and to see if the machine learning algorithm was able to improve its precision on the selected labels.

The in-lab dataset Running Backward and Forward (RBF) was produced to evaluate the machine learning model trained on the mentioned two datasets. This dataset was created by having four young adults run backwards and forwards for short periods. The young adults were two male and two female. The dataset contains mostly *running backward* and *running forward*, since these labels turned out to be difficult for the machine learning model to separate, which in turn will be covered further in chapter 5. In addition to these labels, other relevant labels were also mixed in.

4.4 Annotation Process

During the project period the raw data was engineered into the previously mentioned datasets. To do this the raw data from thigh and back sensor was synchronized by using functionality already present in the HAR Framework. After this process, the data was labelled by aligning a file containing the labels with the synchronized accelerometer data. This was done visually by aligning sensor spikes from heel drops performed at the start of each vigorous session. Then all the subjects from the two datasets were combined into a single dataset, creating our training dataset.

4.5 Data Resampling

The training dataset was resampled from 100Hz to 50Hz by applying a Fourier transformation to the dataset. This was performed to make the machine learning model learn features from 50Hz data, since the data collected in the HUNT4 study was collected in 50Hz.

4.6 Young HUNT

As a test dataset for processing larger amounts of data recorded for public health research, the UngHUNT will be used. This is raw unlabelled data which was collected during the HUNT4 survey. The survey is explained in more detail in section 2.2. The dataset contains a total of 4905 subjects, where 1977 were male and 2928 female. The average height of the participants were 168.86cm, while their average weight were 63.06kg. Average age for the participants were 17.2 years.

The UngHUNT data is particularly interesting since it contains movement data from adolescents over a week, which matches the demographics of the training data created during this study. The data collection was conducted for seven days, creating a large dataset to analyse. This is an example dataset on which the classifiers will be used when conducting public health research. For public health research it is interesting to describe the time spent inactive, in moderate activity and vigorous activity, where the resolution for the measurements usually is minutes of activity per day.

Chapter 5

Methods

This chapter presents the methods used in this thesis to develop the vigorous activity classifier. This work addresses goal 2 and 3 which was set in section 1.1. The datasets presented in the previous chapter are used to train, evaluate and optimize the classifier. An XGBoost model for every day living activities is already present in the HAR framework. A new XGBoost model is created by using the new training dataset, which consists of the TAH and the LOC dataset. This chapter explains how the HAR Framework was used together with various feature engineering methods to create and optimize this vigorous model. During various experimentation performed during this study different window sizes were tested. The window sizes tested through experiments are 1-, 2-, 3- and 5-second windows. Feature extraction together with feature selection and mix-in classification is also experimented with and is explained in this chapter.

The algorithm selected for the machine learning part of this study is XGBoost. The literature review from chapter 3 shows that this algorithm still has not been used in HAR studies using vigorous activity. XGBoost has however displayed great results in previous HAR work at NTNU by Vågeskår [2017] and Reinsve [2018].

5.1 Data Segmentation: Window Sizes

A data stream containing the training data is sent to the machine learning classifier for training, but to classify single data points is a hard task. There could exist overlapping signals for different activities in the data, which would confuse the classifier. To make this easier for the model, the data is separated into segments for both training and classification. This method is called a static sliding window, giving the model windows with a set size to classify. A single point in the accelerometer data from human movement depends on the adjacent data points, since the signal is continuous. Additionally one could also compute various features from these windows, by looking at trends, averages, minimums and maximums in the time series accelerometer data. This also includes frequency domain features as explained in section 2.5. The HAR Framework uses static window sizes and uses a 1.5 overlap when creating these windows.

Previous studies have shown that three to 5-second windows had the best precision for models using everyday living data [Vågeskår, 2017]. However, using large window sizes on vigorous data has a disadvantage of losing activities that are shorter than the window size. Figure 5.1 gives an example on how the window size effects the training data. This figure shows the length of different activities found in a part of the handball dataset. The top figure shows the labels for a 1-seconds window while the lower figure for 3-seconds windows of the same data sequence. The top figure shows the labels for a 1-seconds window while the lower figure for 3-seconds windows of the same data sequence. In the 3-second data labels the vigorous activities *running backward* and *jumping* are discarded as their duration is too short to mark them as a full window.

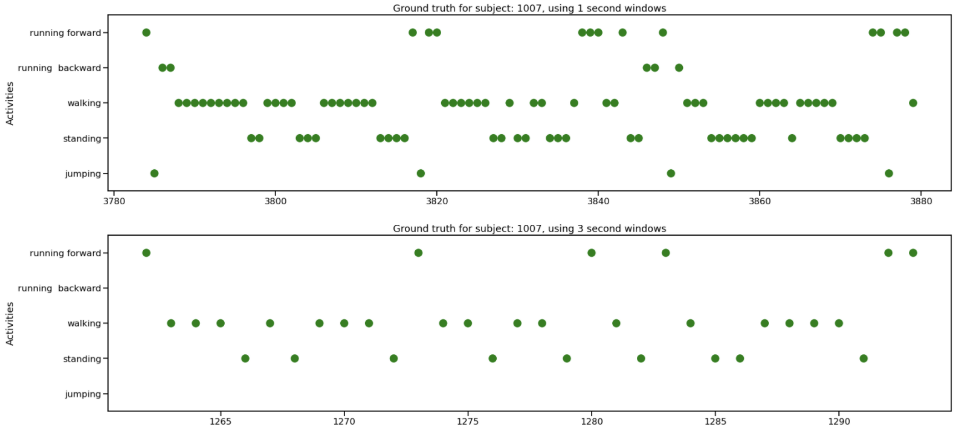


Figure 5.1: Plots of the ground truths for subject TAH1007 using two different window sizes. The top plot has 1-second windows, while the last plot has 3-second windows.

The training data is annotated with ground truths for every entry in the dataset. These ground truths were made from the video recording of the subjects, which in turn was aligned with the accelerometer signal. As an example a 5-second window containing 50Hz data gives 250 datapoints with their own labelled ground truth. The featurized dataset, however, consists of the features produced from these 250 datapoints, with a single label as a ground truth for this whole 5-second period. This is a result of the majority class selection performed on the 250 ground truths when creating this training data, where the majority class is selected from the these 250 ground truths.

This could be problematic for vigorous activities when using larger window sizes, since vigorous activities are shorter than most every day living activities.

The data collected from the handball sessions had most of these short activities. The length of the different activities found in the handball data can be seen in figure 5.2. The top figure shows the labels for a 1-seconds window while the lower figure for 3-seconds windows of the same data sequence. This can especially be seen in the 3-second data labels in figure 5.1, where the vigorous activities *running backward* and *jumping* are discarded as their duration has been too short to mark them as a full window.

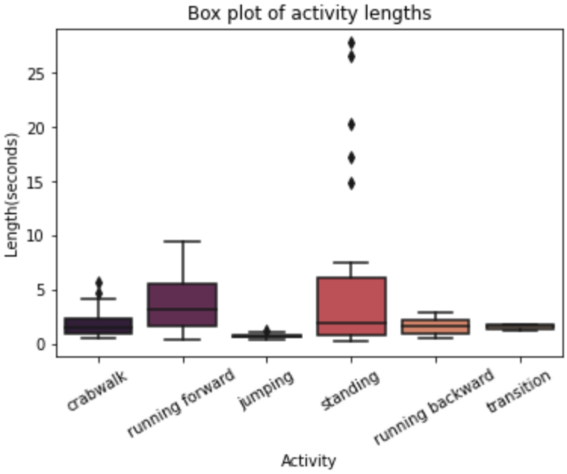
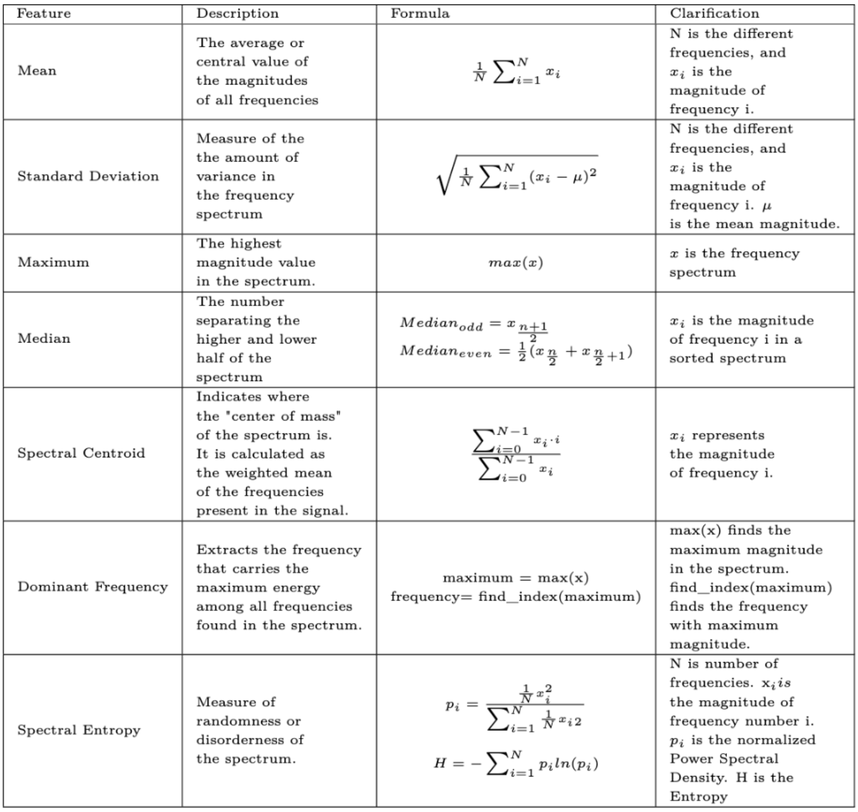


Figure 5.2: Box plot displaying the length of short activities in the TAH dataset from the handball sessions.

However, for the overall aim of this thesis, which is detecting vigorous activity in the HUNT4 data, the minor vigorous activity mistakes will most likely not have an significant impact on the overall results. This is because the vigorous activities most likely will be confused with other vigorous activities. This hypothesis is based of the accelerometer signals from section 2.9.2. Different window sizes are tested in section 6.1 of the experiments. To make the window sizes reasonable for public health research, the results from the machine learning model's predictions need to have some post-processing applied. This will be discussed shortly in chapter 8.

5.2 Feature Extraction

Giving an ensemble learner raw data is usually not the best approach for a high performing model. A set of features was extracted from the data by using functionality present in the HAR Framework. These include both time and frequency domain features. During training and classification, each feature is calculated for every window in the sliding window approach. This is also another reason why the smaller window sizes slow down the model. Every feature is explained in more detail in table 5.1 and table 5.2. Features were calculated for the x-, y-, and z-axis for both sensors and the norm of each data point. The actual models created by these features learn a total of 95 features before feature selection, since they use input from both sensors.

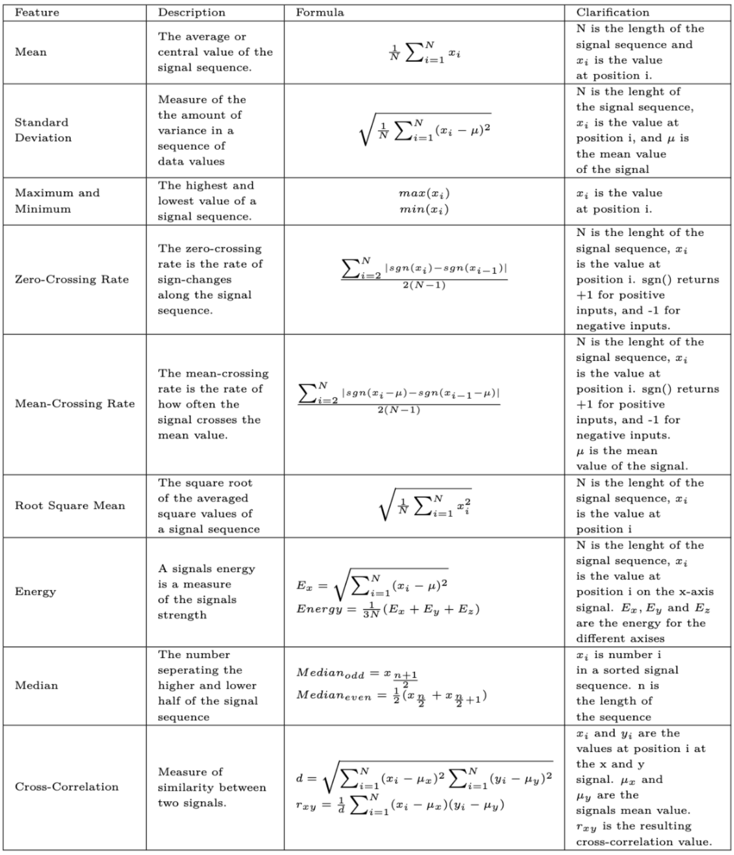


Feature	Description	Formula	Clarification
Mean	The average or central value of the magnitudes of all frequencies	$\frac{1}{N}\sum_{i=1}^{N} x_i$	N is the different frequencies, and x_i is the magnitude of frequency i.
Standard Deviation	Measure of the the amount of variance in the frequency spectrum	$\sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2}$	N is the different frequencies, and x_i is the magnitude of frequency i. μ is the mean magnitude.
Maximum	The highest magnitude value in the spectrum.	$max(x)$	x is the frequency spectrum
Median	The number separating the higher and lower half of the spectrum	$Median_{odd} = x_{\frac{n+1}{2}}$ $Median_{even} = \frac{1}{2}(x_{\frac{n}{2}} + x_{\frac{n}{2}+1})$	x_i is the magnitude of frequency i in a sorted spectrum
Spectral Centroid	Indicates where the "center of mass" of the spectrum is. It is calculated as the weighted mean of the frequencies present in the signal.	$\frac{\sum_{i=0}^{N-1} x_i \cdot i}{\sum_{i=0}^{N-1} x_i}$	x_i represents the magnitude of frequency i.

Dominant Frequency	Extracts the frequency that carries the maximum energy among all frequencies found in the spectrum.	maximum = max(x)
		frequency= find_index(maximum)
			max(x) finds the maximum magnitude in the spectrum. find_index(maximum) finds the frequency with maximum magnitude.

Spectral Entropy	Measure of randomness or disorderness of the spectrum.	$p_i = \frac{\frac{1}{N}x_i^2}{\sum_{i=1}^{N}\frac{1}{N}x_{i2}}$ $H = -\sum_{i=1}^{N} p_i ln(p_i)$	N is number of frequencies. $\mathrm{x}_i is$ the magnitude of frequency number i. p_i is the normalized Power Spectral Density. H is the Entropy

Table 5.1: Most common frequency domain features as described in Hessen and Tessem [2016].



Feature	Description	Formula	Clarification
Mean	The average or central value of the signal sequence.	$\frac{1}{N}\sum_{i=1}^{N} x_i$	N is the length of the signal sequence and x_i is the value at position i.
Standard Deviation	Measure of the the amount of variance in a sequence of data values	$\sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i-\mu)^2}$	N is the lenght of the signal sequence, x_i is the value at position i, and μ is the mean value of the signal
Maximum and Minimum	The highest and lowest value of a signal sequence.	$max(x_i)$ $min(x_i)$	x_i is the value at position i.
Zero-Crossing Rate	The zero-crossing rate is the rate of sign-changes along the signal sequence.	$\frac{\sum_{i=2}^{N} \lvert sgn(x_i)-sgn(x_{i-1})\rvert}{2(N-1)}$	N is the lenght of the signal sequence, x_i is the value at position i. sgn() returns +1 for positive inputs, and -1 for negative inputs.
Mean-Crossing Rate	The mean-crossing rate is the rate of how often the signal crosses the mean value.	$\frac{\sum_{i=2}^{N} \lvert sgn(x_i-\mu)-sgn(x_{i-1}-\mu)\rvert}{2(N-1)}$	N is the lenght of the signal sequence, x_i is the value at position i. sgn() returns +1 for positive inputs, and -1 for negative inputs. μ is the mean value of the signal.
Root Square Mean	The square root of the averaged square values of a signal sequence	$\sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^2}$	N is the lenght of the signal sequence, x_i is the value at position i
Energy	A signals energy is a measure of the signals strength	$E_x = \sqrt{\sum_{i=1}^{N}(x_i-\mu)^2}$ $Energy = \frac{1}{3N}(E_x + E_y + E_z)$	N is the lenght of the signal sequence, x_i is the value at position i on the x-axis signal. E_x, E_y and E_z are the energy for the different axises
Median	The number seperating the higher and lower half of the signal sequence	$Median_{odd} = x_{\frac{n+1}{2}}$ $Median_{even} = \frac{1}{2}(x_{\frac{n}{2}} + x_{\frac{n}{2}+1})$	x_i is number i in a sorted signal sequence. n is the length of the sequence
Cross-Correlation	Measure of similarity between two signals.	$d = \sqrt{\sum_{i=1}^{N}(x_i-\mu_x)^2 \sum_{i=1}^{N}(y_i-\mu_y)^2}$ $r_{xy} = \frac{1}{d}\sum_{i=1}^{N}(x_i-\mu_x)(y_i-\mu_y)$	x_i and y_i are the values at position i at the x and y signal. μ_x and μ_y are the signals mean value. r_{xy} is the resulting cross-correlation value.

Table 5.2: Most common time domain features as described in Hessen and Tessem [2016].

5.3 Feature Selection

This section explains the feature selection process that was performed for multiple reasons. Firstly, the model would become more transparent, therefore simpler to explain with less features. Also, the model could benefit from the removal of potential cluttering features, improving its performance. The feature selection process was performed on 2- and 3-second window models to investigate if slightly smaller window sizes would affect the F1-scores in any way. The process of evaluating the models with different amount of features is shown in figure 5.3. This figure illustrates how training data is processed to extract the feature importances and the metrics for the model with different amount of features.

When performing the feature selection LOOCV was done multiple times as can

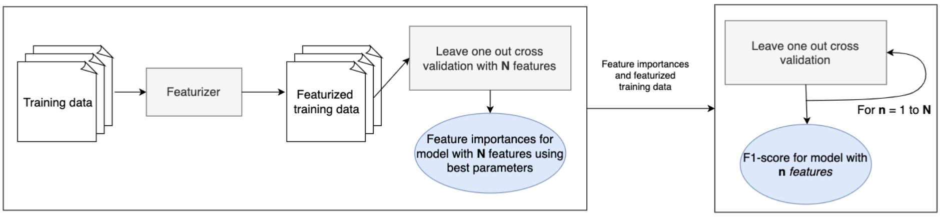

Training data
Featurizer
Featurized training data
Leave one out cross validation with N features
Feature importances for model with N features using best parameters
Feature importances and featurized training data
Leave one out cross validation
For n = 1 to N
F1-score for model with n features

Figure 5.3: The process of computing the F1-score for the development of vigorous classification models using various combination of features.

be seen in the last box in figure 5.3, starting with a run using all features to extract the feature importances. The F1-score of this run was then saved. After the initial LOOCV, multiple LOOCV runs were performed, starting at the most important feature and adding one feature for every iteration. For each run, the best F1-score was saved for comparison later on. After reviewing the results, a cut was made at 22 features. This amount of features was the number needed to surpass the model using all features, while having more features would make the model slower when running predictions. After this step, the 3-second window size model was created by using the 22 best features with a three second window size for further experimentation. The 3-second window model was selected for further experiments, since the model using a 2-second window size did not reach any higher F1-score for 22 features. For further references to this model it will be defined as the Feature Selection Model (FS-MODEL).

5.4 Mix-In Classification

This section explains how in-lab data was used in experiments to see if the model could increase it's performance on specific activities by giving it more data containing activities that were less present in the TAH and LOC dataset. During development we noticed that *running forward* and *running backward* was not detected well and we therefore decided to investigated the issue. It turned out that the periods of *running backward* were small, hence the possible need for more training data in a mix-in model. A successful approach of creating a mix-in model can be seen in [Hong et al., 2016]. Figure 5.4 shows a 3-second window model's predictions on the first subject in the RBF dataset. Here the model struggles to separate *running backward* and *running forward* which is the main focus in this experiments.

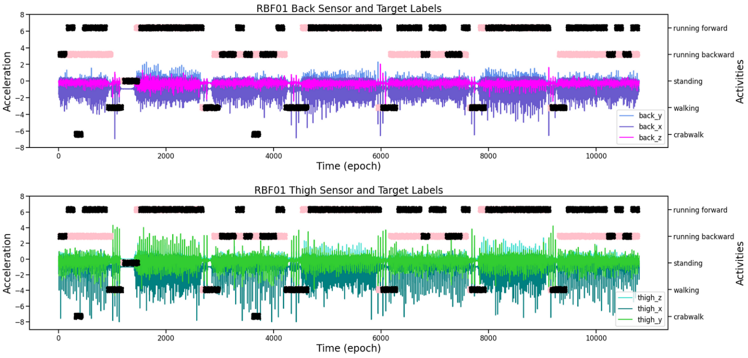


Figure 5.4: Comparison of ground truths (pink) and predictions (black) with the sensor data in the background for subject RBF01 in the RBF dataset. The predictions were created by a 3-second window model. Accelerometer data from both sensors can be seen in the background.

In the mix-in experiment, a model 3-second model is trained to predict on the RBF dataset, which creates a confusion matrix from the predicted values. Thereafter a new model is trained by mixing a single subject from the RBF dataset into the training data. Then the model trained on the mix-in dataset performs predictions on the remaining subjects in the RBF dataset. Then each confusion matrix from the predictions created by the mix-in models get averaged, creating a single matrix for the mix-in classifications. The experimental results can be seen in section 6.4 and will be further discussed in chapter 7.

5.5 Vigorous Activity Detection

For the last part of the project the FS-MODEL was tested on the UngHUNT dataset. A selected window in the data was identified as vigorous if the label was included in the list which was defined in chapter 4. This was implemented as post processing functionality and the results can be seen in section 6.5. This was done for multiple subjects in the UngHUNT dataset, but subject 4184201 proved to be particularly interesting. This subject had a day with a significant period of vigorous activity, which can be seen in figure 5.5. The data from this subject will be used to verify if the machine learning model manages to classify vigorous activity correctly in section 6.5. The plot in the figure shows accelerometer data from the sensors the subject wore. Large sections of lying can bee seen at the start of the data stream, where the subject most likely is asleep and is turning around in bed. The period of vigorous activity is from about 11:30 to 13:00. Before and after this period, the signal looks like combinations of walking, standing and sitting. At the end of the day, the signal from the subject shows lying down once again.

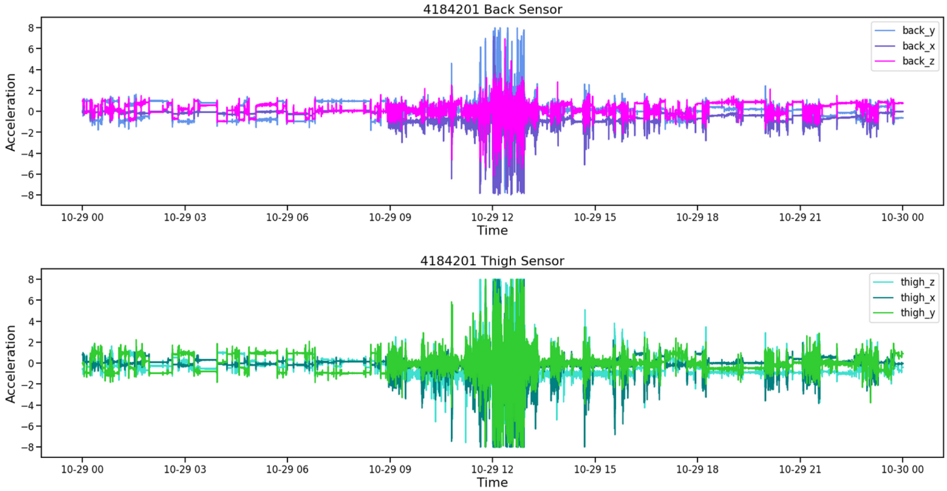


Figure 5.5: Raw accelerometer signal from a single day. The signal is from subject 4184201 in the UngHUNT dataset.

Chapter 6

Experiments and Results

This chapter showcases the results from the experiments conducted and addresses the research questions formulated in the beginning of this thesis. The discussions of the results are presented in chapter 7.

The experiments were conducted on a remote computer containing the HAR Framework, datasets and hardware needed. This remote computer uses multiple Central Processing Units (CPUs) and has a total of 144 CPUs available. To create the results the HAR Framework's implemetation of LOOCV was performed on the training data, which consisted of the TAH and LOC datasets. The average time for a single run of LOOCV took 20 minutes, using every feature defined in the framework. For the mix-in classification, the RBF dataset was used in addition to the training data. In the last experiment model created through feature selection was tested on the UngHUNT data. For each run of LOOCV in the experiments the following parameters were tested:

- Learning rate: 0.1
- Max depth: [20, 30, 40]
- Number of estimators: [60,70]
- Subsample: 0.6
- L2 regularization: 1.0

6.1 XGBoost Model Window Sizes

This section presents the results obtained from the LOOCV run using different window sizes on both the TAH and the LOC dataset. The results are presented using two different metrics, where the models' performance for single activities is shown by a comparison of precision. Subject-wise performance is measured by the models' accuracy for individual subjects. The window sizes being used in these experiments are 1-, 3- and 5-second windows.

6.1.1 Experimental Results

Figure 6.1 and figure 6.2 presents the results of the experiments to decide on the window size. Both figures show the average scores after the LOOCV runs using the training dataset, which consists of the LOC and TAH dataset.

Figure 6.1 shows the precision for selected activities. The precision for each activity was calculated from the confusion matrix created by LOOCV. The different runs' accuracy for each subject can be seen in the last figure. This was acquired by having the LOOCV algorithm save predictions for the subject left out of the iteration, creating predictions for every subject in the training dataset.

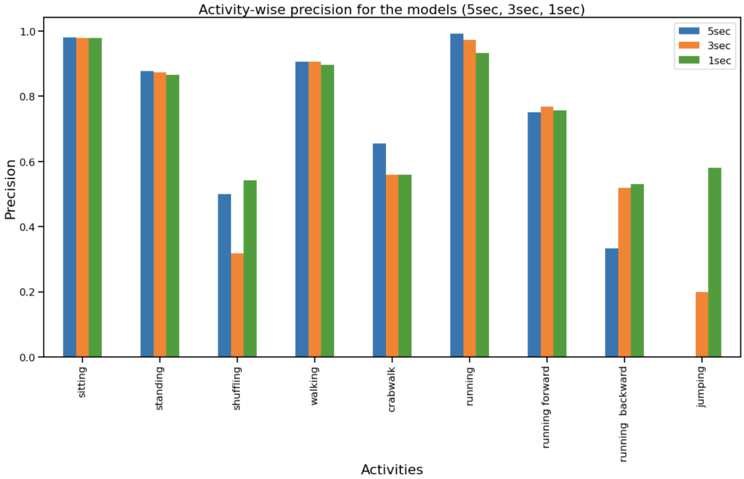


Figure 6.1: The activity-wise precision from both the TAH dataset and the LOC dataset. The plot shows results from the 1-, 3- and 5-second window model.

The 5-second window model has the best precision for every activity, except for *shuffling*, *running backward*, *running forward* and *jumping*. The 1-second window model performed best in these labels, except from *running forward* where the 3-second model had the best precision.

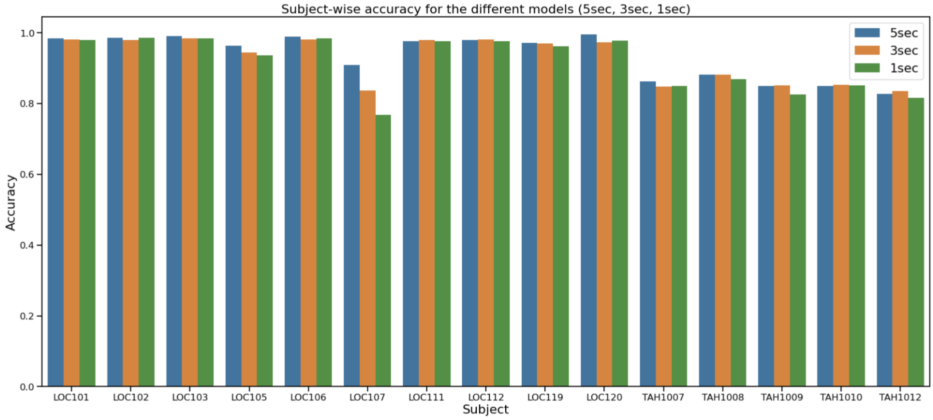


Figure 6.2: The overall subject-wise accuracy for both the TAH and the LOC dataset, using the predictions from the 1-, 3- and 5-second window model.

Figure 6.2 shows performance per subjection, where the 5-second window model has the highest average accuracy, while the overall accuracy is slightly lower for smaller window sizes. The overall accuracy of the models' predictions for the subjects are similar, but by looking into the details, the predictions on the LOC dataset is higher than for the TAH dataset. Every prediction on subjects in the LOC dataset has high accuracy, except for subject LOC107 which has the lowest accuracy for the 5-second window model and even lower for smaller window sizes.

6.2 Vigorous Activity Model

This section includes results from the vigorous activity model created with LOOCV. LOOCV gives the best parameters to avoid overfitting and also has the possibility to give subject-wise and activity-wise performance measures. This model use the training dataset (TAH and LOC) with a window size of three seconds.

6.2.1 Experimental Results

A confusion matrix was created from the result of the LOOCV. The matrix only includes the most relevant activities for the vigorous activity classifier and can be seen in figure 6.3, while the full matrix can be seen in appendix C.1.

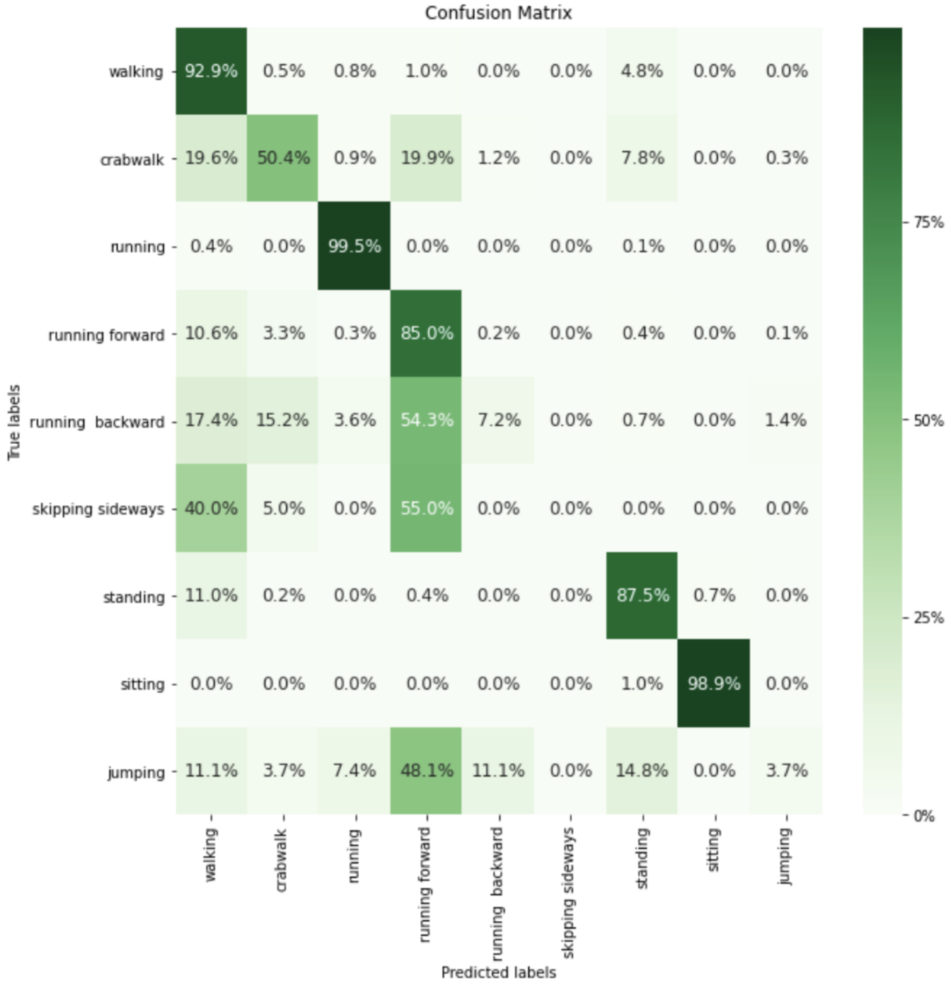


Figure 6.3: A cropped confusion matrix from the standard model's LOOCV run.

This matrix is missing some columns and their respective rows for readability. The cropped matrix contains the most interesting finds and the full matrix can be seen in figure C.1 in the appendix. This is also the reason why the numbers do not add up for every row in figure 6.3, since the matrix miss columns with wrongly predicted values. The optimal result for a confusion matrix would be for every value in a row to be at the diagonal with 100%. This is mostly the case for *walking*, *running*, *running forward*, *standing* and *sitting*, while other activities seem to be more spread. *Sitting* has a very high precision, while the model confuses *standing* and *walking*. *Running backward* is mainly confused with *running forward*, which is also the case for *skipping sideways* and *jumping*. *Running* from the LOC dataset is the vigorous activity with the highest score of 92.5% correctly predicted samples.

The main goal is the classification of vigorous activity as a group. Figure 6.4 shows the model's results for classification of vigorous and non-vigorous labels. This figure adds up every vigorous label and non-vigorous label found in appendix C. The corresponding metrics for this confusion matrix can be seen in table 6.1.

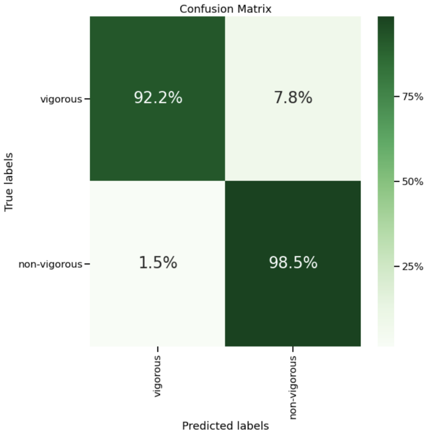


Figure 6.4: The confusion matrix for vigorous and non-vigorous activities from the standard model's LOOCV run.

Metric	Score
Precision	95.56%
Recall	95.38%
F1-score	95.40%

Table 6.1: The metric scores for the standard model's LOOCV run when considering vigorous and non-vigorous labels.

6.3 XGBoost Model Feature Selection

Feature selection is performed to both improve the model's performance and also to simplify the model, and make it more transparent. Feature selection is performed on the machine learning model using both 2- and 3-second window data to compare the performance between the window sizes for different features. An in depth explanation of how the feature selection was done can be seen in section 5.3.

6.3.1 Experimental Results

The result from the feature selection for the different models can be seen in figure 6.5. The 3-second sliding window approach produced the F1-scores shown by the blue graph in figure 6.5, where the score reaches a level comparable to the full feature model at 22 features. The 2-second window model's performance with the different amount of features can be seen in the same figure, where the 2-second window model reaches the same F1-score as the 3-second window model. When more features are added the 3-second window model gets a better F1-score than the 2-second window model in most cases.

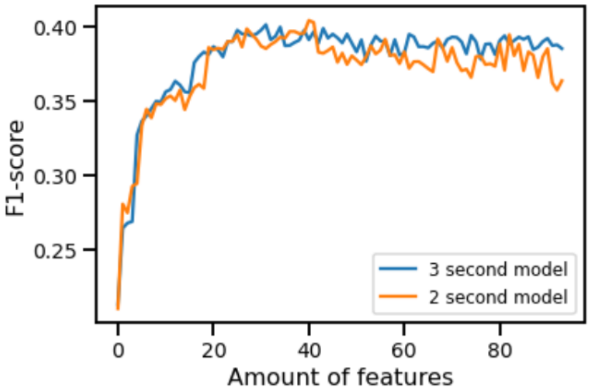


Figure 6.5: Comparison of F1-scores from feature selection for both window sizes.

The final features for the FS-MODEL are shown in figure 6.6. This model is the 3-second window model using the best 22 features. The y-axis in this plot shows the feature names, while the x-axis shows the number of decision tree nodes using the selected feature to split on. For the 22 best features, the x-orientation of the accelerometer data is the most important. The y-orientation is the least used orientation. Both back and thigh sensor are used in the best scoring features, while time domain features are more important than frequency domain features. The feature importances for the 3-second window model with all features can be found in figure C.3 in the appendix.

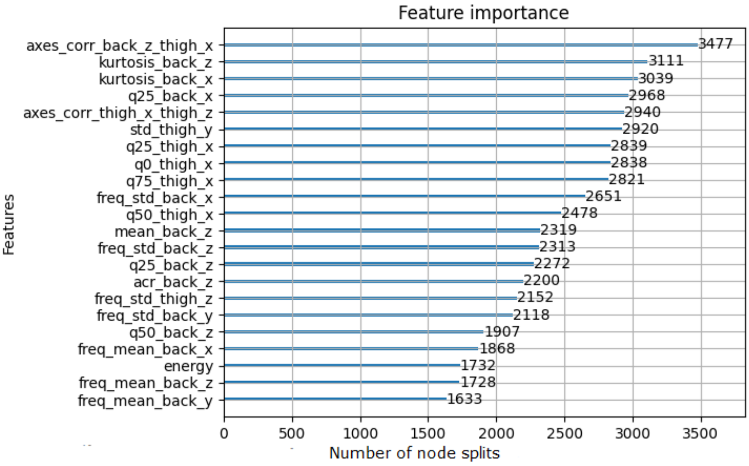


Figure 6.6: The final feature importances for the model created from feature selection.

6.4 XGBoost Mix-In Models

This section shows relevant results from an analysis performed to understand why the 3-second window model misses to classify *running backwards* with a high percentage.

The first part of this experiment uses the standard vigorous 3-second window model which was trained on the TAH and LOC dataset. This model performs predictions on the RBF dataset through using the evaluation script present in the HAR Framework. Finally an experiment is performed where a new model is trained again including one of the subjects in dataset RBF together with the training dataset. Afterwards, these models are evaluated on the remaining subjects in the RBF dataset. This is performed for every subject in the RBF dataset. To create the results for the mix-in models, the results for every individual model are added together and averaged to create a single confusion matrix.

6.4.1 Experimental Results

The results of these experiments are summarized in two confusion matrices, which can be seen in figures 6.7 and 6.8. Figure 6.7 shows the regular 3-second window model's mistakes in a confusion matrix, while figure 6.8 shows the same for the mix-in models. Both models perform well for the labels *running forward* and *standing*, while both confuse *walking* with standing, where the second model performs a bit better. The main focus in this experiment was *running backward* which is confused with *running* *forward* for both models, where the first model classified 28.5% of the samples correctly and the second model got 38.9% correct predictions.

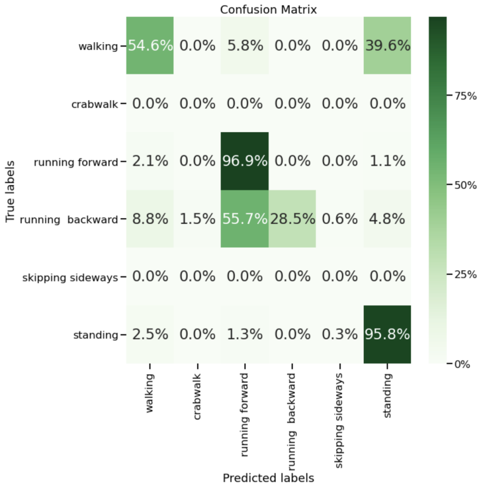


Figure 6.7: Confusion matrix produced from the regular 3-second window model's 0predictions on the RBF dataset.

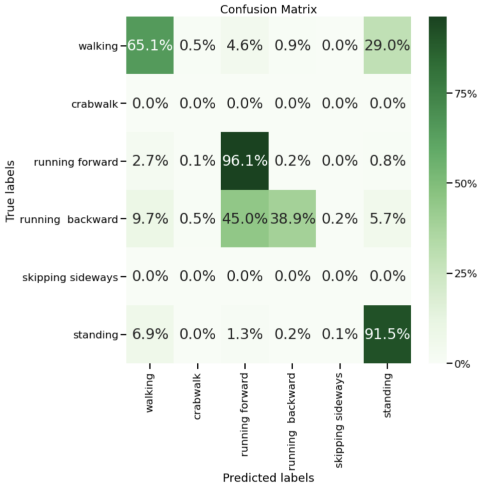


Figure 6.8: Confusion matrix produced from the mix-in models' predictions on the RBF dataset.

6.5 Vigorous HUNT4-HAR Pipeline

This section presents the results from running the vigorous HUNT4-HAR Pipeline on the accelerometer data for subject 4184201 in the UngHUNT data. The FS-MODEL is used for this experiment. The model returns 3-second window predictions, which is then transformed into 5-second window predictions by using the nearest fit transformation. The transformation is needed to make the predictions fit together with the original pipeline, which has classifiers predicting with 5-second windows.

6.5.1 Experimental Results

In figure 6.9 one can see a plot of the predictions of UngHUNT participant 4184201 from the FS-MODEL. This accelerometer signal is from the whole day and is added together with a scatter plot of the machine learning model's prediction of vigorous periods. The scatter plot shows a vigorous period from approximately 11:30 to 13:00.

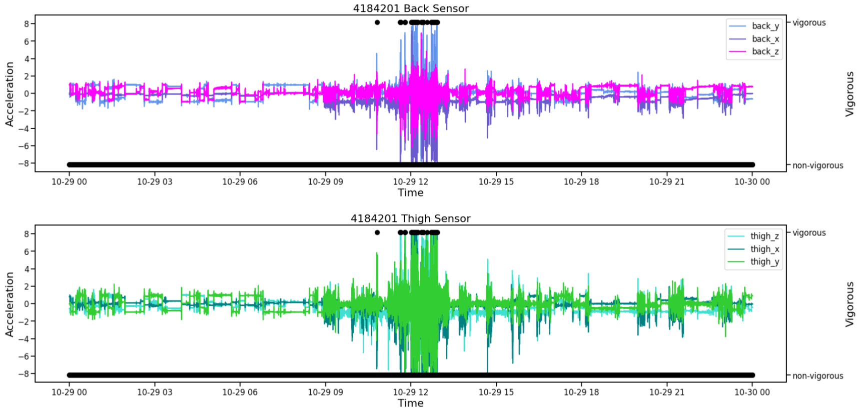


Figure 6.9: A view of the whole day (24 hours) of subject 4184201 in the UngHUNT dataset. The plot shows accelerometer data with a period of vigorous activity around 12:00. The black scatter plot shows if the machine learning model prediction is vigorous or non-vigorous.

Figure 6.10 zooms in the period between 12:00 and 13:00 and includes two variants of the output labels from the FS-MODEL: one with the detailed activity classes (in red with labels on the y1-axis) and the vigorous classification (in black with labels on the y2-axis). The plot is sliced into a time span of about an hour. The data is from a weekday at 12:00 to 13:00 during which the subject includes a high percentage of vigorous activity. The output of the vigorous classifier can be seen in black in the scatter plot.

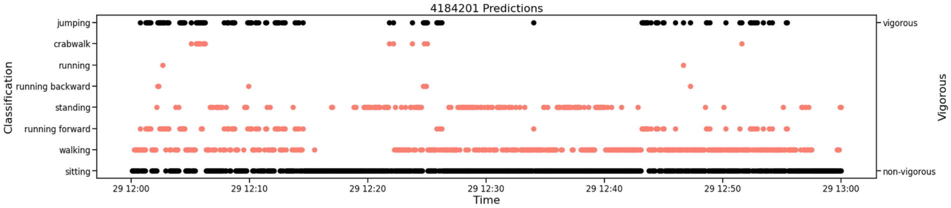


Figure 6.10: An hour containing portions of vigorous activity from subject 4184201 in the UngHUNT dataset. The black scatter plot shows if the data was classified as vigorous or not. The orange scatter plot in the background shows the actual classes predicted.

Figure 6.11 shows the accelerometer signal for the whole day together with a black scatter plot showing whether the predictions from the machine learning model are vigorous or not.

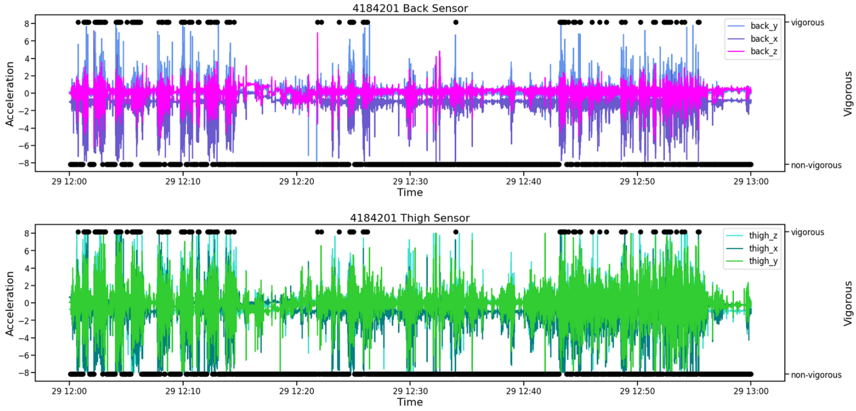


Figure 6.11: The accelerometer signal for subject 4184201 in the UngHUNT dataset together with a black scatter plot showing if the activity was vigorous or not. The plot shows the vigorous period from 12:00 to 13:00 on the selected day.

Chapter 7

Evaluation and Discussion

This chapter summarizes the findings of the experiments and discusses the results presented in the previous chapter. The first section of this chapter focuses on window sizes and how the machine learning models created during experiments performed with these different window sizes. The second part focuses on the performance of the XGBoost classifier for vigorous activity detection, together with the literature study looking at the state of the art in the vigorous physical activity HAR field. In this chapter the research questions defined in chapter 1 is discussed and evaluated in the light of results from the research and experiments.

7.1 Window Sizes

The results in section 6.1 show that the vigorous models with larger window sizes has better average accuracy, but a model with larger windows also has less samples to wrongly classify. Shorter window sizes give better precision for *jumping* and *running backward*, while the model with a 5-second window either outperforms or has a competitive precision compared to the shorter window models for the remaining vigorous activities.

Even though the smaller window sizes give lower subject-wise accuracy in most cases, the models with smaller window sizes manage to detect certain activities which other models do not. This is a result of the training data not having these activities because of the majority voting explained in section 5.1. The 5-second window model could suffer from the majority class problem explained in the same section. Models using larger windows are not able to learn an activity such as *jumping*, since the activity will never be present in the training data using these window sizes. This can be seen in the different models' precision for *jumping*, where the 5-second window model does not detect the activity at all and the 1-second window model clearly outperforms the

3-second window model.

The Importance of Good Training Data

As stated at the end of section 2.3, movement patterns differ for people in different age groups. Since the training data consisted of subjects in the age groups young adolescents from TAH and LOC, the movements from young adults in the RBF dataset could be problematic for the model. So the improvements shown in the mix-in classifier's results should not be taken as conclusive in the sense of the models improvement on the original data. The results do however show that the model is able to adapt to new data if it is given appropriate training data.

7.1.1 Discussion

For research question 2.1, the answer seems to be to use smaller windows when training the machine learning models with the current functionality in the HAR Framework. The models using long window sizes with the current sliding window approach are not able to detect short activities, which includes most of the vigorous activities. The 5-second window model could not classify *jumping*, and did not perform well on other short activities.

The classifier uses the majority class in a given window as the ground truth when creating features from the training data. As for research question 2.1, there seems to be some problems using this basic approach, since it favours more persistent activities in the data. This could be fixed by using smaller window sizes, but this seems to affect the overall precision of the model, since the model gets less data for each prediction.

The majority class data segmentation could make noise in the data, since a label in the window could be the majority by just a few sample, leaving large parts of the window as irrelevant noise. This could impact the purity of the training data, which again could impact the model's performance. This is of extra concern when handling vigorous data since the vigorous activities have a shorter duration than other activities in the training data, as shown in section 5.1. This section also shows both the problems and advantages with smaller window sizes in more detail.

7.2 XGBoost for Vigorous Activity Detection

The results from the training of a XGBoost model on 3-second window training data can be seen in section 6.2. The first confusion matrix containing activities in the results shows that the model has a high precision for daily living activities, such as *walking*, *standing* and *sitting*. *Running* is the vigorous activity with the highest precision in the model's predictions. The matrix also shows that certain vigorous activities are hard to tell apart. The model confuse most vigorous activities with *running forward*. This includes *jumping*, *running backward* and *crabwalk* to some degree. These activities are less present in the training dataset, which can be seen in section 5.1.

The 3-second window model does however separate vigorous data from non-vigorous data with a precision of 95.56% and a recall of 95.38%. This gives a F1-score of 95.40%. The training data consists of young people, or more precisely adolescents. Since the age and demographic for the UngHUNT data is similar to the training data, the model should also be able to separate the vigorous data found in UngHUNT reasonable well, based on the previous results.

Results from predictions on the UngHUNT dataset can be seen in section 6.5. In figure 6.9 the model detects a period of vigorous physical activity in the middle of the day. This can also be seen in figure 6.10 and 6.11, where the model detects dense periods of vigorous activity for the subject. Given the time and date of the collected data being on a weekday at 12:00, this could be an example of a student's physical education lecture or any other mid day physical activity. Either way, the model clearly picks up vigorous activity at this period, which fits the accelerometer signal. For other periods of the day the model predicted moderate amounts of vigorous activity, which in turn also fits the accelerometer data for these periods of the day.

7.2.1 Feature Selection

As shown in section 6.3.1 the models do not improve much after adding about 20 features during the feature selection. The 3-second window model reached a global maximum at about 20-30 features before stabilising just below the maximum. The 2-second window model competes with the 3-second window model by it's F1-score in the experiment, but seems to perform worse when more features are added to the model. Both the 2- and 3-second window model perform the best at about 20 to 30 features.

The thigh-back sensor combination is important in the FS-MODEL, since there is an equal distribution of back and thigh features in the 22 features selected in the model. Some of the most important features do also use the correlation between the two sensors' axes. X- and z- orientations are in this case the most important orientations on the sensors. Time domain features are used the most, but frequency domain features are also important in the model.

7.2.2 Mix-In Classification

The results in section 6.4 show that the model adapts well and improves with the new data. From the mix-in classifier's results one can see a 10% increase of correctly classified windows of *running backward* in the data. *Walking* also had an increase in precision with the mix-in model, while both *running forward* and *standing* performed slightly worse after mixing the data into the model.

7.2.3 Discussion

The literature reviewed in section 3 did not use the XGBoost classifier. This literature search was performed answer the first research question mentioned in this thesis, namely research question 1.1. The most used machine learning algorithms were SVM, K-NN and CNN. The XGBoost algorithm was not found in any of the papers focusing on vigorous physical activity. This algorithm is tested in this study to evaluate it's performance classifying such activities, since it has had good performances in previous HAR work at NTNU. The thigh-back sensor combination also had good performances in the papers found in the literature search, which is the baseline for data collection in this study.

The performance of XGBoost in this project is mixed. The vigorous models all struggle to classify short activities such as *running backward*, *running forward* and *jumping*. In the sense of research question 3.1, the models have high precision classifying everyday living activities. The models do however make mistakes for for multiple short and vigorous activities. The 3-second window model does well when isolating vigorous activities from every day living activities. This model mostly confuses vigorous activities with other vigorous activities, which makes the model work well for it's purpose in this thesis. This answers research question 3.3. The classification problem however needs to be addressed, whether the data is the problem, or the algorithm. This will be further discussed in chapter 8. The model is strict in classification of vigorous activity, which fits the nature of the training data which the algorithm got, containing data from adolescents playing handball and running through an obstacle course. With these results in mind one could argue that XGBoost performs well, also for vigorous activity detection.

For research question 3.2, the vigorous activity classifier needs at least 22 features to reach the same performance as the model using all 95 features. The most important orientation is the sensors' x-axis together with the z-axis, while the y-axis is less important. Time domain features are more important than frequency domain features, while features from both sensors are useful in this model. The thigh-back combination in the data collection is important, since the sensors are both important in the selected features.

Chapter 8

Conclusion and Future Work

This chapter concludes upon the research questions and goals set at the start of this thesis. The goals are met by reviewing related work in the field, together with machine learning experiments. Recommended future work from this study is also discussed and explained in this chapter.

8.1 Conclusion

As a result from our study we have developed a machine learning algorithm that allows researchers to classify vigorous activities in objective measurements through tri-axial acceleromenters. This will lead to a better understanding people's health through vigorous physical activity detection in existing (HUNT4) and newly conducted data collections[1]. The main method developed in this work consists of data segmentation analysis and ensemble methods in gradient boosted trees to distinguish vigorous from non-vigorous activity in accelerometer signals from sensors placed on the thigh and lower back. Three goals were set at the start of this thesis to accomplish the aim of the study, which was to detect vigorous activity in the HUNT4 dataset.

The first goal (Goal 1) of this thesis was to research and describe existing machine learning approaches for vigorous activity recognition in HAR datasets. The literature study reveals that several approaches have been tested in previous work, but there is a lack of studies on vigorous physical activity detection with the Axivity AX3 accelerometers. XGBoost was not used in any of the vigorous activity studies found in the literature search, but is documented in this thesis. There is still work to do within the vigorous physical activity field of HAR studies.

In the beginning of this work, we defined a set of research questions, which can be

[1] The ProPASS consortium is an initiative that aims at combining existing datasets which have the same sensor setup as used in this thesis (https://www.propassconsortium.org)

seen in chapter 1. Goal 2 and 3 in this thesis focus on the machine learning aspects of the task and the windowing of training data. Goal 2 was met through experimentation with the window sizes in the first experiment. The suitable window size found was windows smaller than five seconds, where the 1-second window model performed best for the shortest vigorous activities. Goal 3 was to create and evaluate a machine learning model for vigorous activity, which was completed through various experiments throughout the study. The 3-second window model classifies everyday living activities with a high precision, while it confuse vigorous activities. The model do however reaches a F1-score of 95.40% when detecting general vigorous activity, while it only needs 22 features to perform at the same level as the full 95 feature model.

The overall aim of this thesis was to detect periods of vigorous activity in HUNT4 data. This aim was met with the 3-second window XGBoost classifier separating vigorous from non-vigorous activity in the UngHUNT dataset, which is a subset of the HUNT4 dataset. The vigorous model created is embedded in the HAR Framework and hence can run as part of the HUNT data analysis. This makes the results of this thesis accessible for future studies.

8.2 Contributions

From our research goals and questions, which can be seen in chapter 1 we have made four main contributions. The first and most important contribution is a machine learning classifier using the XGBoost algorithm to distinguish vigorous from non-vigorous movement patterns in accelerometer data. The classifier succeeded with a F1-score of 95.40% and managed to detect vigorous periods in the HUNT4 data.

Our second contribution is the research done on window sizes in HAR work using vigorous accelerometer data. During our work we contributed with insight into suitable window sizes for vigorous activity detection in long-term HAR datasets.

An XGBoost classifier for vigorous activity detection was also created by using data from two body worn Axivity AX3 accelerometers. This novel approach that was not tested in the studies found in the previously performed literature review. The classifier separates vigorous activities from non-vigorous activities well, but work still needs to be done on the individual activity classification.

As our final contribution, a training dataset was created and used to train a vigorous activity classifier. This dataset consists of the TAH and LOC dataset and is embedded into the HAR Framework for use in future studies.

8.3 Future Work

As mentioned in chapter 7, vigorous physical activity HAR has not been researched a lot so far and the work presented in this thesis opens for more research in the field. This section proposes a few next steps in terms of vigorous HAR and focus on training data, and machine learning algorithms.

8.3.1 Data

The training data for a machine learning algorithm is the key for the algorithm to be able to classify data with high precision. With the inclusion of the TAH dataset in the training data, the models created during this study got out-of-lab data which proved difficult to learn. The mix-one-in model proved that the model was able to improve on new data, given relevant training data. Gathering tailored data for particular activities which is not sufficiently available in the existing datasets is a good approach to improve the classifier. Hence the model could improve further and this is shown in the mix-in models' increased performance for *running backward*. The in-lab dataset needs to be further researched though, since the mistakes made for *walking* did not show up during the initial LOOCV with the XGBoost model using the original training dataset.

Also, post-processing should be applied on the final machine learning results, to have the windows make sense for public health researchers. For public health research, vigorous physical activity windows of 1-5 seconds might not make sense. This needs to be addressed and post-processing of the results applied if found necessary.

It would also be beneficial to include the automatic synchronisation in the HAR Framework, which would make work with new datasets a lot faster. This would simplify the work by automatically synchronizing the labels with the accelerometer data, instead of doing this manually.

Some labels were cut from the confusion matrix shown in section 6.2. These labels were cut for either being uninteresting for the results in this report, or for having just a few entries in the dataset. Most of these labels could probably be cut from the training data to remove potential noise in future work.

Hay [2019] recently used body worn sensors to recognize sleep-wake patterns using accelerometer data. In her thesis she mentions a potential next step of including temperature from the sensor data, which is a feature in the Axivity AX3 sensors. During longer periods of vigorous activity the body, and in particular the muscle where the sensors are mounted, gets warmer. If the further focus on vigorous activity recognition is outside everyday living, targeting longer periods like for example a match of handball, then temperature could play a big part in increasing the overall F1-scores for the XGBoost model presented in this thesis.

Dynamic Windows

The current windowing approach has some disadvantages for vigorous activity. A solution to this could be to include some kind of weighting of activities when deciding ground truth for a window, rather than using a majority vote. This would give shorter activities a better chance with the current data segmentation implementation. Another approach would be to implement a new binary classifier, by reformatting the training data presented in this thesis to two ground truths, *vigorous* and *non-vigorous*. By doing this, short vigorous activities close to other short vigorous activities would have a better chance of being detected when majority voting is performed, since both activities are read as *vigorous*.

Hessen and Tessem [2016] previously tested approaches using dynamic windows instead of static window sizes. This study used static window sizes in the sliding window approach. A logical next step would be to implement a dynamic window in the HAR Framework for data segmentation. This would make the data include every label in windows with different sizes, fixing the issue created by using majority vote with as short activities as presented in this study.

In their study, Hessen and Tessem [2016] also used a voting classifier. This classifier contained different machine learning models with different window sizes which selected the prediction from the classifier with the highest confidentiality in the prediction. This could be useful for an eventual combination of vigorous and every day living classifier, since the vigorous classifier does not have *lying* or *sleep* in the training data, hence it does not predict these labels.

8.3.2 Machine Learning Algorithm

This thesis only covered a single algorithm's performance on adolescents data. This could be broadened to also include adults data, with experiments using multiple algorithms to measure their performance towards each other. Notable mentions for machine learning algorithms would be the following:

- CNN
- SVM
- K-NN

Which was found in the previously performed literature review. These algorithms displayed great results in their respective papers where specific vigorous activities were classified, such as badminton moves. These algorithms still need to be measured in general vigorous activity detection using body worn accelerometers.

Bibliography

Bartlett, R. (2007). *Introduction to sports biomechanics: Analysing human movement patterns*. Routledge.

Breiman, L. (1996a). Bagging predictors. *Machine Learning*, page 123–140.

Breiman, L. (1996b). Bias, variance, and arcing classifiers. *Technical Report 460, Department of Statistics, University of California, Berkeley, CA.*, page 123–140.

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

Bulling, A., Blanke, U., and Schiele, B. (2013). A tutorial on human activity recognition using body-worn inertial sensors. *ACM Computing Surveys*, 46.

Bårdstu, H. B. (2016). Detection of physical activity types with accelerometers in adolescents during semistructured free-living. Master's thesis, Norwegian University of Science and Technology.

Freund, Y. and Schapire, R. E. (1996). Experiments with a new boosting algorithm. *Proc. 13th International Conference on Machine Learning*.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Ann. Statist.*, 29(5):1189–1232.

Garcia, D. C. (2019). Sampling rate comparison in accelerometer based human activity recognition. Master's thesis, Norwegian University of Science and Technology.

Hay, A. (2019). Machine learning methods for sleep-wake classification using two body-worn accelerometers. Master's thesis, Norwegian University of Science and Technology.

Hedayatrad, L., Stewart, T., and Duncan, S. (01 Mar. 2021). Concurrent validity of actigraph gt3x+ and axivity ax3 accelerometers for estimating physical activity and sedentary behavior. *Journal for the Measurement of Physical Behaviour*, 4(1):1 – 8.

Hessen, H.-O. and Tessem, A. J. (2016). Human activity recognition with two body-worn accelerometer sensors. Master's thesis, Norwegian University of Science and Technology.

Hong, J.-H., Ramos, J., and Dey, A. K. (2016). Toward personalized activity recognition systems with a semipopulation approach. *IEEE Transactions on Human-Machine Systems*, 46(1):101–112.

Hung, A., Chen, J., and Gill, I. (2018). Automated performance metrics and machine learning algorithms to measure surgeon performance and anticipate clinical outcomes in robotic surgery. *JAMA Surgery*, 153.

Khan, A., Nicholson, J., and Plötz, T. (2017). Activity recognition for quality assessment of batting shots in cricket using a hierarchical representation. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 1(3).

Kohavi, R. and Kunz, C. (1997). Option decision trees with majority votes. *Proceedings of the Fourteenth International Conference on Machine Learning*.

Kongsvold, A. M. (2016). Validation of the ax3 accelerometer for detection of common daily activties and postures. Master's thesis, Norwegian University of Science and Technology.

Lara, O. D. and Labrador, M. A. (2013). A survey on human activity recognition using wearable sensors. *IEEE Communications Surveys Tutorials*, 15(3):1192–1209.

Mitchell, T. M. (1997). *Machine Learning*. MIT Press and The McGraw-Hill Companies, Inc.

Narayanan, A., Stewart, T., and Mackay, L. (2020). A dual-accelerometer system for detecting human movement in a free-living environment. *Medicine & Science in Sports & Exercise*, 52(1).

Ramakrishnan, R., Doherty, A., Smith-Byrne, K., Rahimi, K., Bennett, D., Woodward, M., Walmsley, R., and Dwyer, T. (2021). Accelerometer measured physical activity and the incidence of cardiovascular disease: Evidence from the uk biobank cohort study. *PLOS Medicine*, 18:e1003487.

Reinsve, Ø. (2018). Data analytics for hunt: Recognition of physical activity on sensor data streams. Master's thesis, Norwegian University of Science and Technology.

Sammut, C. and Webb, G. I. (2017). *Encyclopedia of Machine Learning and Data Mining*. Springer US, Boston, MA.

Sani, S., Massie, S., Wiratunga, N., and Cooper, K. (2017). Learning deep and shallow features for human activity recognition. In Li, G., Ge, Y., Zhang, Z., Jin, Z., and Blumenstein, M., editors, *Knowledge Science, Engineering and Management*, pages 469–482, Cham. Springer International Publishing.

Sani, S., Wiratunga, N., Massie, S., and Cooper, K. (2018). Personalised human activity recognition using matching networks. In Cox, M. T., Funk, P., and Begum, S., editors, *Case-Based Reasoning Research and Development*, pages 339–353, Cham. Springer International Publishing.

Small, S. R., Khalid, S., Dhiman, P., Chan, S., Jackson, D., Doherty, A. R., and Price, A. J. (2020). Impact of reduced sampling rate on accelerometer-based physical activity monitoring and machine learning activity classification. *medRxiv*.

Stamatakis, E., Huang, B.-H., Maher, C., Thøgersen-Ntoumani, C., Stathi, A., Dempsey, P. C., Johnson, N., Holtermann, A., Chau, J. Y., Sherrington, C., Daley, A. J., Hamer, M., Murphy, M. H., Tudor-Locke, C., and Gibala, M. J. (2021). Untapping the health enhancing potential of vigorous intermittent lifestyle physical activity (vilpa): Rationale, scoping review, and a 4-pillar research framework. *Sports Medicine*, 51(1):1–10.

Steels, T., Van Herbruggen, B., Fontaine, J., De Pessemier, T., Plets, D., and De Poorter, E. (2020). Badminton activity recognition using accelerometer data. *Sensors*.

Trost, S. G., Loprinzi, P. D., Moore, R., and Pfeiffer, K. A. (2011). Comparison of accelerometer cut points for predicting activity intensity in youth. *Med Sci Sports Exerc*.

Vågeskår, E. (2017). Activity recognition for stroke patients. Master's thesis, Norwegian University of Science and Technology.

Widianto, A., Sugiarto, T., Lin, Y.-J., Lee, Y.-H., and Hsu, W.-C. (2019). Physical activity intensity classification using a convolutional neural network and wearable accelerometer.

Appendix A

Data Streams

A.1 Accelerometer data for single labels

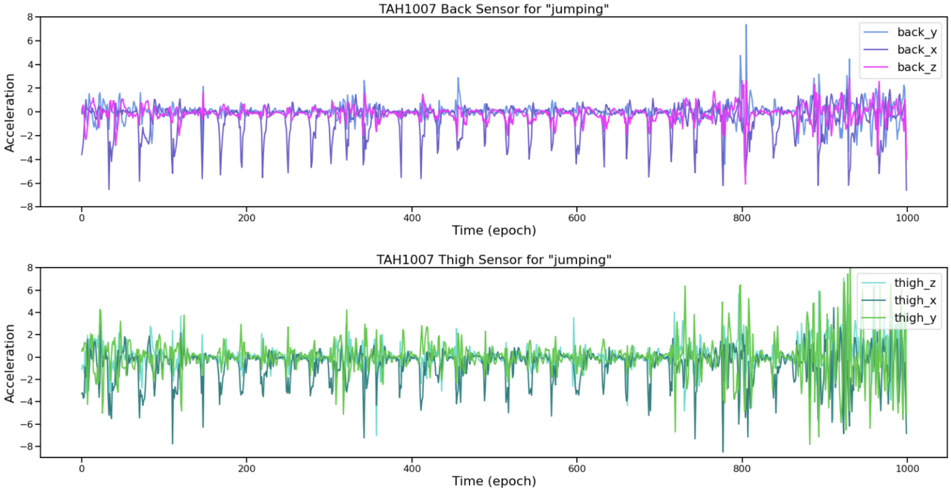


Figure A.1: Signal from subject TAH1007 jumping.

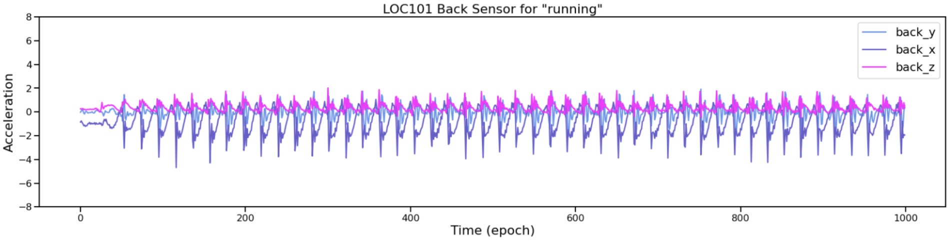
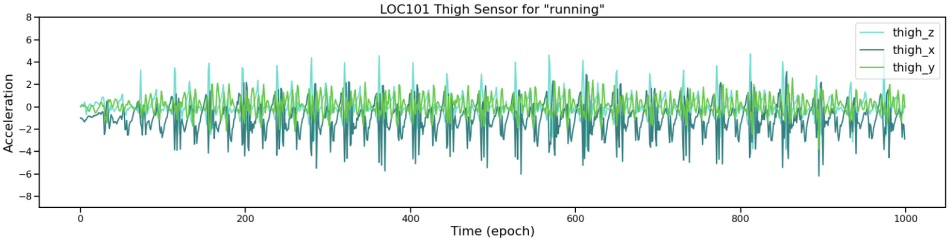



Figure A.2: Signal from subject LOC101 running.

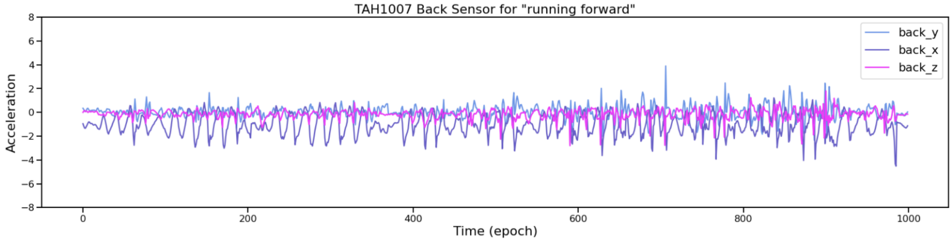
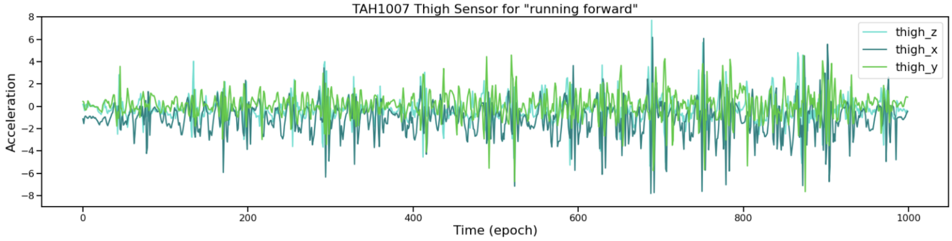



Figure A.3: Signal from subject TAH1007 running forward.

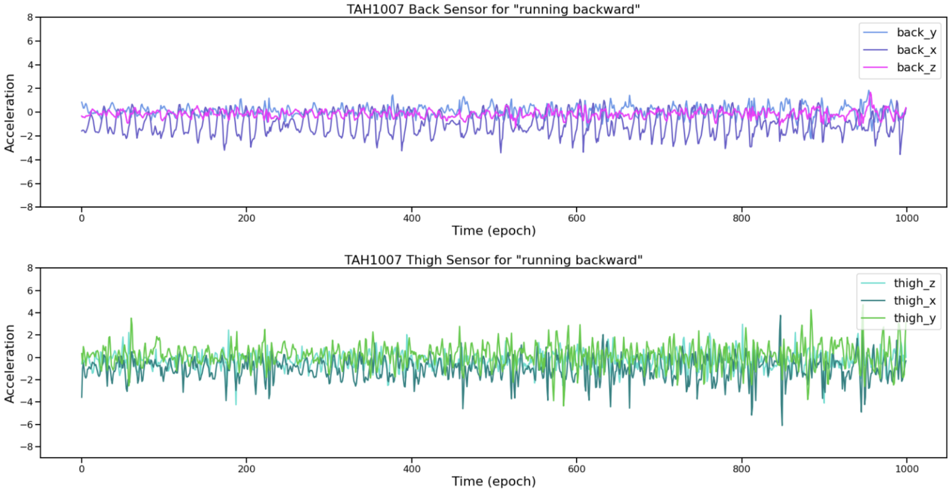


Figure A.4: Signal from subject TAH1007 running backwards.

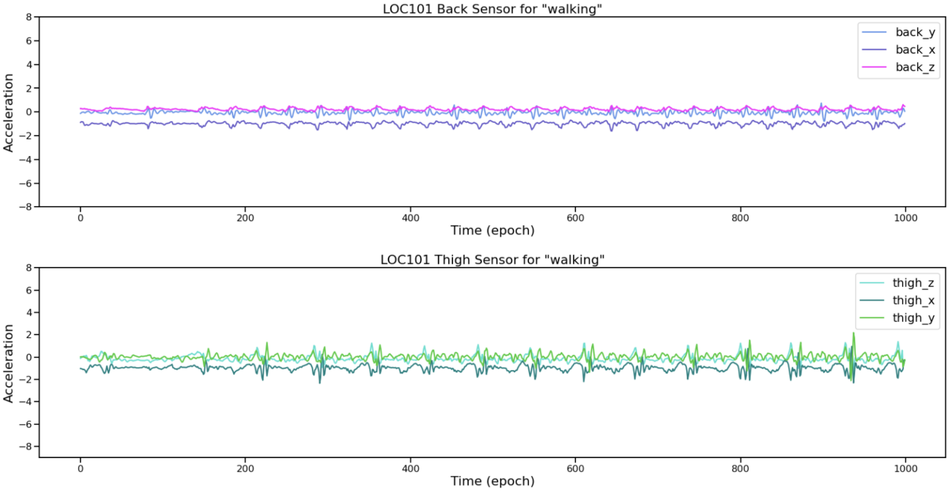


Figure A.5: Signal from subject LOC101 walking.

Appendix B

Literature Search Quality Assessments

B.1 Quality Assessments

Criteria identification	Criteria
IC1	The study's main concern is Human Activity Recognition.
IC2	The study focuses on solving a machine learning problem.
IC3	The study uses the relevant Axivity sensor.
IC4	The study uses vigorous data to some degree.
QC1	Is there a clear statement of the aim of the research?
QC2	Is the study put into context of other studies and research?
QC3	Are system or algorithmic design decisions justified?
QC4	Is the test data, study algorithm and experimental setup reproducible?
QC5	Is it clearly stated in the study which other algorithms the study's algorithm(s) have been compared with?
QC6	Are the performance metrics used in the study explained and justified?
QC7	Are the test results thoroughly analysed?

Table B.1: The criteria used to select studies for the final quality assessments. IC stands for inclusion criteria and QC for quality criteria.

Appendix C

Results

C.1 Standard Model Confusion Matrix on Training Data

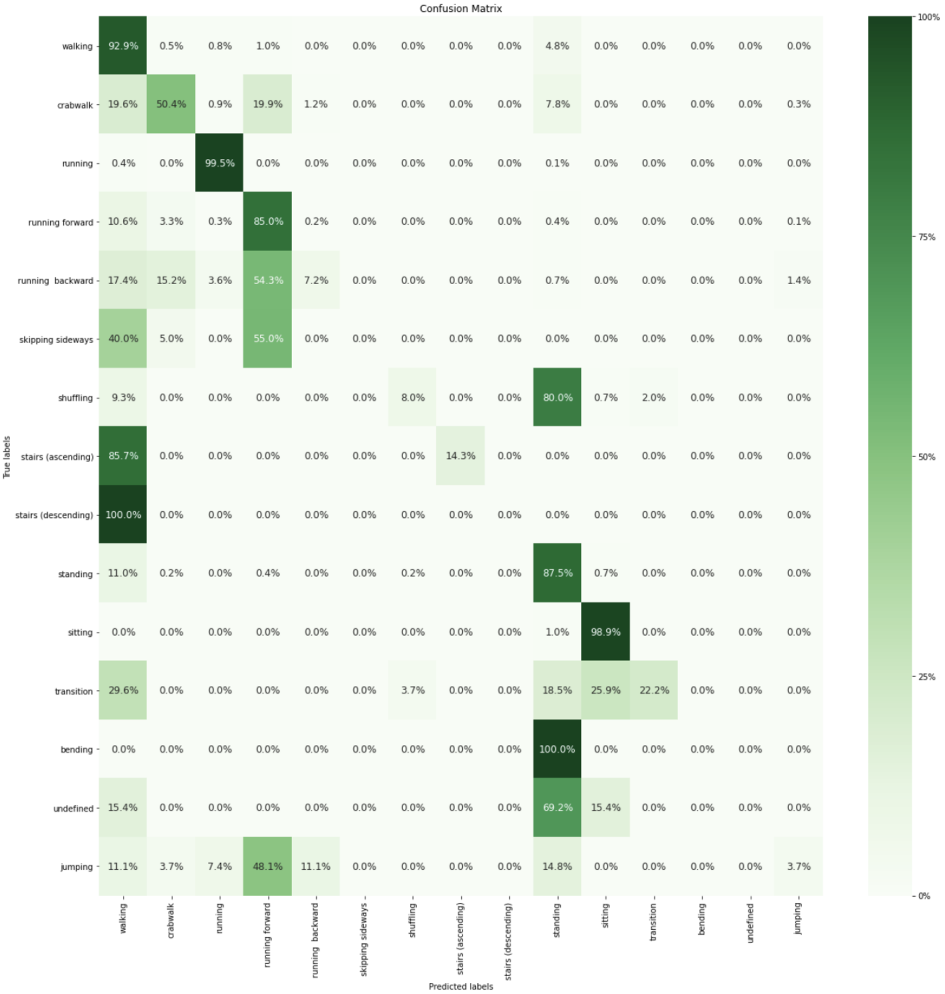


Figure C.1: The confusion matrix from the standard models LOOCV run on the training data.

C.2 Predictions on the RBF Dataset

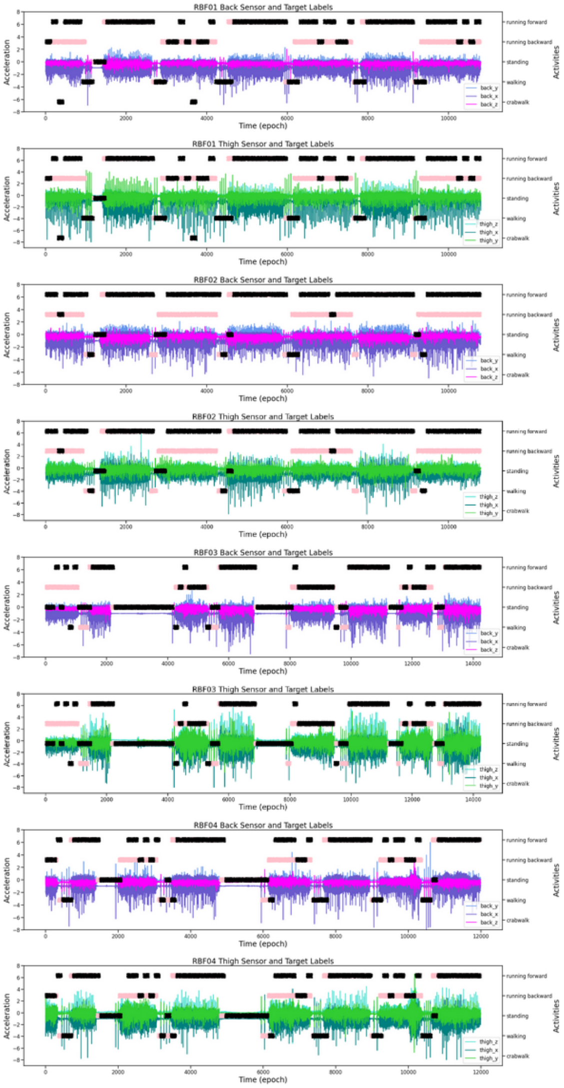


Figure C.2: Comparison of ground truths (pink) and predictions (black) with the sensor data in the background for subject RBF01, RBF02, RBF03 and RBF04 in the RBF dataset. The predictions were created by FS-MODEL. Accelerometer data from both sensors can be seen in the background.

C.3 Feature Importances

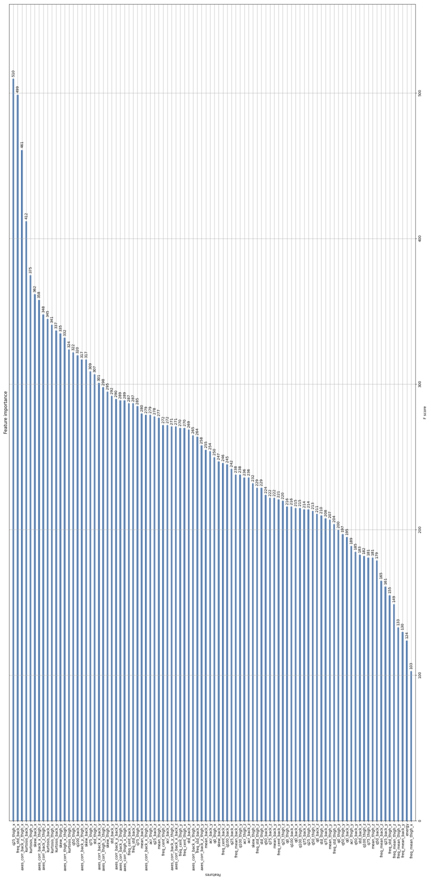


Figure C.3: Number of times a feature is used to split a node in the decision trees in the 3-second vigorous activity classifier.

NTNU

Norwegian University of
Science and Technology