FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

# Artificial Intelligence for recognizing school and training sports activities from sensorised vests

**Francisco Maria Ferreira de Melo de Sá Couto**

Mestrado em Bioengenharia

Supervisor: Prof. Miguel Fernando Paiva Velhote Correia

Second Supervisor: Dr. Rafael Castro Aguiar

July 28, 2025

# Artificial Intelligence for recognizing school and training sports activities from sensorised vests

**Francisco Maria Ferreira de Melo de Sá Couto**

Mestrado em Bioengenharia

Approved in oral examination by the committee:

President: Prof. Joaquim Gabriel Magalhães Mendes

Referee: Prof. Miguel Fernando Paiva Velhote Correia

Referee: Prof. Cristina Manuela Peixoto dos Santos

Assinado por: **Francisco Maria Ferreira de Melo de Sá Couto**
Num. de Identificação: 14629462
Data: 2025.07.28 17:29:05+01'00'

July 28, 2025

# Resumo

O Reconhecimento de Atividade Humana (HAR) é uma área de estudo com crescente importância, impactando diferentes setores como saúde, desporto e educação. Esta foca-se sobretudo em acompanhar e avaliar movimentos do corpo de forma autónoma. O foco do presente trabalho recai precisamente no âmbito do HAR, tendo como principais objetivos a avaliação da performance de diferentes modelos de *machine learning*, bem com a classificação de movimentos primários relativos a atividades de desporto. Para tal, é utilizada uma unidade de medida inercial (IMU) composta por um acelerómetro e um giroscópio. Este projeto está integrado na solução do TexP@ct, um colete inteligente capaz de quantificar métricas para aferir a performance de estudantes através de um único IMU e um sensor de ECG na região superior do tronco. De forma a recorrer-se a apenas um sensor localizado nesta região, é necessário que toda a informação relevante para o reconhecimento motor consiga ser captada, aspeto que se torna particularmente importante para movimentos realizados por extremidades do corpo.

Esta dissertação avalia diferentes metodologias descritas na literatura para a construção de uma *pipeline* de processamento. De forma a desenvolver os algoritmos de classificação, foram utilizadas duas bases de dados publicas: PAMAP2 e RWHAR. Ambas as bibliotecas apresentam dados de um IMU na região do tronco com atividades fundamentais como andar, saltar, e correr. Na fase de pré-processamento, foram utilizadas técnicas para a remoção da gravidade e filtragem de sinal. Depois, foram implementadas dois algoritmos de extração de parâmetros. Foi estabelecido um modelo base de referêcia para análises posteriores. Primeiramente, foram testadas três alternativas de redução de dimensionalidade de parâmetros: *PCA, RFE e SelectKBest*. Foi ainda realizada a aprimoração de híper parâmetros nos 5 melhores modelos, utilizando diferentes configurações de otimização de performance.

De todos os modelos testados (*LightGBM, GBC, KNN, RF, ExtraTrees*), o melhor classificador foi o *LightGBM* com exatidão de 94.24% para a biblioteca PAMAP2 e 90.92% para a RWHAR. As técnicas de redução de dimensão de parâmetros, mais concretamente a *SelectKBest*, não só permitiu aumentar a performance do algoritmo, como também diminuiu os custos computacionais. Para além disso, a aprimoração de parâmetros permitiu obter ganhos adicionais de performance. Curiosamente, para ambas as bibliotecas, os híper parâmetros com melhor classificação foram idênticos, permitindo concluir que este modelo pode ser generalizado para outras bases de dados, no âmbito do HAR.

Em sumário, os resultados demonstram que, mesmo com a limitação do uso de apenas um IMU colocado no tronco, é possível atingir classificações com elevada exatidão para movimentos fundamentais dentro do desporto. Esta metodologia valida o potencial da integração destes algoritmos na plataforma da solução do TexP@ct, e aplicações relacionadas com saúde e reabilitação, assim como avaliação de performance profissional desportiva.

# Abstract

Human Activity Recognition (HAR) is a research field of growing importance, positively impacting the health, sports and education sectors, with the intent of monitoring and evaluating movement and motion patterns in an autonomous manner. The presented study primarily aims to evaluate the performance of different machine learning models, to classify fundamental movements within a sports' context. To do so, data from a single inertial measurement unit (IMU), composed of an accelerometer and gyroscope, is used. This project is integrated in the TexP@ct solution, a sensorised smart vest capable of extracting quantifiable metrics for evaluating the performance of students from a single IMU sensor and ECG in the upper torso. The use of a single sensor located in this region of the body requires that all relevant information for recognizing activities, including those involving the extremities, be effectively extracted.

This study evaluates different methodologies explored in the literature review to assemble a pipeline of movement classification. For the development of classification algorithms, two public datasets were procured: PAMAP2 and RWHAR. Both these datasets have a sensor in the upper torso region, with activities such as running, jumping, and walking. In the pre-processing stage, gravity removal and noise reduction was performed contributing to a smoother signal. After this, the dataset was divided in two distinct dataframes: train and test. Furthermore, two different feature extraction techniques were implemented. A baseline classification model was implemented for comparison purposes and three dimensionality reduction techniques were tested: PCA, RFE, and SelectKBest. After this, hyperparameter tuning was conducted on the 5 best classifiers, using different configurations in order to improve performance.

Among the tested models (LightGBM, GBC, KNN, RF, ExtraTrees) the best classifier was LightGBM, achieving accuracies of 94.24% for the PAMAP2 dataset and 90.92% for the RWHAR dataset. Dimensionality reduction techniques, particularly SelectKBest algorithm, not only allowed the model to improve its performance for both datasets but also reduced the amount of features for extraction, lowering the computational costs for the model. Additionally, the hyperparameter tuning achieved better results by evaluating a predefined set of configurations. Interestingly, the best result for both datasets had the same parameters, concluding this model can be generalised for other HAR datasets.

The results demonstrate that even with the limitations of a single IMU positioned on the upper torso, it is possible to achieve high classification accuracy for fundamental sports movements. This approach validates the potential of this pipeline for the integration in educational platforms like TexP@ct, and broader applications such as rehabilitation, professional sports performance, and other health related areas.

# UN Sustainable Development Goals

The United Nations Sustainable Development Goals (SDGs) provide a global framework to achieve a better and more sustainable future for all. It includes 17 goals to address the world's most pressing challenges, including poverty, inequality, climate change, environmental degradation, peace, and justice.

The specific Sustainable Development Goals mentioned have the following names:

**SDG 3** Ensure healthy lives and promote well-being for all at all ages

**SDG 4** Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all

| **SGD** | **Target** | **Contribution** | **Performance Indicators and Metrics** |
|---|---|---|---|
| 3 | 3.d | Increase sports analysis allowing for accurate sports activity classification improving general health and encouraging the practice of sports | Classification of fundamental movements in sports |
| 4 | 4.a | Allows PE teachers to have quantifiable sports metrics for education | Quantifiable instances of activity |

# Acknowledgements

Firstly, I would like to thank my supervisor Miguel Velhote Correia for the opportunity to integrate the TextP@ct consortium. To my co-supervisor Rafael Castro Aguiar for the daily guidance and unwavering support throughout the entirety of the project.

Additionally, I would like to thank my friends and colleagues for their constant support, encouragement and belief in me.

Lastly, my family and girlfriend for always being there for me and allowing me to pursue my work, supporting me in every challenge I face.

This work is co-financed by Component 5 - Capitalization and Business Innovation, integrated in the Resilience Dimension of the Recovery and Resilience Plan within the scope of the Recovery and Resilience Mechanism (MRR) of the European Union (EU), framed in the Next Generation EU, for the period 2021 - 2026, within project TEXP@CT, with reference 61.

Francisco Sá Couto

*"Engenheiro é aquele que tem engenho"*

Eng. Edgar Cardoso

# Contents

# List of Figures

# List of Tables

# List of Acronyms

| | |
|---|---|
| ADL | Activities of Daily Living |
| AI | Artificial Intelligence |
| DNN | Deep Neural Network |
| DT | Decision Tree |
| ECG | Electrocardiogram |
| ET | Extra Trees |
| GBC | Gradient Boosting Classifier |
| HAR | Human Activity Recognition |
| IMU | Inertial Measurement Unit |
| KNN | K-Nearest Neighbors |
| LightGBM | Light Gradient Boosting Machine |
| ML | Machine Learning |
| PCA | Principal Component Analysis |
| PE | Physical Education |
| RF | Random Forest |
| RFE | Recursive Feature Elimination |
| SMO | Sequential Minimal Optimization |
| SVM | Support Vector Machine |

# Chapter 1

# Introduction

## 1.1 Context and motivation

Human Activity Recognition, within the context of healthcare, rehabilitation and sports performance assessments, has steadily increased in relevance and research in the last few decades. The ability to automatically identify human activities opens the door to intelligent systems capable of providing personalized feedback with impact in quality of life, injury prevention and dedicated personnel monitoring.

Usually this system of automatic detection of HAR uses multiple sensors placed in strategic body parts to build a kinematic representation of body segments, like head, chest, pelvic region, arms and legs. This approach can simultaneously capture various movements, making it a precise method for sensor data acquisition. However, the usage of multiple sensors comes with other downsides like the complexity of data capturing, uncomfortable and impractical full suits to encompass the sensors and higher implementation costs.

With the advancements in inertial sensor technology and the development of artificial intelligence, mainly machine learning algorithms, it has been possible to explore simpler and more effective ways to identify human movement in everyday scenarios. These advances enabled the identification of sports related metrics with a reduced number of sensors, while still maintaining good accuracy and analytical performance. This motivate the research for solutions that integrate a single sensor configuration with traditional machine learning models to investigate fundamental movements within sports and physical activity assessment. Additionally, this study is conducted within the TexP@ct project: a vest with an IMU to aid physical exercise (PE) professors by calculating live metrics and make post-exercise reports with movement patterns and performance metrics.

## 1.2 Objectives and research questions

The objective of this work is to evaluate the performance of different classification algorithms based on Machine Learning (ML) models for HAR using IMU data from a single sensor placed around the chest/ upper back area. The sensor module includes accelerometer, gyroscope and electrocardiogram sensors with the intention of identifying daily living activities and fundamental movements that incorporate sports activities. It is considered a good performance when achieved an accuracy of above 85%.

Taking the presented challenge and sensor solution into account, the following research questions were formulated. The primary research question is:

- **Is it possible to achieve an accuracy above 85% for the classification of different human activities utilizing a single IMU sensor, placed on the chest/upper back?**

In addition, this study seeks to answer the following secondary questions:

- What type of ML models provide better performance in activity recognition with this sensor configuration?
- How effective are pre-processing techniques like dimensionality reduction and hyperparameter tuning to improve model classification?

## 1.3 Document Outline

This study is organized by chapters in a sequential structure. Each chapter presents an important component to the research, from a theoretical component to the methods presented and discussion of results. This proposed structured aims to facilitate the reader to understand the investigation conducted.

Chapter 2 presents the fundamental scientific and technological concepts for HAR. In this section the primary sensors that make up the sensor module are presented. The fundamental sensors to aid this research project are explored herein, detailing how accelerometers, gyroscopes and ECG sensors acquire relevant data. Additionally, it is also relevant to address the purpose of artificial intelligence with historic background and primary machine learning models used in this research, like Random Forest, Extra Trees, Gradient Boosting Classifier, LightGBM and K-Nearest Neighbour.

Chapter 3 presents the state of the art in HAR classifications using IMU sensors. The recent literature is explored based on a systematic review using the PRISMA methodology. The primary datasets, pre-processing techniques, feature selection and integration of data with machine learning models are all presented in this review. Additionally, the current commercial smart vest solution and the TexP@ct solution are presented and discussed.

Chapter 4 details the methodology proposed in this research. It encompasses the preparation of the PAMAP2 and RWHAR datasets, pre-processing, different feature extraction techniques using sliding windows, baseline model selection, feature selection algorithms for dimensionality reduction purposes, hyperparameter tuning and performance metrics for evaluation.

Chapter 5 presents the results obtained at each step of the methodology. Several analyses are done on the effects of filtering the signals, the performance of primary ML models, the gains obtained in feature reduction techniques and hyperparameter tuning and the evaluation of the final model with the discussed performance metrics and confusion matrices.

Chapter 6 presents the discussion and interpretation of the obtained results. Discussion points are conducted on the purpose of the full methodology, the efficacy of the performance enhancement techniques, the interpretability of the final model, comparison with state of the art models and the implications for real world scenarios with the context of integrating it with TexP@ct platform.

Finally, Chapter 7 presents the conclusion, by discussing the primary contributes of this research, highlighting the limitations and future work in order to have a more scalable and robust algorithm and to encompass other population targets.

# Chapter 2

# Theoretical Concepts

This chapter outlines the fundamental theoretical concepts pertinent to the advancement of HAR systems utilizing signals and artificial intelligence. To develop effective and accurate classification systems, a solid understanding of physiological and technological concepts is fundamental for enabling HAR. This chapter starts with an overview of the electrocardiogram (ECG), a non-invasive method for capturing heart activity that plays a crucial role in detecting heart rate and physiological states. The Inertial Measurement Unit (IMU), composed of accelerometers and gyroscopes, as a core sensor is discussed for capturing motion data. Finally, different artificial intelligence techniques, ranging from classical machine learning models like Random Forests and K-Nearest Neighbors to more advanced methods such as Gradient Boosting Machines and LightGBM are introduced, outlining their relevance and application in activity classification tasks. Together, these primary components form the basis for the approach explored in this project.

## 2.1 Electrocardiogram (ECG)

The Electrocardiogram is a test to detect the physiological signal of the heart's electrical activity, translating into a waveform that can be analysed. This medical exam is used to identify abnormal heart function, monitor heart rhythm and test physical activity strain [1].

The ECG waveform can be decomposed into different sections depending on the recorded part of the heart: It starts with the P wave that represents the depolarisation of the atria, then the QRS complex represents the depolarisation of the ventricles and the T wave represents the repolarisation of the ventricles, while repolarisation of the atria is normally masked by the T wave leading to no discernible wave form. [2, 3].

The ECG signal typically has a small amplitude ranging from 10 $\mu$V to 5-mV [5]. This signal is highly susceptible to noise sources, as poor electrode contact points or muscle activity crosstalk. Surface electrodes are placed on the subject's skin, at key locations, using from 1 to 12 electrode leads depending on the study or application. For example, smartwatches only use 1-2 leads, while for clinical evaluations, it is common to use up to 12 leads. Since the ECG will only be used for

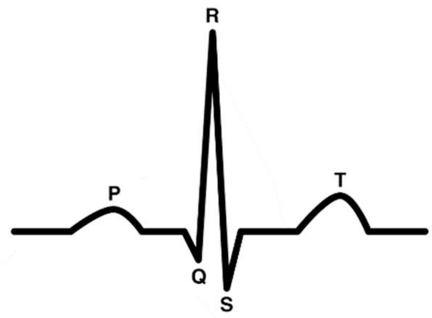



Figure 2.1: ECG waveform ([4])

detecting heart rate in this project, one to two leads are enough to detect it, without compromising signal quality.

The QRS complex represents the ventricular contraction phase, the most critical for detecting the heartbeat. There are multiple algorithms to detect this phase. The Pan-Tompkins Algorithm [6] is the most common algorithm used for heart rate detection and employs an adaptive threshold. It starts with the filtering using a bandpass filter, which integrates a low-pass filter to remove noise and a high-pass filter to remove baseline wander and T wave interference. Then, a five-point derivative filter is used to enhance the steep changes in the ECG. After this, a squaring function makes every point positive and enhances the higher frequencies, important for the QRS complex detection. Then, a Moving Window Integration is used to smooth the signal and detect waveform features. Finally, a threshold-based peak detection is employed, and two thresholds are used. The first one to detect the R-waves and a smaller second threshold to detect smaller peaks to avoid missing the signal [6].

This algorithm is highly used due to its small computational costs and high accuracy, making it reliable for real-time acquisitions. With this said, accurately defining the thresholds is essential, as it is sensitive to motion artefacts in sports. Some alterations have been proposed to improve detection of the heart rate without compromising computational efficiency Wu et al. [7].

Another algorithm highly used to extract the QRS complex is the Wavelet Transform-Based QRS Detection, first introduced by Kadambe, Murray, and Boudreaux-Bartels [8], and since then, it has been refined to perform better in noisier conditions, as dynamic exercises [9]. This algorithm also uses a bandpass filter to remove noise, then it applies a Dyadic Wavelet Transform (DyWt), decomposing the ECG into scales. Next, a peak detection algorithm is used across the Dyadic scales, where the absolute values of the wavelet show local maxima and the QRS complex, and these peaks align at successive scales. Then, using a threshold smaller than the maximum peak values allows to detect all QRS complexes. Additionally, a refractory period is employed to ensure only one peak per heartbeat, deleting false QRS complexes. This algorithm allows for higher accuracy in noisy data but imposes higher computational costs.

Finally, many other algorithms employ ML models capable of detecting these signals Zhao et al. [10]. One example is a deep neural network (DNN) with a feature pyramid network and a location head to predict the QRS complexes. This type of algorithm showed comparable results to the state-of-the-art algorithms and better results than the Pan-Tompkins algorithm. With this said, this algorithm has the highest computational costs of them all.

## 2.2 Inertial Measurement Unit (IMU)

The Inertial Measurement Unit or IMU was created in the 1930s for aircraft navigation [11]. With the introduction of micro-electro-mechanical systems (MEMS), it was possible to integrate these into smaller electronic circuits and embed them into wearables and smart devices. This technology is essential for motion recognition, and its main components are accelerometers and gyroscopes, with some solutions also integrating magnetometers. Although the magnetometer can give valuable complementary information on orientation, this sensor is out of scope for this project due to its unavailability. IMUs are widely used in gait analysis, rehabilitation, wearable medical devices, and for performance analysis in sports [12–14].

### 2.2.1 Accelerometers

Accelerometers detect changes in motion and orientation by measuring acceleration along one or more axes, depending on the application. These sensors are integrated in navigation for position estimation and airbag deployments. They can also be implemented in daily use devices like smartphones and wearables for screen rotations, step counting, and sports applications for tracking movement and physical exercise.

The accelerometers used in IMUs can be capacitive, piezoresistive, optical, thermal and resonant-frequency-based [15]. The capacitive sensors have three main components for detecting force: a substrate or frame, a spring and an inertial mass. The movement of this inertial mass compared to the frame causes a change in capacitance. This change in capacitance generates a difference in the electrical signal, and the acceleration is calculated. This principle applies to the three axes of movement, allowing for a complete 3D movement analysis. Important aspects when choosing the correct sensor is assessing the different bandwidth ranges, sensitivity, response frequency and dynamic range [16].

### 2.2.2 Gyroscope

Gyroscopes are sensors that measure angular velocity or the rate at which an object rotates around an axis. The Gyroscope present in MEMS functions using the Coriolis effect.[16] This effect states that when a mass moves within a rotating reference frame, it experiences an apparent force perpendicular to its motion. This works by having vibrating masses that oscillate at a fixed frequency. Then, when the device rotates, the Coriolis force causes the vibrating mass to shift perpendicular to both the oscillating and rotation axes. This displacement is then quantifiable using capacitive sensing, allowing this measurement to turn into an electric signal [17].

MEMS gyroscopes have become widely used due to their compact size, low power consumption, and cost-effectiveness, making them suitable for consumer electronics, automotive systems, and navigation applications.

## 2.3 Artificial Intelligence

### 2.3.1 Historical background

Artificial Intelligence can be defined as the ability to construct algorithms that mimic human intelligence. The goal is to train these algorithms on data to enable automated data processing and decision-making with greater efficiency and speed than manual human analysis [18]. A clear advantage of artificial intelligence is its capacity to provide responses based solely on objective data, without being influenced by emotional factors, which affects human decision-making.

AI can be traced back to 1950 with the published work "Computing Machinery and Intelligence" of Alan Turing, inventor of *The Bombe*, a machine capable of deciphering the Enigma code used by the German army during the Second World War, and in 1955 by John McCarthy with the first introduction of the term "Artificial Intelligence" during a workshop at Dartmouth College [19, 20].

### 2.3.2 AI applications

AI can be implemented in a wide variety of scientific applications. In medicine, AI can be used for unsupervised protein-protein interaction algorithms to discover novel therapies [21], process and cross-examine patient medical records to detect patterns, aid and perform surgeries [22], and care for patients [23]. In robotics, AI can be integrated to assist in tasks that used to be conducted by humans but required physical aspects of robotics like assisted and self-driving vehicles [24] and handling objects [25, 26]. AI has grown in Education, with students and professors increasingly using it [27]. AI can be integrated in PE classes for automatic detection of sports metrics allowing teachers to better examine students.

### 2.3.3 AI for sports activity detection

AI has been implemented in sports science for the automatic detection of HAR, step counting techniques [28] and score prediction [29]. Considering this broader areas, it can be concluded that the ability of detecting school and training activity with the integration of sensorised vests provides quantifiable value for educators. This enables for a more objective evaluation and improved overall performance allowing teachers to provide feedback on how to better execute a task within the sport.

For HAR, two main techniques in AI can be used: shallow machine learning and deep learning (DL). The main difference between shallow ML and DL is the number of hidden layers. When talking about artificial neural networks, data preprocessing and understanding what is being done inside the algorithm are also important. For shallow machine learning, it is necessary to introduce well-constructed features extracted from the raw data. In contrast, deep learning can analyse high-dimensional data and extract features from it without human interaction [30].

### 2.3.4 Machine Learning Models

This subsection presents different machine learning models used in the methodology of this thesis.

#### 2.3.4.1 Random Forest

The Random Forest algorithm is a supervised learning technique utilized either for classification or regression purposes. Its is based on another machine learning model known as Decision Tree (DT). This method constructs a forest (multiple decision trees) generated based on subsets of data and prediction variables. The final result of the model is obtained by aggregation: in the case of regression by the average of the individual predictions of the multiple decision trees; in the case of classification by a voting scheme.

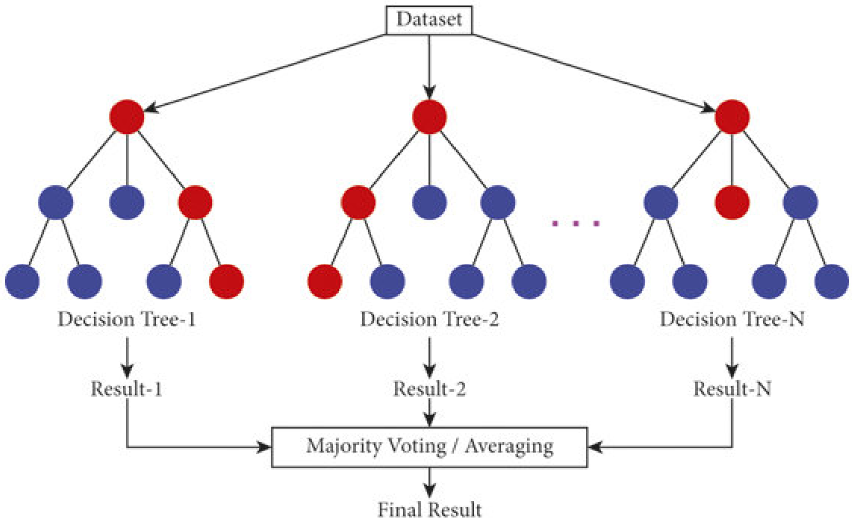



Figure 2.2: Ilustration of Random Forest dacision treen and voting scheme ([31])

The random forest classifier starts of by, dividing data into smaller data structures at a random order (instead of using the entire dataset to train one unique model), and then it implements smaller and simpler decision trees. After the model is trained it combines the results in order to have a more robust prediction. This is a technique known as bootstrap aggregation. The random forest allows for a high prediction accuracy even when there is a high number of features or noise. This is also a good technique against overfitting because of the bootstrap aggregation. One drawback of this ML model is the difficulty of interpreting individual decisions due to its complex ensemble of decision trees creating a "black box" effect [32].

#### 2.3.4.2 ExtraTrees

Extra Trees Classifier is a machine learning model that closely resembles the random forest algorithm. In fact, this model builds on the existing principal of RF to allow for a faster classification without compromising accuracy. This models also starts by choosing a number of random features like RF but for each feature it draws a random threshold, then it selects the split that gives the best score. It then builds on this principle to grow the decision trees and, just like RF it it follows a voting scheme for classification. A core difference with the RF model is that ET uses the entire training data to grow each tree.

This technique allows for a faster classification process due to the inexistent optimization over thresholds. Additionally, due to higher randomization this model also contributes to a lower variance [33].

#### 2.3.4.3 Gradient Boosting Classifier

The Gradient Boosting Classifier is also a supervised algorithm based on decision trees that builds DT in a sequential matter. This model uses the boosting technique in order to iterate over the training models to correct previous errors. It uses a weighting scheme for handling harder challenges in the classification process. It iterates over the DT in order to minimize the loss function allowing for better results than simpler models. This process is illustrated in Figure 2.3. The GBC starts by initializing a DT with random parameters. Then, residual errors are calculated and a new decision tree is adjusted. These errors are minimized based on the loss function and, a learning factor is attributed to each tree. This is done for a predefined number of iterations that can be adjusted to improve accuracy.One drawback of this model is that its iterative process makes it a sequential algorithm. This differs from the parallel nature of random forests, making it more computationally expensive and should be taken into account for bigger datasets [34, 35].

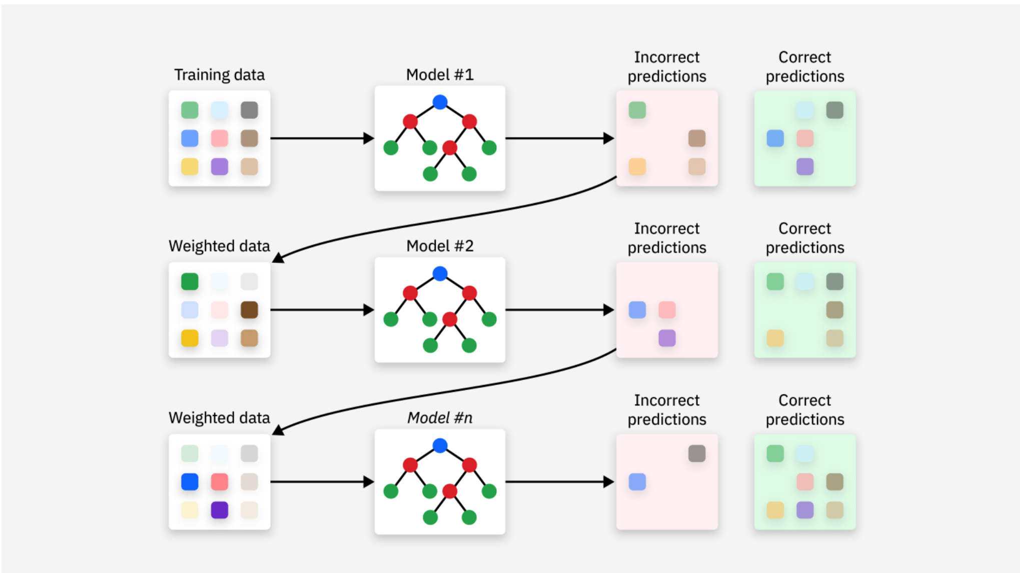



Figure 2.3: Weighting technique for GBM([34])

#### 2.3.4.4 LightGBM

The lightGBM machine learning algorithm comes to solve some of GBM shortcomings mainly its high computational costs, making this algorithm useful for larger datasets. This model implements two different innovations.

The Gradient-based One-Side Sampling (GOSS) maintains the observations with higher gradients, (observations the model struggles to classify), and uses the lower gradients (easier classification) to do a random sample and build a decision tree using this data.

Additionally, it uses the Exclusive Feature Bundling (EFB) where mutually exclusive features (features that usually explain similar characteristics) are grouped together in order to reduce dimensionality. This algorithm does this without losing significant information. These two techniques highly reduce the training and testing times, and also improve the effectiveness of the model by reducing the memory needed [36].

#### 2.3.4.5 K-Nearest-Neighbors

K-Nearest-Neighbors or KNN is a non-parametric and simpler model than the previously analysed algorithms, where the classification is based on distances. This algorithm is considered a "lazy model" since it does not build an explicit model during the training phase but, instead, it just stores the data.

The classification process for the test data is based on the distance from the test information to the K (K being a predefined number of points) nearest points. This can be calculated based on euclidean distance or other similar metrics. The result of the classification is determined by the majority number of classes present in this space. This takes into consideration that the same class should have the majority of its data bundled together [37]. In figure 2.4 there is an example of the classification process and bundles of data mentioned before.

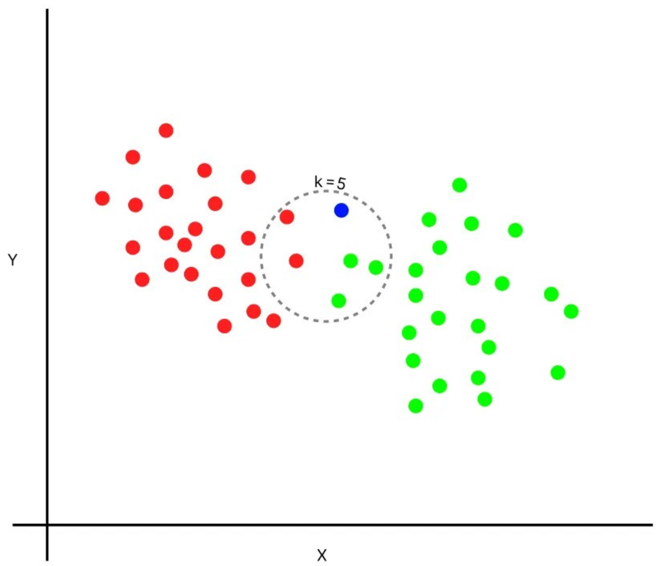



Figure 2.4: KNN classification process([38])

This model presents two important drawbacks: its computational cost because, for each new data point, it is necessary to calculate the distance to its neighbours; the the number of neighbours highly influences the classification accuracy. If too many neighbours are chosen it can integrate data from two different classes and with too little neighbours one the data is more susceptible to outliers affecting the classification.

# Chapter 3

# State of the art in physical activity classification based on body sensors

## 3.1 Literature review

A literature review was conducted using the PRISMA ( Preferred Reporting Items for Systematic reviews and Meta-Analyses) [39] methodology for the state of the art. With the advances in MEMS technology, there has been an increase in studies of HAR. This research depict the type of movements detected, the hardware being used, and the algorithms for pre-processing data and classification methods. For this research, three literature databases were analysed using advanced search settings to search for specific keywords. These databases were Web of Science, IEEE, and SCOPUS. This research was limited to 5 years from 2020 to 2025. Figure 3.1 shows the search process that went into this review. Additionally, these papers where extracted through the advanced search mechanisms in these databases using the following keywords present in the title or abstract: machine learning, ML, artificial intelligence, AI, deep learning, physical education, physical activity, basic movements, fundamental movements, detection, recognition, identification, chest mounted, accelerometer, inertial sensor, inertial measurement and IMU. However, for some databases, certain keywords where excluded from the research like: exosckeleton, step count and more. These research was also limited to fileds of interest like, computer science, engineering and medicine. The three databases searched culminated in 186 records for screening after removing duplicates, of these,88 papers were excluded for out of scope reasons. One paper was not retrieved due to limited access. Then after analysing the 97 papers, 42 were included in this review and 55 were excluded for the listed reasons.

### 3.1.1 Sensor placement

Sensor placement plays a key role in the performance and generalization of HAR classification models, as it influences the quality and interpretability of the collected signals. This research is focused on full body description in datasets that have a sensor in the torso region. Table 3.1 shows the amount of different sensors placements among the most used datasets, including the waist,

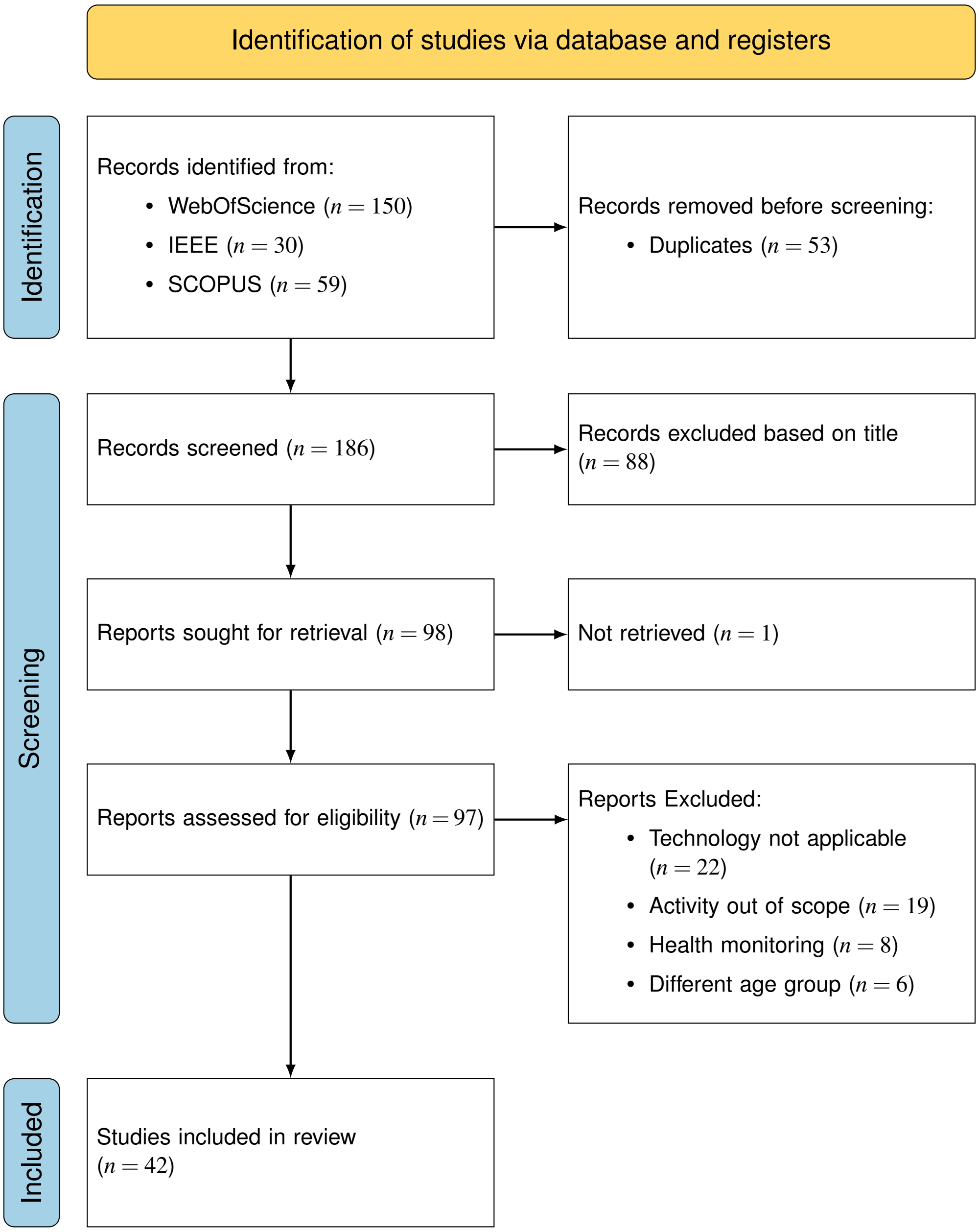



Figure 3.1: Methodology for literature review highlighting databases, duplicates removal and exclusion criteria

wrist, chest, hip, thigh, shin, and upper arm. The pocket is the most common placement, as seen in WISDM, MobiAct, MOTIONSENSE and HHAR, offering a central and relatively stable point to capture body movement. However, other datasets such as PAMAP2, mHealth, and RWHAR include multiple sensor placements, such as the chest and limbs, to enrich motion data and capture complex activities like jogging or cycling more accurately. Some studies look at every day devices, as smartphones and smartwatches, to capture motion signals. Others employ specific motion capture sensor systems to more accurately characterize specific movements of limb segments. The usage of the smartphones has some disadvantages like reduction in resolution and uncontrollable sampling frequency, disadvantages that can be mitigated, for example, using a specific external sensor as an IMU, used and placed specifically for the purpose of movement characterization [40].

Table 3.1: Sensor characteristics, placement and number of activities in each dataset for daily living activities

| **Sensors** | **Samp. Freq. (Hz)** | **Location and # Sensors** | **# Act.** | **Dataset** |
|---|---|---|---|---|
| IMU, Mag. Temp, ECG | 100 | Chest, Wrist, Ankle (3 sensors) | 12 | PAMAP2 [41] |
| IMU | 50 | Waist (1 sensor) | 6 | UCI-HAR [42] |
| IMU | 20 | Hand, Front pocket (2 sensors) | 18 | WISDM [43] |
| IMU, Mag. | 85 | Front pocket (1 sensor) | 13 | MobiAct [44] |
| ECG, IMU, Mag. | 50 | Chest, Wrist, Ankle (3 sensors) | 12 | mHealth [45] |
| IMU | 50–200 | Wrist, Pocket (2 sensors) | 6 | HHAR [46] |
| IMU | 50 | Front pocket (1 sensor) | 6 | MOTION-SENSE [47] |
| IMU, Mag. | 100 | Hip (1 sensor) | 12 | USC-HAD [48] |
| IMU | 100 | Hip (1 sensor) | 18 | KU-HAR [49] |
| IMU, Mag. Light, Sound, GPS | 50–200 | Chest, Forearm, Head, Thigh, Shin, Upper arm, Waist (7 sensors) | 8 | RWHAR [50] |
| IMU | 50 | Waist (1 sensor) | 6 | SBHAR [51] |

The use of multi-position configurations, as seen in RWHAR with up to seven different sensor placements, enables higher detail capturing and is beneficial for recognizing more complex movements. However, this comes at the cost of increased system complexity, data synchronization requirements, and reduced wearability, which may limit real-world deployment.

On the other hand, single-sensor configurations, such as those in MOTIONSENSE or UCI-HAR, provide simplicity and ease of use but may lead to lower classification accuracy, particularly for activities involving higher movement in the extremities of the body.

Another aspect to consider is the fact that some datasets are used for specific applications, and the sensors can be integrated in a specific configuration to achieve more success in activity recognition. For example, hip or thigh sensor placements, as seen in datasets like in KU-HAR and USC-HAD, can be effective for lower-body movements like walking, running, or stair use, while wrist-mounted sensors (HHAR) are more relevant in applications that emphasize upper-body motion or require user comfort in long-term monitoring scenarios. Finally, datasets like mHealth and PAMAP2 that include sensor fusion from different body parts (chest, wrist, ankle) allow for the development of robust models capable of recognizing full-body movement patterns with greater accuracy.

In summary, sensor placement strategy is a trade-off between recognition accuracy and practical deployment constraints, and the choice depends heavily on the target application. While multi-sensor, multi-location setups yield more informative data, the trend towards minimal, wearable systems pushes the field to develop models that can maintain high accuracy with fewer, strategically placed sensors.

### 3.1.2 Datasets procured

Most papers utilize the same publicly available datasets from the UCI-HAR, WISDM, PAMAP2 and mHealth and MobiAct [44, 52–55]. The mHealth dataset uses IMUs placed on the chest, right wrist and left leg, and an ECG sensor with a 2-lead system. The ECG provides the heart rate data, and the IMU provides the accelerometer, gyroscope and magnetometer. This dataset includes sensor data from 10 participants for 12 activities: standing, sitting, lying down, walking, climbing stairs, waist bends forward, frontal elevation of arms, knee bending, cycling, jogging, running and jumping front and back [54, 56].

The UCI-HAR dataset uses data from 30 participants captured by a smartphone with 3D accelerometer and gyroscope placed at the waist. The activities recorded with this system are: walking, walking upstairs, walking downstairs, sitting, standing, and lying down [57].

The WISDM dataset comprises recordings of 36 participants with a smartphone in their pockets. It integrates an accelerometer sensor and captures information of six activities: walking, sitting, standing, walking upstairs, walking downstairs and jogging [58].

The PAMAP2 dataset is the most extensive dataset, with 12 main activities: lying, sitting, standing, walking, running, cycling, Nordic walking, ironning, vacuum cleaning, rope jumping, ascending, and descending stairs. In addition, some extra activities include watching TV, working on the computer, driving a car, folding laundry, cleaning the house, and playing soccer. This was recorded in 9 subjects with 3 IMUs and an ECG. One IMU and the ECG are placed in the chest area, the other IMUs in the dominant wrist and ankle [53, 57].

The MobiAct dataset is focused on a daily living and fall detection approach. It uses accelerometer sensors to detect 15 activities with a total of 67 participants. Additionally it was extracted from each individual the sex, age, size and weight [59]. These datasets help evaluate a model's performance since they are standardized tests with many applied models. This makes it worthwhile to compare these datasets with different models.

In addition, datasets have also been developed for specific applications. For example, in [60], a dataset of basketball activity was created, with specific training and in-game records; the participants had various skill levels, allowing for a more generalizable dataset. This data was recorded in 24 participants from Germany and the USA. This also introduced differences in the dataset due to the different basketball rules in each country. Data was collected using a wrist-worn accelerometer and validated using an optical motion capture system, as a ground truth system, that enabled specific basketball fundamental motions labelling.

In table 3.2, studies are presented that use either custom datasets developed by the specific research group that are not available for general use or publicly available datasets. A notable amount of papers use a custom dataset, which may be due to specific application requirements, or novel sensor configurations making these datasets not applicable for this study benchmarking purposes. Some papers use more than one publicly available dataset or both a custom and a publicly available dataset for validity purposes. Additionally, it is possible to verify the usability of the UCI-HAR, PAMAP2, WISDM, MobiAct and mHealth datasets since these are top 5 most used dataset in this research.

Table 3.2: Datasets used in the literature review

| **Dataset** | **Paper** | **Dataset** | **Paper** |
|---|---|---|---|
| Custom Dataset | [57, 61–68] [55, 60, 69–75] | [45]- mHealth | [54, 56, 59] |
| [42]- UCI-HAR | [57, 59, 76–79] [52, 53, 56, 80–82] | [41]- PAMAP2 | [53, 56, 57, 59, 79, 82–85] |
| [86]- SHO | [76] | [43]- WISDM | [53, 55, 58, 59, 80, 87–89] |
| [43]- Capture24 | [90] | [46]- HHAR | [59, 90, 91] |
| [92]- Pervasive Systems | [93] | [94]- IM-WSHA | [79] |
| [44]- MobiAct | [56, 59, 79, 95] | [47]- MOTIONSENSE | [79, 95] |
| [48]- USC-HAD | [59] | [49]- KU-HAR | [80] |
| [96]- UniMIB-SHAR | [95, 97] | [98]- Opportunity | [82, 84, 91] |
| [99]- Daphnet | [82] | [100]- SPHERE | [82] |
| [101]- Wetlab | [91] | [50]- RWHAR | [91] |
| [51]- SBHAR | [91] | [102]- PUC-Rio | [54] |
| [103]- AReM | [54] | [104]- ADL | [84] |

Most of the literature and datasets encompass only IMU sensor data for activity recognition. Only around 12% of the papers analysed used a heart rate detector (for example, an ECG) integrated with the IMU for any feature extraction. For example, the PAMAP2 dataset has heart rate detection. This limited adoption of heart rate sensors may be attributed to increased hardware complexity, power consumption, and data synchronization challenges.

### 3.1.3 Pre-processing and feature selection of IMU and ECG data

The literature evaluates alternatives and strategies to pre-process the IMU and ECG data. In Uddin et al. [54] tri-axial accelerometer data was extracted from the waist, left thigh, right ankle, and right upper arm from the dataset [102]. Then, for each activity features like mean, variance, standard deviation, skewness, and kurtosis were extracted. After this, the data is represented in a fused feature space. This is a key step for Kernel Principal Component Analysis (KPCA). This method is a alternative to the traditional PCA method that instead of using the original feature space, it maps the input features into a higher dimensional space using a Gaussian kernel. After this a PCA technique is used reducing the dimensionality again and extracting the principal components. After this a RNN (Recurrent Neural Network) is used to classify the model.

Additionally, some literature integrates signals from ECG and accelerometer, like [72]. This paper uses ECG, pedometer, and accelerometer data to predict HAR. They calculate the median heart rate of each participant to have a baseline measurement. This allows for a more specific approach, minimises bias, and considers different variables that can influence the heart rate, like age and physical conditions. Then, they calculate the first and second derivatives to the heart rate characterize the signal more efficiently over time. Finally, the signal is normalized using a conventional maximum heart rate of 220 bpm. For the accelerometer data, they hypothesize that the algorithm could learn information from each individual, and that could interfere with the activity recognition. This way, they read all accelerometer data of each axis and participant, compute the median, and use this to debias the raw accelrometer signal. Then, they use a Gaussian filter to remove noise and smooth the signal. Finally, the first and second derivatives are computed to extend the information available for the recognition. This was essential to introduce the information to separate neural networks for each signal and later introduce a product fusion for classification.

### 3.1.4 How are machine learning algorithms integrated with the IMU and ECG

Many algorithms were deployed in the literature for HAR classification. Around 37% of algorithms use machine learning, 47% use one type of neural network and 16% use specifically a Long Short-Term Memory network (LSTM), as shown in Figure 3.2. Of all the LSTMs used, deploying it in a hybrid model is common, integrating it as a layer of a more complex structure using CNN for its feature-extracting capabilities. The LSTM is used to analyse sequential data, such as time-series signals. It is a type of Recurrent Neural Network capable of analysing dependencies in sequential data. It usually has an input gate that controls how much new information enters the memory cell, a forget gate that decides what part of previous data should be discarded, and an output gate that determines what information is sent to the next time frame. This type of algorithm is frequently used for HAR since it can analyse dependencies in sensor data and motion patterns [65].

For example, in Mekruksavanich, Jantawong, and Jitpattanakul [85], a bi-directional LSTM (biLSTM) is used. This algorithm processes information in forward and backwards directions, capturing past and future dependencies. This is important for HAR, as the classification of a

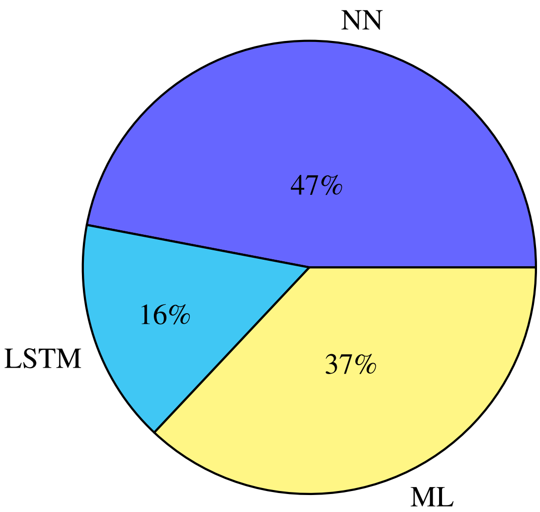



Figure 3.2: Percentage of NN, LSTM and ML models used in the literature

single movement involves analyzing both preceding and succeeding windows, enabling the model to recognize the movement as a whole. This type of LSTM achieved 97.5% accuracy, which outperformed traditional machine learning algorithms.

One example where the LSTM is used in a layer where CNN is also implemented is in Vakacherla et al. [57], where the CNN is used for its feature extraction capabilities and the LSTM for its time dependency. This was tested offline and online, achieving accuracies of 97.8% and 97.2%, respectively.

An example where only CNN is used is Sharen et al. [80] where a new WISNet model, comprised of a Convolved Normalize Pooled block that extracts essential features from the raw data, an Identity and Basic block to extract residual features and a Channel and Spatial Attention block that assigns importance to more relevant features. Of all the implementations tested in this paper's methodology, this integration achieved the highest accuracy in 3 public datasets: 96.41% for the WISDM dataset, 95.66% for the UCI-HAR and 94.01% for the KU-HAR.

Finally, an implementation where only machine learning algorithms are deployed is Tahir et al. [79], where a random forest is used after a pre-processing and feature extraction series. This random forest has a feature of bagged trees, which has become an optimal model for creating a training test. This ML model achieved the best outcome of the other discussed models, with an accuracy above 90% in all datasets tested.

Table 3.3 provides a comparative overview of various ML models applied to HAR tasks, specifically Activities of Daily Living (ADL), fitness exercises, and fall detection. The reported accuracy values demonstrate that the performance of each model varies significantly depending on the type of activity and the dataset used in each referenced study. In this table the accuracy range illustrates minimum and maximum values found in the documented papers. Some papers use different datasets that appear in this range but the objective with this table it to present the effectiveness of some machine learning algorithms. Additionally, with more referenced studies usually comes a higher range, since different techniques are used for feature extraction, reduction etc.

The following paragraphs discuss the machine learning algorithms used for Table 3.3:

- In general, Support Vector Machines (SVM), Random Forests (RF), and K-Nearest Neighbors (KNN) are the top-performing models in the context of ADL and fitness recognition, with reported accuracies often exceeding 90%. For instance, SVM achieves accuracy values ranging from 87% to 97% for ADL, and 96% in fitness-related tasks. Similarly, RF shows high performance, with accuracies between 93% and 95% for ADL and fitness activities. Additionally, RF is the most documented classifier both in ADL and ADL with fitness exercise. KNN also demonstrates competitive performance, with accuracy reaching up to 97% depending on the dataset and experimental setup.

- In general, Support Vector Machines (SVM), Random Forests (RF), and K-Nearest Neighbors (KNN) are the top-performing models in the context of ADL and fitness recognition, with reported accuracies often exceeding 90%. For instance, SVM achieves accuracy values ranging from 87% to 97% for ADL, and 96% in fitness-related tasks. Similarly, RF shows high performance, with accuracies between 93% and 95% for ADL and fitness activities. Additionally, RF is the most documented classifier both in ADL and ADL with fitness exercise. KNN also demonstrates competitive performance, with accuracy reaching up to 97% depending on the dataset and experimental setup.

- Traditional models such as Logistic Regression (Log. Regr.) and Naive Bayes also perform well, especially for ADL classification, with reported accuracies up to 96%. However, these models show a higher variability and are documented in fewer studies than the previous models.

- Gradient Boosting Classifier (GBC) and Sequential Minimal Optimization (SMO) also yield strong results, achieving up to 94% and 92% accuracy, respectively, showing promising results even though they are used in the fewest papers.

- Finally, it is worth noting that when the dataset ADL is combined with activities like fall detection the accuracy fell in all used ML models. This can be explained by the increase in the number of activities contributing to a more difficult separation between them.

In summary, ensemble-based models such as RF and GBC, as well as margin-based classifiers like SVM, are particularly well-suited for HAR tasks involving ADL and fitness exercises, offering high classification accuracy and robustness across different implementations. Because of the high impact and presence of deep learning models in literature, these models were included in the literature review, even though they were not used for implementation due to out of scope reasons such as higher computational costs and lower model interpretability comparing to traditional ML models.

Table 3.3: Machine Learning Models and Their Performance on Different Activities

| ML Model | Type of Activities | Papers | Accuracy (%) |
|---|---|---|---|
| SVM | ADL | [77], [87], [66], [81] | 87–97 |
| | Fitness exercise | [73] | 96 |
| | ADL and fall detection | [65] | 73 |
| KNN | ADL and fitness exercise | [67], [93], [73] | 86–97 |
| | ADL | [66], [58], [81] | 90–95 |
| | ADL and fall detection | [65] | 68 |
| Log. Regr. | ADL | [77], [66], [89], [81] | 79–96 |
| Naive-Bayes | ADL and fitness exercise | [93] | 96 |
| | ADL | [77] | 77 |
| DT | ADL and fitness exercise | [93] | 96 |
| | ADL | [77], [66], [89], [68], [81] | 83–88 |
| | ADL and fall detection | [65] | 61 |
| RF | ADL and fitness exercise | [61], [67], [93] | 93–95 |
| | ADL | [77], [87], [66], [68], [58] | 73–95 |
| | ADL and fall detection | [65] | 75 |
| GBC | ADL | [66], [58], [81] | 90–94 |
| SMO | ADL and fitness exercise | [67] | 92 |

## 3.2 Commercial solutions for smart sports vests

New developments in sports technology lead to the design of smart sports vests, capable of continuous, accurate and objective physical activity monitoring. These vests integrate several sensors, usually IMU's and ECG into the fabric. This allows for a standardized and ergonomic system for data collection, which solves some of the issues seen with smartphones.

The fixed sensor location is one of the key benefits of smart vests. The signals are more repeatable and reliable between sessions since sensor modules and the vest are stitched together. This is especially helpful for HAR that need complex full-body movements or prolonged monitoring. Real-time monitoring and feedback are made possible by the wireless transmission and embedded processing units found in certain smart vests. However, smart vests are not without limitations. They are more expensive to produce and there can be problems with the connections in the garment with real world applications where the vest due to excessive movement can be subjected to disconnections. Also, user comfort and the ability to wash the garment are still challenges being worked on. Even so, with the improvements in textile electronics and flexible sensors, smart vests are becoming more viable for integration [105].

In general, smart vests offer a middle ground between the precision of dedicated IMUs and the convenience of smartphones. Because of that, they're gaining popularity in areas like sports performance tracking, healthcare, elderly monitoring, and workplace safety.

To better understand the versatility and impact of smart vests, it is useful to examine various models that integrate different sensor configurations, form factors, and use-case optimizations.

KINEXON sports [106] offers a vest known for its load management characteristics in sports. The vest is equipped with a small ultra-lightweight wearable (measuring only 15g) IMU with accelerometer, gyroscope and magnetometer and a heart rate detector derived from an ECG sensor. This vest is specialised in calculating acceleration load, mechanical load, jump load, number and height of jumps, max speed, distance and changes of orientation. This data can be viewed live using an anchor that captures, processes and displays the live data within the KINEXON app. An example of the KINEXON vest can be seen in Figure 3.3


Figure 3.3: KINEXON sports vest [107]

The Light Brain Smart Athlete Vest [108] has an ECG with 12 sensors built in, GPS, accelerometers, a gyroscope, and a digital stethoscope. These sensors are capable of calculating heart rate, breathing rate, body temperature, fatigue levels, real time positional movements and orientation, impacts and forces experienced across the body, and monitor wellness and injury prevention.

Additionally, the Hexoskin garment [109] provides vital information that can be useful either for sports, elderly and workers that need constant check on health for safety purposes. This garment uses a 1-lead ECG that can capture heart rate, heart rate variability for stress monitoring, effort, load and fatigue, QRS intervals, etc. It also has Respiratory inductance plethysmography sensors for breathing rates, minute ventilation and VO2max, and a 3-axis accelerometer for activity intensity, steps cadence and positions. Overall this garment focuses on health monitoring when compared to the rest of the vests that have a sports-focused approach with different sensors.

Finally, the Catapult vest [110] uses an IMU with 3-axis Accelerometer, Gyroscope and magnetometer and a heart rate device based on ECG. This device is capable of calculating total distance covered during activity, as well as detailed velocity metrics, including maximum speed and time spent in various speed zones such as walking, jogging, or sprinting. This vest also tracks acceleration and deceleration events, capturing both their frequency and intensity, which are crucial indicators of an athlete's workload during dynamic movement.

## 3.3 TexP@ct solution

The solution created by TexP@ct [111] has the primary objective of accurate monitoring and continuous analysis of different biomechanical and physiological sensors through a sensorised vest. This project approach backs the HAR, by integrating an IMU in the textile of a sports vest for real time acquisition and post exercise report. This vest only integrates accelerometer data, gyroscope data and electrocardiogram for heart rate detection. This project is aimed at a solution for education and sport. Enabling PE professors to not only monitor the heart rate and velocity intensity in real time, but also get a full post exercise report with the fundamental movements of each sport.

This project is under development for a consortium of companies and each one has a different focus for the development, such as: The vest with integrated sensors, that needs to have an ergonomic fit, be durable, comfortable and washing machine resistant; The mobile platform that works as an interface between the user and the system, displaying live metrics and post exercise reports using a clean UI in order to present relevant information and be easy to navigate; A software solution capable of efficiently computing all necessary metrics from sensor data. It needs to performs signal preprocessing and leverages either machine learning models for classification or algorithms for real-time parameter detection and localization; Finally, a robust IMU equipped with accelerometer and gyroscope sensors, powered by a battery and with wireless communication properties built with a durable structure to withstand impact forces. In the table 3.4, additional sensor information can be visualised.

| **Sensor** | **ECG** | **ACC** | **GYR** |
|---|---|---|---|
| **Number of channels** | 1 | 3 | 3 |
| **Resolution** | 16 bit | 16 bit | 16 bit |
| **Input full-scale** | ± 2.5mV | ± 8G | ± 500dps |
| **Sample rate** | 1000Hz | 1000Hz | 1000Hz |

Table 3.4: Sensor specifications for ECG, accelerometer, and gyroscope.

# Chapter 4

# Methodology

This chapter goes over the overall process of the methodology: extracting the datasets; building the required dataframe; preprocessing of data in order to remove noise and gravity effects; splitting the data into train and test sets; different feature extractions for different sets; classification processes including feature reduction techniques and hyper parameter tuning. The aim of this methodology is to contribute to a generalizable classification process for human activity recognition.

In Figure 4.1 presents the complete process, dependencies and results that this methodology is able to extract.

## 4.1 Datasets description and preparation

In this section the datasets PAMAP2 and RWHAR are described in detail regarding sensor location, characteristics, participant number and preparations to build a dataframe for easier and homogeneous pre-processing and classification. These datasets were chosen due to activities present resembling fundamental movements within sports and because both datasets had accelerometer and gyroscope data in the torso region. Although both datasets have demographic information like age group, sex, height and weight, in the dataset itself the subject information is anonymous making it harder to appropriately balance the train and test parts of the dataset.

### 4.1.1 PAMAP2 dataset

As discussed in the previous chapter, the PAMAP2 dataset is a widely used benchmark in the field of human activity recognition. It encompasses annotated recordings from nine participants—eight males and one female—with an average age of 27.22 $\pm$ 3.31 years. The participants performed a predefined set of physical activities while wearing three inertial measurement units (IMU's) and a heart rate monitor[41].

In order to extract the data, three wireless Colibri IMU's from Trivisio Technologies were used and positioned on the dominant wrist, chest, and dominant ankle of each subject. The IMU's sampled data at 100 Hz with two types of tri-axial accelerometer signals ( $\pm$ 16g and $\pm$ 6g), gyroscope data (rad/s), and magnetometer data ($\mu$T) and sensor for temperature capturing. This

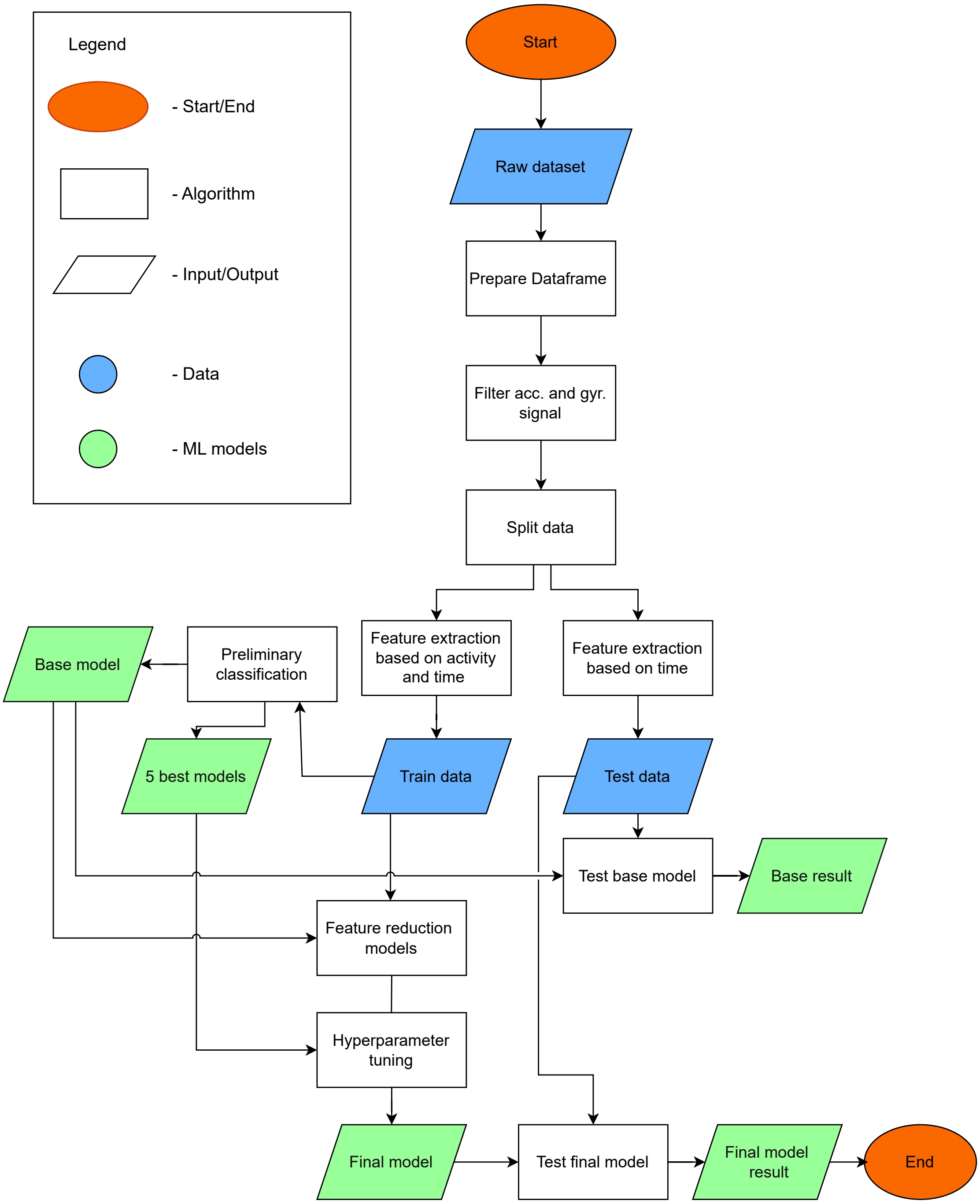



Figure 4.1: Fluxogram of general methodology

dataset has two different accelerometers it is suggested the use of $\pm$ 16g accelerometer due to its higher resolution in order to minimize saturation. Additionally, heart rate data was collected at approximately 9 Hz using a BM-CS5SR monitor. All signals were stored in .dat files containing 54 columns, including timestamps, activity labels, heart rate, and raw sensor signals. Only one file per participant was stored.

Participants performed a series of 18 different physical activities, from a protocol and a set of optional tasks. Each activity was performed for a defined duration, and transition periods between activities were labelled as undefined (activity ID 0) and excluded. The optional tasks were automatically discarded from this study since not every participant performed them and since they had the smallest amount of total time recorded, making it difficult for splitting and classification purposes. Additionally, the activities: sitting, Nordic walking, ironing, vacuuming, ascending and descending stairs were discarded since these activities do not fit the basic movements inside a sport, while the remaining are core movements of numerous sports: standing, walking, running, lying, cycling and rope jumping.

For this dataset there is already one .dat file per participant, in order to assemble a full dataframe it was only necessary to extract the timestamp, participantID, accelerometer $\pm$ 16g for axes X, Y, Z, gyroscope X, Y, Z, ActivityID and heart rate, and concatenate it for every participant.

### 4.1.2 RWHAR dataset

The Real World Human Activity Recognition Dataset is a robust and realistic dataset with natural variations in human behaviour. This allows for a more complete dataset without artificial constraints. This is an advantage over other datasets that have rigorous data capture and protocol activities with very more constrained environments and sensor placements. This datasets presents data extracted from 15 participants with an average age 31.9 $\pm$ 12.4, where 8 were males and 7 females. These participants performed eight activities: climbing stairs up and down, jumping, lying, standing, sitting, running/ jogging and walking [50]. For the purpose of this study only jumping, lying, standing, walking and running were selected.

As mentioned previously in Table 3.1, this dataset has 7 sensor locations and features an accelerometer, gyroscope, GPS, magnetometer, light and sound sensors. The data for these sensors was captured using either a smartphone (Samsung Galaxy S4) or a smartwatch (LG G Watch R). For the purpose of this research only the accelerometer and gyroscope were used. For data balancing purposes each participant performed all activities for about 10 minutes each, except for the jumping activity where it was an average of 1.7 minutes to minimize physical exhaustion.

Although the RWHAR dataset is reported to have a sampling rate of 50 Hz for all sensors, analysis of the raw sensor data revealed some sampling inconsistencies. Specifically, the accelerometer data is close to the expected sampling frequency of 50 Hz with some minor deviations in the start and end of the transmission of the signal. In contrast, the gyroscope data demonstrates substantial variability, with the sampling frequency sometimes having the correct sampling rate but occasionally having higher sampling, reaching 180 Hz. This can all be observed in the readMe files inside each activity.

With this in mind, some alterations to the dataset were made. Firstly, since only the accelerometer and gyroscope data is relevant, it is taken into account the timestamp of accelerometer data as the correct time. Secondly, in order to make sure the accelerometer data is sampled at 50 Hz the first and last timestamps have to be separated between samples 20 ms otherwise it shall be discarded until it is possible to verify this. After this the gyroscope timestamps are aligned with the accelerometer. Then if the sampling rate is different the signal is down sampled by a ratio of the length gyroscope data divided by the length of the accelerometer data. This ensures equal samples of data for both signals. After the sampling correction, the dataframe is built iterating for each participant, activity and sensor, discarding all other placements besides the chest, so that PAMAP2 and RWHAR have the same sensor placement location (i.e., near the chest).

## 4.2 Pre-processing of datasets

### 4.2.1 Handling missing values

After getting the dataframes for the two datasets some data imbalances were found, suggesting missing data, likely caused by packet loss during wireless data transmission. Such interruptions in data integrity are common in setups with wireless communication, where interference or problems with the buffer may lead to missing or corrupted data samples.

In order to counteract these missing values, linear interpolation is used to fill in the NaN (Not a Number) entries within the time series data. This method assumes there is a linear relationship between consecutive data. This is useful since a sampling frequency of 100 and 50 Hz for the PAMAP2 and RWHAR datasets, respectively, is high enough to ensure there is not a big shift between samples in IMU data. Therefore, significant non-linear fluctuations are highly unlikely to occur within such short intervals, especially in the context of biomechanical signals, such as those typically encountered in HAR.

### 4.2.2 Filtering Raw data

After handling the missing values the following step is to filter the raw signals from the accelerometer and gyroscope. This is a crucial step for the pre-processing because filtering reduces noise and motion artifacts. Real world signals are affected by various types of noise either environmental, static or electrical components. By introducing a filter, the signal-to-noise ratio can be enhanced allowing for more accurate results [112].

The band-pass filter, composed of a high pass and a low pass filter, allows frequencies within a specific range (band) to pass through while attenuating frequencies outside that range. Because of its flat frequency response in the bandpass, the Butterworth filter was chosen to provide smooth filtering without creating ripple effects.[113].

For this filter the user just needs to input the sampling frequency. The low pass cutoff frequency should be 20 Hz and the sampling frequency depends on the dataset used. For the high-pass cut off frequency is set to 0.05 Hz in order to remove the gravity component of each signal. This works because gravity is a constant acceleration vector ( 9.81 m/s$^2$) that affects accelerometers depending on orientation but does not vary rapidly over time and the human movement normally ranges between 0.5 and 10 Hz, so by setting the cut off to 0.05 Hz it should not have an effect on the human activity component of the signal [114].

### 4.2.3 Train test split of the dataset

In order to have a training and testing dataset, a group-aware data splitting strategy was employed to ensure the integrity and generalizability of the machine learning models. Specifically, the dataset was divided into training and testing subsets using the GroupShuffleSplit method from the scikit-learn library [115]. This split guarantees that the data is not divided at random but rather according to a grouping factor. Participants served as the grouping variable to direct the split after the features (X) and ground truth labels (y) were taken from the pre-processed dataset.

A single split was performed, allocating 80% of the participants to the training set and 20% to the testing set, while maintaining group independence between sets. This approach is critical in scenarios where multiple samples originate from the same individual, as is common in HAR tasks using physiological or motion sensor data (the case for IMU data). Without group-aware splitting, samples from the same participant could appear in both training and testing sets, and this would lead to data leakage and unrealistic performance metrics due to the model learning subject-specific patterns rather than generalizing activity features.

By conserving the participant independence between training and testing data, the GroupShuffleSplit method ensures a more realistic and robust evaluation of model performance, better reflecting how the model would perform on unseen individuals. This is especially important for applications intended to generalize across diverse populations. As addressed before, due to participant anonymity it is impossible to secure a division where demographic data is also well defined.

## 4.3 Feature extraction based on sliding window

For the feature extraction, two distinct sliding window-based feature extraction strategies were implemented to support both the training and evaluation phases of the activity recognition pipeline. This design choice was made to balance the need for clean, discriminative training data with the goal of realistic model evaluation under conditions that more closely resemble real-world scenarios.

The sliding window is a common technique for segmentation of time dependent data (like the accelerometer and gyroscope data) commonly used in HAR [93]. This technique divides the continuous signal into smaller fragments that may or may not be overlapping, depending on the use case. These fragments are called windows and within each one features can be calculated that represent certain characteristics of the signal.

This algorithm had 2 different controlled parameters: the window size that represents the amount of samples in each window (normally represented by the product of the sampling frequency and the seconds used); and the step size between the windows, this represents the amount of samples that will introduce new data, this means that for consecutive windows, if x is the number of samples, x minus step size is data from the previous window and the length of the step will contribute with new data.For this implementation a window size of 3 seconds and a step size of 2 seconds is used.

### 4.3.1 Training Data Feature Extraction

For the training data, features are extracted exclusively from separate activity segments. The dataset is firstly divided by activity class and a sliding window with an imputed sampling frequency is used. This ensured that all windows for training contain only a single activity type, free from transitions or mixed labels. For each window and for each of the six inertial sensor signals (accelerometer and gyroscope on three axes), a set of time and frequency-domain features are computed, including:

**Time-domain features:** mean, standard deviation, minimum, maximum, median, energy, and kurtosis.

**Frequency-domain features:** Peak frequency and its corresponding amplitude.

This approach allows the model to learn well-defined activity-specific patterns in a controlled and noise-free context, enhancing the differences between each activity category.

### 4.3.2 Test Data Feature Extraction

In contrast, for the test data, the sliding window is applied over the full temporal sequence without prior filtering by activity. This results in windows that could contain either just one activity or more. Each window is labelled based on the majority activity class (mode) of the samples within the window. This strategy reflects the realistic evaluation scenario because in a real world scenario there can be windows that can have 2 activities.

By separating the feature extraction strategies for training and testing in this manner, the approach maintains the integrity of clean learning during training while ensuring realistic and robust evaluation. This methodological distinction helps mitigate data leakage and provides a fair estimate of the model's generalization capability.

### 4.3.3 Feature Extraction Equations

Let $x = \{x_1, x_2, \ldots, x_n\}$ be a time-series segment of length $n$ from a sliding window, and let $f_s$ denote the sampling frequency. The following time and frequency-domain features are extracted from each sensor signal in the window:

#### Time-Domain Features

- **Mean**: Represents the average value of the signal within the window.

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{4.1}$$

- **Standard Deviation**: Measures the variability or dispersion of the signal values.

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2} \tag{4.2}$$

- **Minimum**: Captures the lowest signal value in the window.

$$x_{\min} = \min(x_1, x_2, \ldots, x_n) \tag{4.3}$$

- **Maximum**: Captures the highest signal value in the window.

$$x_{\max} = \max(x_1, x_2, \ldots, x_n) \tag{4.4}$$

- **Median**: Provides the central (middle) value of the sorted signal.

$$\text{median}(x) = \begin{cases} x_{\left(\frac{n+1}{2}\right)} & \text{if } n \text{ is odd} \\ \frac{1}{2}\left(x_{\left(\frac{n}{2}\right)} + x_{\left(\frac{n}{2}+1\right)}\right) & \text{if } n \text{ is even} \end{cases} \tag{4.5}$$

- **Energy**: Quantifies the signal strength by summing the squared values.

$$E = \sum_{i=1}^{n} x_i^2 \tag{4.6}$$

- **Kurtosis**: Describes the tail heaviness of the signal distribution.

$$\text{Kurt}(x) = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^4}{\left(\frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2\right)^2} \tag{4.7}$$

### Frequency-Domain Features

Let $X_k$ be the magnitude of the $k$-th component of the real-valued Discrete Fourier Transform (RFFT) of the signal.

- **Peak Frequency**: Identifies the dominant frequency component in the window.

$$f_{\text{peak}} = f_s \cdot \frac{k_{\max}}{n}, \quad \text{where } k_{\max} = \arg\max_k |X_k| \tag{4.8}$$

- **Peak Amplitude**: Represents the amplitude of the most dominant frequency.

$$A_{\text{peak}} = \max_k |X_k| \tag{4.9}$$

## 4.4 Preliminary Models Classification

In order to establish a baseline classification for HAR, a first round of test classification was deployed. This work was carried out using the PyCaret library [116]. This library is an open source software based in python that allows for an automatic and faster deployment of classification and regression models, utilizing a "low code" approach.

This software implements already established open source libraries that are well respected in the field of machine learning, like scikit-learn. It introduces layers that simplify the deployment of these models like pre-processing of data, comparison between models, cross validation, hyper-parameter tuning, etc. One drawback is that it is difficult to visualize what is under the hood, like the automatic feature engineering algorithm lowering the reproducibility of these techniques.

After extracting the features as discussed before, the train data is used to configure PyCaret. In this step it was important to define the normalization, removal of outliers and the pre-processing to false since this is all done previously and this would change and impact the classification. Then the training data was split again in training and validation, allocating 10% to validation (representing 8% of the total dataset). Then the 5 best models are selected based on validation data from the ten fold cross validation results. After, these 5 models are finalized by retraining in the whole training data. Some metrics are calculated for comparison and will be discussed further in this chapter. The overall best classifier becomes the base model for the rest of the methodology. This model was chosen based on predefined metrics discussed in chapter 5

With this first methodology the 5 best classifiers will be fixed for the remaining datasets. These classifiers are fixed to evaluate the RWHAR dataset, and potentially any other HAR dataset, as it standardize the processes between datasets. This permits for the direct comparison and generalization for different experimental conditions. This also makes the methodology process faster and more efficient since these datasets have comparable activities it is expected that the same models are selected for both datasets.

## 4.5 Baseline model

After identifying the best-performing classifiers, the top-ranked model was selected as the baseline model for all subsequent comparisons. This selection is used to establish a strong model so that future modifications can be compared to this already optimized model. The baseline model represents the highest performance achieved under standard configurations using the initially extracted and pre-processed features, without the application of additional tuning or specialized techniques.

From this point forward, all subsequent steps in the machine learning pipeline including feature selection or hyperparameter optimization are guided by the primary objective of surpassing the baseline model's performance. Improvements may be reflected in enhanced classification metrics like accuracy or f1 score, reduced computational costs, or faster training and testing times. Using a validated model as a starting point ensures that any performance improvements are due to real methodological enhancements rather than random variation. This supports a more reliable evaluation for HAR systems.

## 4.6 Dimensionality reduction techniques

With the objective of optimizing the performance of the previous model and at the same time reduce the complexity of the features extracted a comparison of three distinct dimensionality reduction techniques was performed: PCA (Principal Component Analysis), RFE (Recursive Feature Elimination) and SelectKBest. Each technique was analysed with the same base classifier.

### 4.6.1 Principal Component Analysis (PCA)

The PCA is a multivariate statistical technique used to reduce the dimensionality of data. It transforms the original variables in new axes that are called principal components (PC). These are linear combinations of the original variables. Each PC is aligned in order to hold as much variance it is possible [117].

This way the first principal component has the most variance and successive PC's hold less and less variance. For this work it was iterated using of 2 to 53 PC's (as much as the amount of features in the data) and for each iteration the base classification model was used to extract the accuracy of the test data. These results were gathered and used in a graph to visualize the impact of adding one PC.

### 4.6.2 Recursive Feature Elimination (RFE)

The RFE is wrapper technique for feature selection. This algorithm uses a machine learning model to evaluate the importance of each variable and, by iterating through every feature it eliminates the least important until the desired number of features is reached. This processes is guided by the performance of the ML model being used [118].

For this research it was selected the base model for the classification of the importance of the features and again iterated to 53 features extracting the performance of each iteration for further analysis.

### 4.6.3 SelectKBest

The SelectKBest algorithm selects the k best features of a future space, using a statistical metric to classify each feature like F1-score for example [119]. Each feature is evaluated independently for the target classification.

### 4.6.4 Selection of the best feature reduction algorithm

A comparative analysis is done after executing the three feature reduction techniques. For each algorithm the best performed model was detected based on accuracy and the number of features/ principal components used and the best one was selected. Additionally, considering the original accuracy of the base model trained with all features, a tolerance gap of 0.2% was established. This was done to take into consideration the reduction of computational cost in detriment of the accuracy of the model. By assuming a 0.2% tolerance gap, it is ensured that the accuracy does not lose its importance for a worse performing model with a smaller number of features. If this condition is met, then the smaller feature space is selected and applied to the whole dataset.

## 4.7 Hyperparameter tuning

With the intention of maximizing the classification process of the 5 best models discussed before, and guarantee the robustness of this methodology, an optimization method was implemented for the models known as hyperparameter tuning. A manual grid search is implemented for each classification algorithm with the help of the PyCaret library. The grid search iterates through every possible configuration in the proposed parameters to find the best possible combination.

### 4.7.1 LightGBM

For the LightGBM [36] 4 distinct parameters are controlled. The `learning_rate` controls the contribution of each tree to the process of boosting. The `max_depth` parameter limits the maximum depth of trees, the `num_leaves` defines the number of leaves in each tree and the `min_child_samples` specifies the minimum number of samples to create each leaf.

- `learning_rate`: {0.01, 0.05, 0.1, 0.2}
- `max_depth`: {3, 5, 7, -1}
- `num_leaves`: {15, 31, 63}
- `min_child_samples`: {10, 20, 30}

### 4.7.2 Random Forest

For the Random Forest [120], different parameters are tested like the `n_estimators` that defines the number of trees in in the forest, the `max_depth` essentially does the same job as in LightGBM, the `min_samples_split` is the minimum number samples in order to split an internal node and the `min_samples_leaf`, the minimum number required in a leaf node.

- `n_estimators`: {100, 200, 300}
- `max_depth`: {None, 10, 20}
- `min_samples_split`: {2, 5}
- `min_samples_leaf`: {1, 2}

### 4.7.3 Extra Trees

Similarly to the Random Forest the parameters in Extra Trees [33] have the same purpose. This model offers more randomness that can be useful for a better generalization.

- `n_estimators`: {100, 200}
- `max_depth`: {None, 10, 20}
- `min_samples_split`: {2, 4}
- `min_samples_leaf`: {1, 2}

### 4.7.4 Gradient Boosting Classifier

Similar to the LightGBM, the `learning_rate` and `max_depth` have the same characteristics. The `n_estimators` represents the number of boosting stages and the `subsample` represents the fraction of training data used for boosting iterations in GBC [121].

- `learning_rate`: {0.01, 0.1, 0.2}
- `n_estimators`: {100, 150}
- `max_depth`: {3, 5}
- `subsample`: {0.8, 1.0}

### 4.7.5 K-Nearest Neighbors

Finally, for the KNN [122], the `n_neighbors` specifies the number of neighbours to be included in classification. `Weights` represents if the distance of the neighbours influences the classification process. Lastly, `algorithm` determines which algorithm is used to calculate the distance to neighbours.

- `n_neighbors`: {3, 5, 7}
- `weights`: {uniform, distance}
- `algorithm`: {auto, ball_tree, kd_tree}

## 4.8 Evaluation metrics

The models built were tested for Accuracy, Recall and F1-score. These metrics serve to characterize the classification results.

### Accuracy

Proportion of correctly predicted observations out of the total observations (Good for balanced data).

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{4.10}$$

### Recall

Evaluated the proportion of actual positives correctly identified. This evaluation metric is good in applications when missing a positive case is critical but does not take into account false positives.

$$\text{Recall} = \frac{TP}{TP+FN} \tag{4.11}$$

### F1 Score

Harmonic mean of Recall and Precision, this metric is better for imbalanced data. Precision evaluates the correctness of the positive results.

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4.12}$$

Where:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{4.13}$$

# Chapter 5

# Results

## 5.1 Filtering results

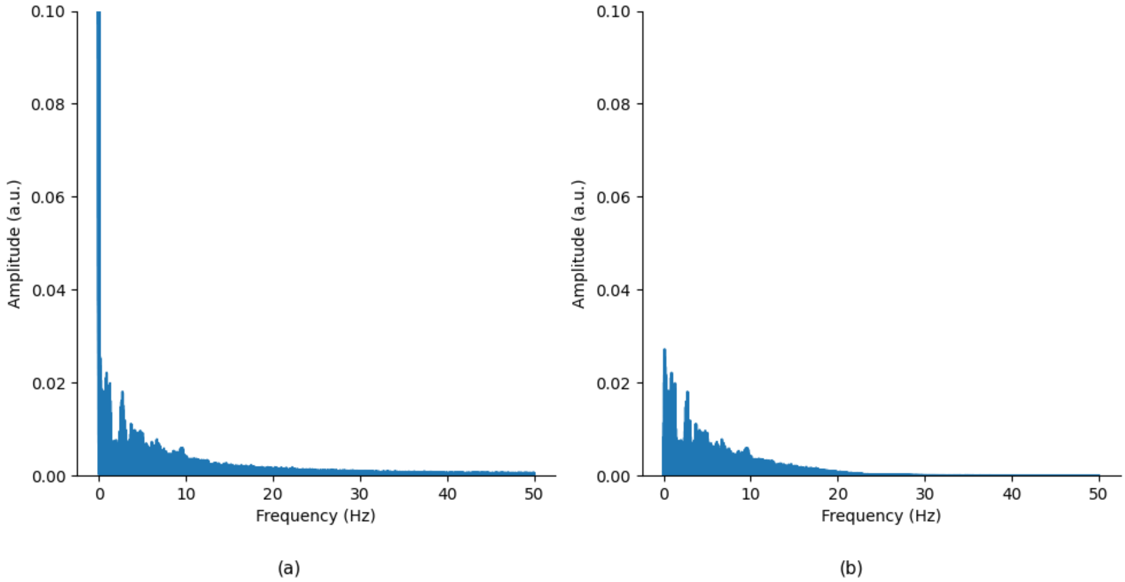
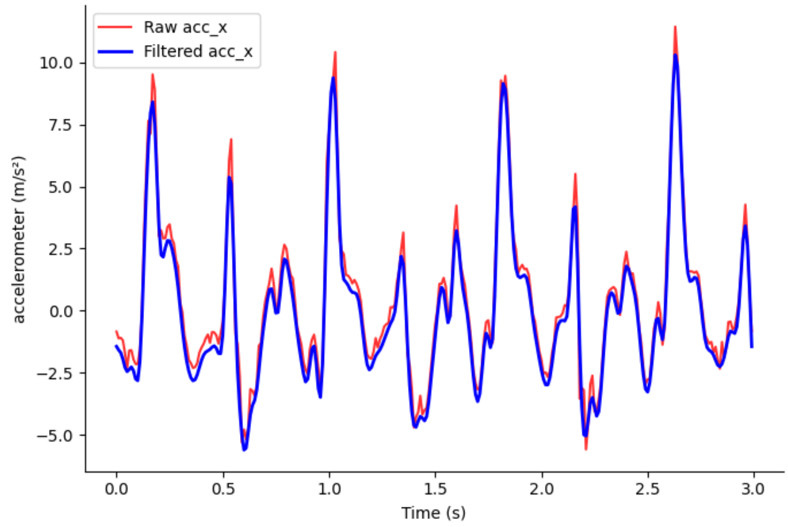


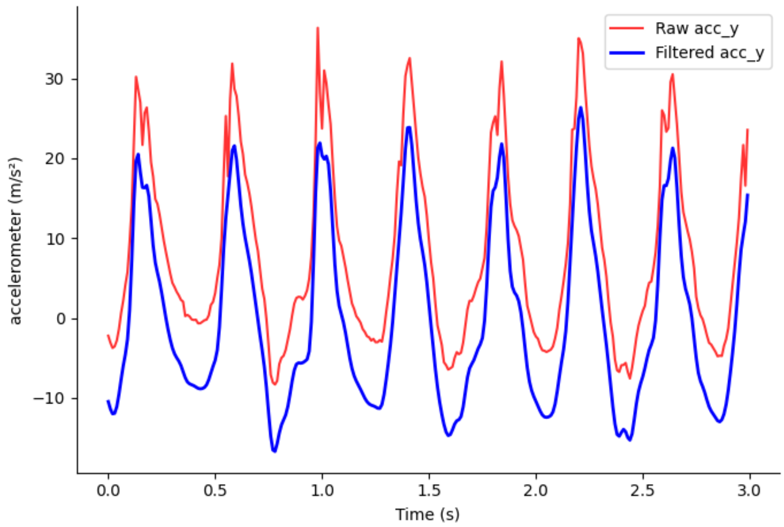

Figure 5.1: Fast Fourier transform of accelerometer data before and after bandpass filter. (a)- before filtering (b)- after filtering

The frequency spectrum obtained through a Fast Fourier Transform (FFT) is presented in Figure 5.1. The power spectrum of the original signal is presented in the left panel, while in the right panel, the resulting power spectrum, after filtering, is shown. In Figure 5.1 (a), corresponding to the original signal, it is possible to see that there is a peak of amplitude around frequency 0 Hz. This peak indicates the gravity component, representing a constant force affecting the sensors. Higher frequency components are witnessed above 20 Hz although, as previously mentioned, these are unlikely to be of biomechanical nature.

After applying the high-pass filter to the signal, as it can be seen in Figure 5.1(b), the 0 peak is significantly attenuated. For the low-pass filter, there is also an attenuation of the signal.




(a) x axis (axis pointing posterior to anterior relative to the subject)




(b) y axis (axis pointing inferior to superior)

Figure 5.2: Accelerometer data of bandpass filter vs. unfiltered data in the PAMAP2 dataset, participant 102 performing rope jumps for 3 seconds

Figure 5.2 shows the acceleration signal in the x axis pointing forward 5.2a and the signal for the y axis pointing upwards 5.2b both for the pre and post processed using the bandpass filter.

In Figure 5.2a the application of the bandpass filter results in the attenuation of the high frequency component of the signal. This process removes noise and motion artifacts especially sensor vibrations and environment interference from the signal leaving only human movements. Since this image corresponds to the axis oriented anteriorly to the subject, the influence of gravity on the signal is minimal; therefore, the cutoff frequency of 0.05 Hz has a negligible effect on the signal. The preservation of the primary frequency components, associated with human motion, indicates that the cutoff frequency is appropriate for isolating relevant activity-related dynamics without significantly altering the essential characteristics of the signal. The filtered and unfiltered signals for the Y axis (axis that point upwards) can be seen in Figure 5.2b. A baseline shift of around 9.81 m/s$^2$, akin to a DC removal, is perceived in the filtered signal, indicating the effective removal of the gravitational component.

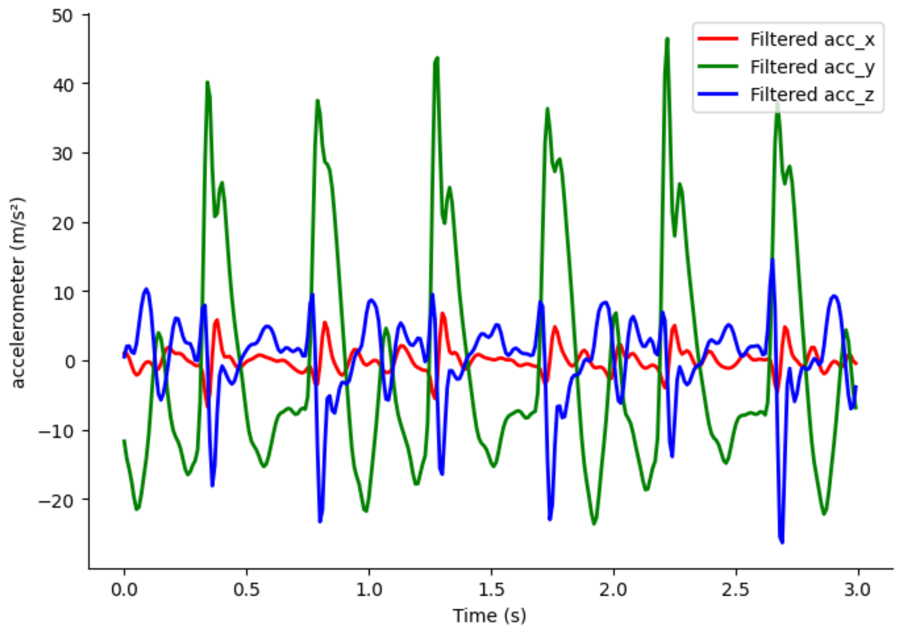



Figure 5.3: Three axis accelerometer data after bandpass filtering for participant 101 while doing rope jumps for a total of 3 seconds with Y axis being the vertical axis, X axis anterior to posterior axis and Z axis medial to lateral axis

In Figure 5.3 the combination of the 3 axes is seen.The Butterworth bandpass filtering process effectively smooths the signal and removes the gravity component, maintaining a well-defined output. In this exercise, it is possible to verify the six amplitude peaks in the Y axis suggesting the acceleration movement regarding each rope jump.For the X axis, there is minimal acceleration, which is expected given that rope jumping requires little force in that direction. Around the Z axis, there is less acceleration amplitude than in the Y axis but more than the X axis, this can be explained due to equilibrium forces being applied and incorrect landing wilie performing these jumps.

## 5.2 Preliminary Models Classification

After performing the filtering, and feature extraction through sliding windows technique, the training data is used to compare the 5 best models in the PyCaret library, and this is done by cross validating with a 10 fold cross validation. As seen in Table 5.1 the 5 best classifiers were Light Gradient Boosting Machine, Extra Trees, Random Forest, Gradient Boosting Classifier and K-Nearest-Neighbours. These models were evaluated based on accuracy, recall, F1 score and training time. The lightGBM is the best classifier with an accuracy of 0.977, a recall of 0.9668 and an f1 score of 0.9733 toping every metric regarding performance. One drawback of this classifier is the training time being the second highest, and, since this is the training time per fold, the real total time is 10*TT totalling at 15.06 s. GBC has the forth best classification and is the classifier that takes the longest to train with a significantly higher time (8.073 s per fold). This table allowed to select the optimal base model and, regardless of extra training time, it was decided on the lightGBM. This models is used for future performance comparisons.

Table 5.1: Validation metrics after 10 fold cross validation of the 5 best classifiers in PyCaret. Best performing classifier is highlighted (T.T. stands for training time and is the average of the 10 folds)

| **Model** | **Accuracy** | **Recall** | **F1 Score** | **TT (s)** |
|---|---|---|---|---|
| LGBM Classifier | 0.977 | 0.967 | 0.973 | 1.506 |
| Extra Trees Classifier | 0.972 | 0.963 | 0.970 | 0.129 |
| Random Forest Classifier | 0.969 | 0.961 | 0.967 | 0.321 |
| Gradient Boosting Classifier | 0.967 | 0.958 | 0.964 | 8.073 |
| KNN Classifier | 0.946 | 0.940 | 0.948 | 0.943 |

By selecting the lightGBM as the base model, this classifier is retrained with the whole training dataset and used to classify the test data. It was then established a base accuracy of 0.9275 for the PAMAP2 dataset and 0.9025 for the RWHAR dataset. In Table 5.2 it is possible to see the accuracy, recall and F1 score for both datasets.

Table 5.2: Performance metrics comparison between PAMAP2 and RWHAR datasets after selection of best performing base model (lightGBM)

| **Dataset** | **Accuracy** | **Recall** | **F1 Score** |
|---|---|---|---|
| PAMAP2 | 0.928 | 0.914 | 0.922 |
| RWHAR | 0.903 | 0.903 | 0.900 |

## 5.3 Dimensionality Reduction Results

To optimize the performance of the base model while reducing the dimensionality of the feature set, three different dimensionality reduction techniques were compared: PCA, RFE, and SelectKBest. Each method was evaluated using the same classification model, and the accuracy was measured as a function of the number of selected features/components, ranging from 2 to 53.

In Figure 5.4 it is possible to see the performance of the 3 dimensionality reduction techniques for the PAMAP2 dataset. For the PCA the accuracy improves faster for the the first 10 components reaching a plateau in that region with the maximum accuracy reaching around 88% until it starts decreasing at a steady rate finalizing the 53 principal component with an accuracy of around 85%.

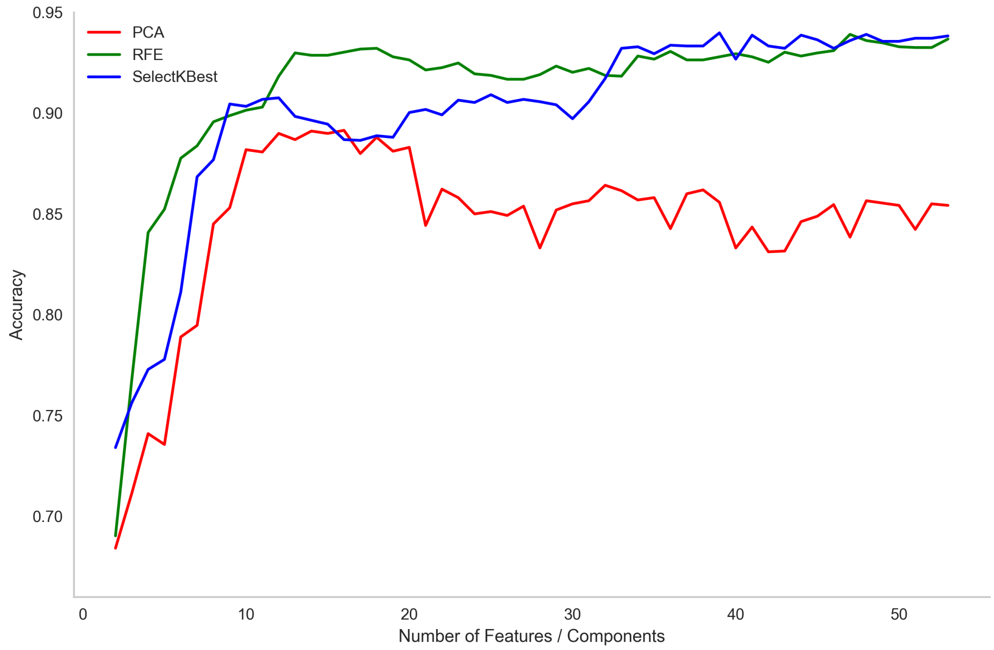



Figure 5.4: Accuracy vs. number of features/components using PCA (red line), RFE (green line) and SelectKBest (blue line) on the PAMAP2 dataset.

For the RFE, it reaches the top accuracy at around 13 features and consistently maintains the maximum accuracy for the remaining features. This is also the first classifier to reach 85% accuracy with just 4 features. This highlights the value of the feature reduction techniques allowing for less training time and less computational power.

For the SelectKBest technique, with fewer features it is able to reach higher accuracies than the other example reaching 90% accuracy with less than 10 features. However, it stabilizes around that accuracy remaining lower than the RFE until it reaches 30 features, where it increases and tops off at the maximum value with an accuracy of almost 95%. This was also the feature reduction technique that achieved the higher accuracy.

For the RWHAR dataset the feature reduction techniques are seen in Figure 5.5. For the PCA it starts with the worst accuracy of the three and increases steadily from 0.65 until 0.85 with no sharp increases or decreases. This is the worst performing of the three, something that can also be verified in the PAMAP2 dataset.

For the RFE technique it starts with around 0.80 accuracy for 2 features and rapidly increases to around 0.90 features at the 15 features mark, after this the accuracy stays practically the same with only minor changes for the remaining features.

Finally, the SelectKBest technique has the lowest and one of the highest performances for different features used. It starts with a relatively high accuracy of 0.75 with 2 features but declines to around 0.50 with 4 features this can be due to this model not considering interactions between features, so contextual correlated information can be missed. The accuracy recovers sharply to round 0.90 around 10 features and increases very slightly until around 50 features.

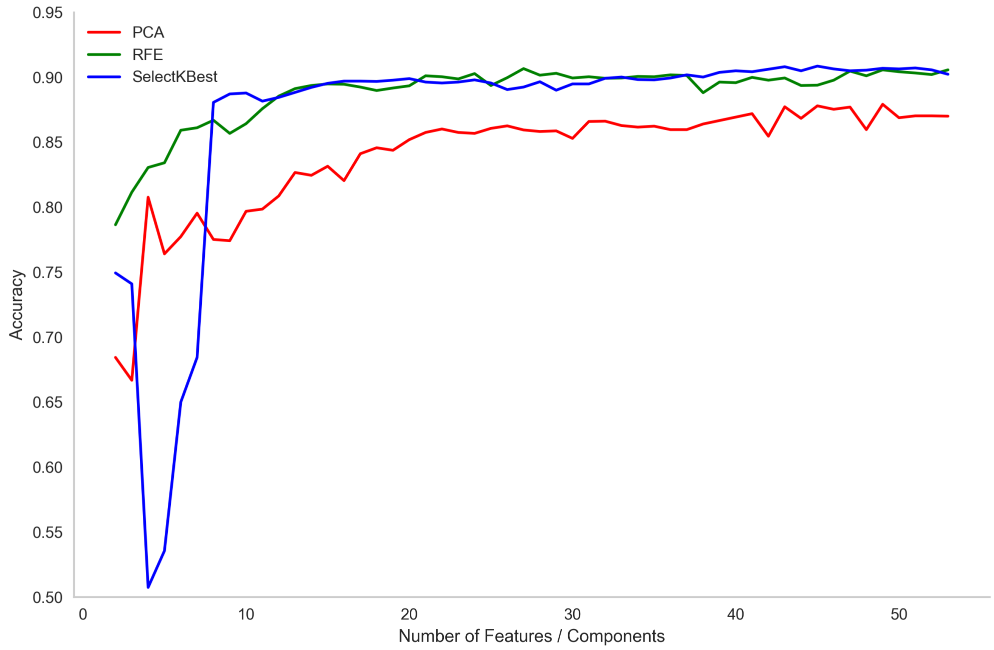



Figure 5.5: Accuracy vs. number of features/components using PCA (red line), RFE (green line) and SelectKBest (blue line) on the RWHAR dataset.

Considering the base model and the application of dimensionality reduction techniques, it is evident that reducing the number of features offers advantages both for the PAMAP2 and RWHAR datasets. Notably, two out of the three methods (RFE and SelectKBest) resulted in higher accuracy compared to the full feature set. Additionally, by decreasing the number of features, the training and testing times are significantly reduced, contributing to more efficient model development without compromising and in some cases improving performance of the models. In table 5.3 is the concrete accuracy and number of features for the three dimensionality reduction techniques and the base model for both the PAMAP2 and RWHAR datasets.

Additionally, in Table 5.4 it is possible to see the ten best features extracted from each dataset with both the RFE and SelectKBest algorithms. PCA was discarded from this table due to its inability to be represented in the same way. In the RWHAR dataset 4 features were common to both techniques: `acc_x_max`, `acc_y_max`, `acc_y_min`, `acc_y_peak_amp`. For the PAMAP2 dataset the same thing happened but for only 3 features: `acc_y_std`, `gyr_x_std and gyr_y_peak_amp`. This suggests that these features are more robust and generalizable across each dataset, independently of the feature selection technique used.

One interesting aspect is that the RWHAR dataset seems to rely more on accelerometer data specially from the Y axis, while the PAMAP2 dataset seems to rely more on gyroscope data, at least for the common features to both feature selection techniques. Additionally SelectKbest features seem to be more selective to one type of signal focusing more on data from the Y axis for both datasets while for the RFE technique the selection seems more spread with a more diverse signal approach.

Table 5.3: Comparison of dimensionality reduction techniques for PAMAP2 and RWHAR datasets

| Dataset | Method | Best Accuracy | Number of Features |
|---|---|---|---|
| PAMAP2 | Base Model | 0.928 | 53 |
| | PCA | 0.891 | 16 |
| | RFE | 0.939 | 47 |
| | SelectKBest | 0.940 | 39 |
| RWHAR | Base Model | 0.903 | 53 |
| | PCA | 0.879 | 49 |
| | RFE | 0.907 | 27 |
| | SelectKBest | 0.909 | 45 |

## 5.4 Hyperparameter tuning results

In this section the results for the hyperparameter tuning of the 5 best models and the respective best parameters are presented in table 5.5 and 5.6. The models were evaluated based on accuracy, recall and F1-score. In this segment, The hyperparameter configurations were evaluated by the primary metric as accuracy, since this metric gives insights to the overall performance of the model. After this, the training time was taken into consideration because it is necessary for the model to perform as efficiently as possible with large datasets and multiple classifications. Finally, the last parameter for selection were the recall and F1-score, for complementing the first parameters.

In table 5.5 corresponding to the results obtained in the PAMAP2 dataset, the models GBC and lightGBM have the highest F1-score (0.9401) and highest recall (0.9321). However the lightGBM classifier has the highest accuracy (0.9424) resulting in an good overall classification. The Random Forest has a lower accuracy than the 2 previous models but still a higher accuracy than the Extra Trees model, something that can be expected since the Extra Trees model has a more random approach in selecting the optimal parameters. The KNN model had the lowest accuracy, achieving just 0.8987, almost 0.05 lower accuracy than the top model, something that can be attributed to the lower complexity of the model. From this table it is possible to conclude that the LightGBM should be the chosen since it it was the best performing model on all parameters. Additionally, as seen in Table 5.1 the training time for the GBC model for each fold is way superior when comparing to the lightGBM model, making it a disadvantage for implementation.

Table 5.6 presents the same models for the RWHAR dataset. For this dataset, the results were more spread than PAMAP2 models. The LightGBM had the best accuracy (0.9092), then the GBC had the best F1-score (0.9081) and the Extra Trees classifier had the best recall. Random Forest had comparable results with ET, which is expected based on the similar workflow of these two models. Again, KNN had the worst results with an accuracy of just 0.8411 consolidating the

Table 5.4: Selected features using KBest and RFE for the RWHAR and PAMAP2 datasets

| Feature | RWHAR-KBest | RWHAR-RFE | PAMAP2-KBest | PAMAP2-RFE |
|---|---|---|---|---|
| acc_x_max | X | X | | |
| acc_x_mean | | | | X |
| acc_x_min | | X | | |
| acc_x_peak_amp | | | | X |
| acc_x_std | X | | X | |
| acc_y_energy | X | | X | |
| acc_y_kurtosis | | | | X |
| acc_y_max | X | X | X | |
| acc_y_mean | | | | X |
| acc_y_min | X | X | X | |
| acc_y_peak_amp | X | X | X | |
| acc_y_std | X | | X | X |
| acc_z_max | | | | X |
| acc_z_min | X | | | |
| acc_z_peak_amp | | X | | X |
| acc_z_std | X | | | |
| gyr_x_max | | X | X | |
| gyr_x_min | | X | | |
| gyr_x_std | | | X | X |
| gyr_y_min | | X | | |
| gyr_y_peak_amp | X | | X | X |
| gyr_y_std | | | X | |
| gyr_z_energy | | | | X |
| gyr_z_max | | X | | |

**Note:** Blue cells indicate the same feature is used by the two techniques for the RWHAR dataset (KBest e RFE). Green cells indicate the same but for the PAMAP2 dataset.

theory before that this model might be too simple for this classification. With this in mind, the model selected for evaluation was the lightGBM with the highest accuracy and comparable recall and F1-score to the GBC with a significantly less training time. This also supports the results in the PAMAP2 dataset making this the best overall model.

Table 5.5: Classification metrics for each model's best parameters in PAMAP2 dataset. Highlighted is the best performing metric for each category

| Model | Parameters | Accuracy | Recall | F1-score |
|---|---|---|---|---|
| lightgbm | 'learning_rate': 0.2, 'max_depth': -1, 'num_leaves': 31, 'min_child_samples': 10 | 0.942 | 0.932 | 0.940 |
| gbc | 'learning_rate': 0.1, 'n_estimators': 150, 'max_depth': 5, 'subsample': 1.0 | 0.9413 | 0.932 | 0.940 |
| rf | 'n_estimators': 200, 'max_depth': 20, 'min_samples_split': 5, 'min_samples_leaf': 1 | 0.929 | 0.923 | 0.930 |
| et | 'n_estimators': 200, 'max_depth': None, 'min_samples_split': 4, 'min_samples_leaf': 2 | 0.915 | 0.912 | 0.919 |
| knn | 'n_neighbors': 7, 'weights': 'distance', 'algorithm': 'ball_tree' | 0.899 | 0.899 | 0.906 |

Table 5.6: Classification metrics for each model's best parameters in RWHAR dataset. Highlighted is the best performing metric for each category

| Model | Parameters | Accuracy | Recall | F1-score |
|---|---|---|---|---|
| lightgbm | 'learning_rate': 0.2, 'max_depth': -1, 'num_leaves': 31, 'min_child_samples': 10 | 0.909 | 0.911 | 0.906 |
| gbc | 'learning_rate': 0.1, 'n_estimators': 150, 'max_depth': 5, 'subsample': 0.8 | 0.905 | 0.915 | 0.908 |
| et | 'n_estimators': 200, 'max_depth': None, 'min_samples_split': 4, 'min_samples_leaf': 1 | 0.900 | 0.920 | 0.901 |
| rf | 'n_estimators': 200, 'max_depth': None, 'min_samples_split': 2, 'min_samples_leaf': 1 | 0.900 | 0.918 | 0.891 |
| knn | 'n_neighbors': 7, 'weights': 'distance', 'algorithm': 'ball_tree' | 0.841 | 0.854 | 0.829 |

## 5.5 Final Classification Model results

Based on the previous section assumptions, further performance evaluations were done to the lightGBM model since this was the best performing classifier. The two confusion matrices presented in Figures 5.6 and 5.7 illustrate the classification performance of this model applied to the PAMAP2 and RWHAR datasets. Each confusion matrix compares the true activity labels (rows) with the predicted activity labels (columns), providing insight into the classifier's ability to distinguish between activities.

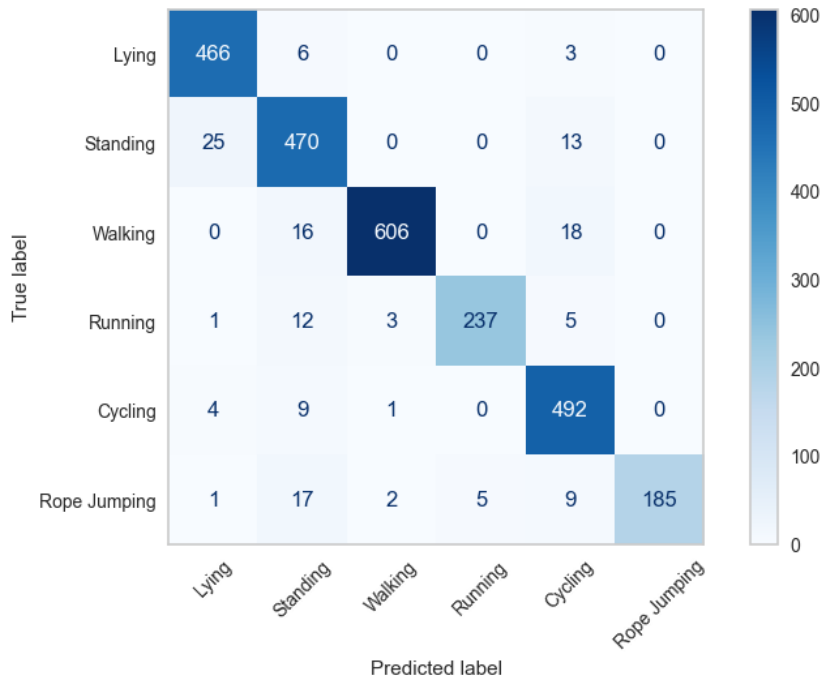



Figure 5.6: Confusion matrix for the PAMAP2 dataset after hyperparameter tuning for the test data.

The confusion matrix for the PAMAP2 dataset demonstrates interesting information about the correct classifications and what activities each true label is confused for. For starters, no activity outside of rope jumping was confused for rope jumping, making this activity the most distinguishable activity in the global scale. Regardless of this, some rope jumping activities were misclassified for other, with the most predominant being standing (17 misclassifications). This makes sense because there should be periods in the datasets where the participant is not jumping, probably for recovering after a missed rope jump where the subject is just standing.

Cycling, running and walking are fairly classified with minimal misclassification interchangeably, they all share the same confusion with standing, which again could be explained by recovering times. Additionally, standing has the biggest misclassification with lying. Due to the gravity removal factor through filtering, both activities the distinguishing factor becomes only the orientation since both activities produces minimal accelerations. Finally, lying has minimal misclassifications with other activities.

For the RWHAR dataset the jumping activity only has some minor misclassifications for running, even though this activity has the lowest prevalence in the dataset with only 132 windows of features. For the lying activity, one of the most prevalent activities, there was minimal misclassification occurring mostly at standing predicted label, which as discussed before seems to be fairly common. Standing and running seem to be the activities with the most confusions in

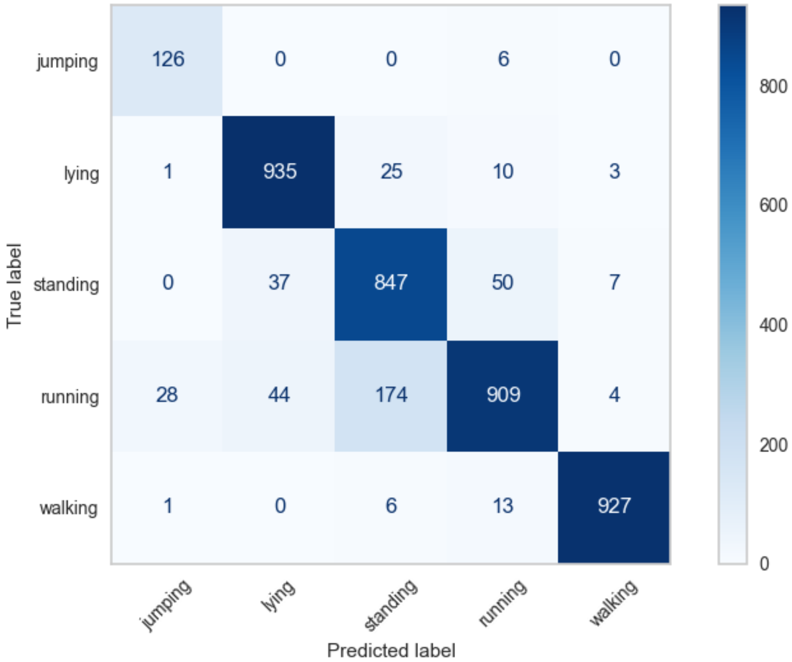



Figure 5.7: Confusion matrix for the RWHAR dataset after hyperparameter tuning for the test data.

the prediction. 50 standing activities were misclassified as running while 174 running activities were confused with standing. Walking on the other hand has a good performance with only 13 misclassifications as running and minor confusions for other activities.

In summary, both these datasets show promising results without major misclassifications. The PAMAP2 dataset with one additional activity seems to perform better than the RWHAR dataset on all activities except for walking. The confusion between running and standing is a common issue in both datasets which might be due to certain movement transitions or posture maintenance. However, both datasets show a degree of imbalanced data, specially regarding jumping/ rope jumping activities. This can impact the classification process because the model is not trained on a well balanced dataset contributing to less represented activity.

The classification reports shown in Table 5.7 and 5.8 provide a complementary and quantitative analysis to the confusion matrices providing class-wide metrics. By analysing the PAMAP2 report walking achieved the highest F1-score (0.9681) with a even higher precision (0.9902) culminating in the dataset being extremely confident in predicting this class. Rope jumping as discussed before offers a perfect precision but a fairly low recall meaning that activities are only classified as rope jumping if the model is sure this activity is rope jumping, but, this leads to misclassifying some of these windows as other activities. Standing also had the lowest F1-score due to, as mentioned before, some activities being classified as lying. Overall the activities are well classified and yield good results both for the weighted and macro F1-score coming in at 0.9425 0.9395 respectively.

For the RWHAR dataset walking achieved the highest values across the board with 0.9851

Table 5.7: Classification report for the best model on the PAMAP2 dataset

| **Activity** | **Precision** | **Recall** | **F1-score** |
|---|---|---|---|
| Lying | 0.938 | 0.981 | 0.959 |
| Standing | 0.887 | 0.925 | 0.906 |
| Walking | 0.990 | 0.947 | 0.968 |
| Running | 0.979 | 0.919 | 0.948 |
| Cycling | 0.911 | 0.972 | 0.941 |
| Rope Jumping | 1.000 | 0.845 | 0.916 |
| **Macro Avg** | 0.951 | 0.932 | 0.940 |
| **Weighted Avg** | 0.945 | 0.942 | 0.943 |

Table 5.8: Classification report for the best model on the RWHAR dataset

| **Activity** | **Precision** | **Recall** | **F1-score** |
|---|---|---|---|
| Jumping | 0.808 | 0.955 | 0.875 |
| Lying | 0.920 | 0.960 | 0.940 |
| Standing | 0.805 | 0.900 | 0.850 |
| Running | 0.920 | 0.784 | 0.847 |
| Walking | 0.985 | 0.979 | 0.982 |
| **Macro Avg** | 0.888 | 0.916 | 0.899 |
| **Weighted Avg** | 0.905 | 0.902 | 0.901 |

precision, 0.9789 recall and 0.9820 f1-score. This complements the other dataset and the confusion matrix seen before as the best classified activity. Jumping and Standing show the lowest F1-scores (0.8750 and 0.8500, respectively), mostly due to reduced precision (0.8077 and 0.8051), reflecting false positives likely with high-motion activities such as running. Even though, running has a strong precision with 0.9200, there is a decline in recall due to the model misclassifying the activity. This dataset also confirms the good classification capacities of this model with a macro average F1-score of 0.8987 and a weighted average score of 0.9010.

# Chapter 6

# Discussion

## 6.1 Overview and Purpose of the Study

This study had the objective of evaluating the effectiveness of different classification algorithms based on machine learning techniques for human activities classification. The analysis was performed using only one IMU in the upper torso area with accelerometer, gyroscope and electrocardiogram data.

In order to accomplish this study, two datasets commonly addressed in literature are used to support the proposed analysis. The PAMAP2 and RWHAR datasets use different number of sensors, different sampling frequencies and different experimental strategies. Additionally, the two datasets contain different but partially overlapping activity sets, and for the purposes of this study, it was necessary to ensure compatibility between them to enable meaningful comparisons. For this reason, it was selected a subset of activities from each dataset with the focus on fundamental activities common in the majority of physical activity and daily living. Walking, running, standing, and lying activities were selected for both datasets. Additionally, for the PAMAP2 dataset, rope jumping and cycling were also selected, for being activities with higher physical demand. For the RWHAR dataset it was selected jumping to match the previous dataset. This selection of activities wasn't arbitrary since this study evaluates the fundamental movements within a sport, from activities of high exertion of energy like cycling, running and jumping, to posture and recovery phases for example lying and standing.

This study includes steps for preprocessing of data, with noise reduction and gravity component removal, feature extraction based on sliding windows, primary evaluation of classification models to establish a baseline for comparison, dimensionality reduction techniques, hyperparameter tuning and final model evaluation. With this methodology, evaluating the different machine learning classification models and finding the best performing algorithm was the main objective. Additionally, it seeks to address the complexities associated with this type of classification in order to develop a generalizable technique that meets the requirements across different scenarios and a variety of activities.

## 6.2 Signal Pre-processing

As discussed in the results chapter the first step of the methodology after setting the requirements for the datasets was the preprocessing of the signals. For this step a bandpass filter with cutoff frequencies between 0.05 Hz and 20 Hz was implemented. This effect can be seen through the frequency spectrum in Figure 5.1. The high pass filter at 0.05 Hz shows to be efficient in attenuating the gravity component from the axes that are affected by this force without compromising the signal structure. This works because the gravitational component of the signal behaves as DC voltage meaning that it has a frequency of basically 0 Hz and can easily be removed by this filter [123].

The low pass filter with the frequency of 20 Hz is able to smooth the signal removing noise due to environment interference, sensor vibrations and other external factors. However, as it is possible to see in Figure 5.1 it does not eliminate every frequency beyond the cutoff point, which can be explained by the non ideal performance of the filters.

This technique demonstrated good results by filtering the noise and gravitational effect but keeping the human activity movement. Therefore, the signal to noise ratio is improved, contributing to a more efficient final signal. This step ensures that the signals remain unaffected by gravitational effects, even if the sensors are flipped in other datasets, therefore enhancing the generalizability and robustness of the approach.

## 6.3 Dimensionality Reduction Evaluation

The application of dimensionality reduction techniques had the main objective of improving the performance of the base model of LightGBM, lowering the number of computed features in order to lower computational costs without compromising, or even enhancing the accuracy score. This can be very beneficial in scenarios where computer efficiency is necessary like in instances where these metrics could be calculated in real time.

Both in the PAMAP2 and RWHAR dataset the results demonstrate the possibility of obtaining higher accuracies by using less features. The RFE and SelectKBest feature reduction techniques were the most useful techniques achieving good results even with minimal features, where above 10 features both for RWHAR and PAMAP2 seem to achieve above 85% of accuracy. With higher number of features both techniques also pass the base model as it can be seen in 5.3. Additionally, after 10 features, these techniques show a plateau effect with less than 5% increase in accuracy. This signifies the possibility of choosing the 10 features as a tipping point, reducing computational costs even more.

Even though, the PCA technique is widely used and documented in literature (see for example [73]) it showed limitations in the classification process and had a clear disadvantage either to the other 2 methods or the base model. This technique also hinders the interpretability of the results because it does not select certain features but it creates linear combinations of the original variables. This way this technique shows the lowest results of the three analysed methods.

One takeaway from this implementation is that for HAR models, more features do not directly correlate with higher performance of the model. Actually, having too many features can overload the model with conflicting information, leading to harder decisions and, therefore, reducing accuracy. By eliminating irrelevant or redundant features, not only can the model be optimized with less computational costs but, also it allows to achieve better results and enhance comprehensibility.

## 6.4 Hyperparameter tuning evaluation

The hyperparameter tuning revealed to be a significant step in improving and extracting the best outcome of the chosen models. Even though, classification models already had improvements after feature reduction techniques, the tuning also allowed to extract better values for each parameter which culminated in a better accuracy, recall and F1-score.

The selection of lightGBM as the final model, validated in both PAMAP2 and RWHAR, was done because it was the model with higher accuracy but also due to the balance of performance and smaller computational costs when comparing to the GBC model. Even though the GBC model had better results in F1 and recall for the RWHAR dataset, it also has much higher training times. This increases the processing demand on devices, potentially limiting real-time evaluations and requiring greater processing capacity for post-exercise analysis.

Additionally one interesting aspect is that lightGBM has the same hyperparameters for both datasets. This consistency suggests that the model presents a well structured, robust, and most importantly generalizable model, capable of adapting to different contexts, signal frequencies, and movements. This behaviour indicates a well defined model that in practice represents an advantage to the other models: the ability to utilize this model without fine tuning to different scenarios and reducing training times, simplifying the deployment to other datasets and therefore also minimizing computational costs. However, it is also important to take into consideration that these 2 datasets have almost the same activities and this fine tuned model might differ to entirely different datasets of HAR.

## 6.5 Final Model Discussion

The final model evaluation gives insights to the performance metrics of the overall pipeline. Additionally, the confusion matrices and classification reports reveal where the models lacks performance and what activities are misclassified. Analysis of the final model shows consistent prediction patterns with well defined activities like jumping, walking, lying and cycling, but also patterns of ambiguity in activities like running and standing where there is the biggest discrepancy in prediction. As mentioned before in the results chapter, since these are two completely different activities it was speculated that there could be instances of resting/recovering in the running activity. So, even though the true label comes out as running and the predicted label as standing/lying, in theory, the activity is being well classified due to its inaccurate labelling, this ends up hindering the results and lowering the performance metrics. The graph showing the differences in accelerometer data can be seen in figure A.1 in the appendix chapter, where the initial 1000 data points are from a run, and the following ones likely demonstrate a rest period. This makes this part of the dataset improperly segmented contributing to errors in the classification of the trained models since these errors are come from the dataset itself. It is also important to point out the fact that there is a substantial difference in class balance with activities like jumping/ rope jumping being around one quarter of samples of the average activity for the PAMAP2 dataset and almost one tenth of the average samples for the RWHAR. This can create problems with the classification process.

Regardless of these inconsistencies and structural differences in both datasets like different sampling rate, different activities and different number of subjects, the obtained performance confirmed the well structured approach for classification. Using a traditional machine learning model showed that these activities can be well classified with comparable results to deep learning algorithms using only one sensor placed in the chest area.

## 6.6 Model Interpretability

Beyond classification performance it is also important to take into consideration model interpretability and its practical viability. In fact some machine learning models are so complex mathematically that it is difficult to interpret what is happening inside the algorithm. One example is the usage of Neural Networks that can have multiple hidden layers. Even though this type of models can have excellent performances with minimal pre-processing, it is very difficult to understand the weight and interactions between each neuron inside a layer, creating a black-box effect, becoming difficult to understand the impact of the feature space and outcomes.

This way by testing machine learning models that are less complex and without these hidden layers, not only it is necessary to have a better understanding of the preparation of data, but it also allows to acquire information within each model. For example for the lightGBM model, it is possible to calculate the importance of each feature through the gain (how much a feature contributes to the reduce the total loss) and the total number a feature is used to divide a tree. Additionally, it is also possible to visualize the trees and the classification process through the voting system [124].

Furthermore, using simpler machine learning algorithms allows the possibility to have smaller datasets without compromising performance (since deep learning requires large amounts of data), have a better control with the preprocessing because in deep learning this done automatically and finally ML has lower computational costs making it faster to process new data and therefore have faster results.

## 6.7 Comparison with the State of the Art

The obtained results in this study suggest competitive performance when compared to the literature even surpassing some results. Table 3.3 of the state of the art chapter presents the comparison of different algorithms of machine learning applied to classification of daily living, fitness exercise and even fall detection with varying accuracies depending on the type of activity and references used.

In this research, the lightGBM model demonstrated a superior performance achieving an accuracy of 94.24% in the PAMAP2 dataset and 90.92% for the RWHAR. These results demonstrate superior or comparable results to the models evaluated. These accuracy scores exceed the average accuracy score for the SVM, KNN and Random Forest classifiers. Additionally, lightGBM does not appear to be used in the research conducted, which contributes to a relevant and original study.

Additionally, the Random Forest, KNN and GBC machine learning models used frequently in literature were tested in this research achieving 92.04%,94.13% and 89.87% respectively for the PAMAP2 dataset and 90.01%, 90.46% and 84.11% for the RWHAR dataset. These results, especially the ones for the first dataset fit within the range studied representing therefore accurate and comparable results to the literature. Furthermore, the Extra Trees classifier tested, even though not mention in literature, can be compared to the RF model in terms of algorithm methodology. This model also fits in the mentioned range for RF with less computational costs making this a good alternative to the RF classifier.

Moreover, many datasets used in this literature review use more than one placement for the sensors (per Table 3.1). This allows to have more information available for classification and can contribute heavily for activities that require limb involvement. For example, in the running activity, the lower limbs acquire more relevant information than the chest area making the legs the optimal location for the sensor. Since the PAMAP2 and RWHAR were segmented to only have chest information it was expected to lose valuable information, however, due to the extensive pre processing, feature analysis, and hyperparameter tuning, this was counterbalanced resulting in comparable results to studies that incorporate full body sensor analysis.

Additionally, to integrate this algorithm into the TexP@ct solution it is essential to deploy a classification algorithm capable of accurately recognizing movements based on data from a single location in the torso region. Given the anatomical proximity between the TexP@ct sensor placement and the chest-mounted sensors used in the PAMAP2 and RWHAR datasets, these datasets are well-suited for training and evaluating the model under comparable conditions. Moreover, the unbalanced dataset compromises the accuracy metric used for general evaluation of the model performance since it does not take into account the fact that it can hide poor performance on minority classes. This way for unbalanced dataset the F1-score metric should be the overall performance evaluation metric, due to its capabilities of capturing relevant information for the evaluation process. Regardless of this, both dataset acheive an F1-score of above 90%.

## 6.8 Real-world applications and Deployment Considerations

The development of this software will culminate in its integration with the TexP@ct platform and, in order to accomplish this, it is necessary to have some considerations with the data. First of all, the datasets used to test this algorithm were both lab related with well defined activities and protocols. Additionally, these datasets lack participant complexity, hindering generalization for different populations.

So, in order to successfully have good performance in a "real world" scenario, these activities should be done in an unconstrained environment with many subjects performing different sports with varying skill levels. This way, the sports can be broken down into corresponding fundamental activities. Even fundamental activities will vary from sport to sport. For example running in a football field will be different to a basketball court due to different dimensions, movement patterns and intensities, and materials. Different aspects also influence the acceleration of athletes performing the same activity, factors like sports shoes and type of ground will have this effect. Football cleats are entirely different to basketball shoes, the first ones have spikes and have a minimal cushioning while the later usually have bigger midsoles with a responsive effect to lower energy dissipation.

Furthermore, the conceptualized algorithm performs an extensive methodology to find the best parameters and features so, in theory, this work can be transferred to other HAR fields.

# Chapter 7

# Conclusion

## 7.1 General Conclusions

This study had the objective of studying the HAR, specially fundamental activities within sports using only one inertial sensor (IMU) in the chest area, integrated in a smart vest configuration. This work was motivated by the challenges regarding physical activity assessments. Current machine learning techniques and sensor capturing, allow for simpler, ergonomic and cheaper alternatives than full body suits with sensor and camera setups. This allows to detect fundamental movements essential for activity classification and performance analysis with minimal intrusion and improved user comfort. This work is also a crucial step for the integration with the TexP@ct platform.

During this thesis, a complete methodology was developed capable of selecting and preparing two publicly available datasets (PAMAP2 and RWHAR), with different characteristics and activities but compatible with the proposed work: have one sensor in the chest area capable of detecting accelerometer and gyroscope data. These datasets were segmented to have specific fundamental activities within a sport. Then these datasets were pre-processed with filtering techniques for noise reduction and gravity component removal, then a sliding window was used to extract a set of features and a base model was constructed using the PyCaret library. This base model was used for comparison with the feature reduction techniques and hyperparameter tuning final model.

The results demonstrate that even with a one sensor configuration in the upper torso area and using traditional ML models it is possible to extract sufficient information to have a high performance classification algorithm. The best performing ML model was the lightGBM achieving a F1-score of 94.3% for the PAMAP2 dataset and 90.1% for the RWHAR dataset. The feature reduction techniques and hyperparamter also allowed to increase the models performance metrics but also reduce the computational cost by having less features to select. Additionally, the lightGBM final model had the same hyperparameters for both datasets which contributes to a robust approach and better generalization for other HAR datasets. However, the confusion matrices revealed that some movements are easier to classify than others, for example, walking, jumping and lying while other were were easier than standing and running. It was concluded that this happened in the RWHAR dataset due to periods of resting between running segments lowering the accuracy for these exercises.

## 7.2 Limitations and Future Work

Even though, it was considered that these results were successful, this work does not come without its limitations. As discussed in the previous chapter both datasets are recorded in constrained experimental scenarios and do not take into consideration different environments. These controlled environments do not reflect the unpredictability of sports based activities. The number of individuals, age gap, sex and fundamental activities performed in an independent matter also affect the study. This creates a false optimal dataset, because having the activities done one by one does not represent the real scenario where an athlete/student will transition from activity to activity. These datasets also do not encompass other movements that usually are present in sports, for example changes in direction and ball related movements like passes, shooting, dribbling, etc.

In order to counteract these limitations, the creation of a new dataset using the TexP@ct vest can be introduced. This way, it would be possible to directly extract activities from students attending physical education classes. With the creation of a custom dataset, additional parameters can be controlled like demographic data, ensuring an age group typical of students, sex can also be controlled allocating half the data represented by male students and the other half female students. Additionally, the dataset should have each activity represented with the same amount of weight to guarantee a well balanced dataset. This dataset would validate the proposed work by testing the classification of these fundamental activities in the direct environment this study intends to evaluate. Also, even though this algorithm was primarily designed for recognizing fundamental activities in sports, it holds potential for broader applications, such as fall detection in elderly populations. Datasets like [44] already explore this subject and could be beneficial for future work.

Additionally, the viability for the integration of the heart rate parameter was tested and discarded for the classification model. However the ECG sensor can have other application in the context of this project, for example, evaluating effort while performing each activity, base heart rate, and maximum heart rate to assess fitness. Furthermore, deep learning models could also be tested for more complex combined tasks. This could either work has a hybrid model using both machine learning models and deep learning or as single model. With this implementation some alterations could also be developed, for instance, the introduction of a parameter of transition activity for windows that are harder to classify. With this, a weight system to take into account the previous and the next window can be used. This can be beneficial since subsequent windows can have relations with each other due to the overlap.

Overall, it can be considered that this project fulfilled its academic purpose, achieving results with high accuracy for classification of fundamental movements within sports. The conducted preprocessing extracted valuable information using the IMU configuration in the upper torso region.

Additionally, the machine learning model proved its generability for other HAR datasets with lower computational costs than those of higher complexity models.

# References


[1] Roshan Joy Martis, U Rajendra Acharya, and Hojjat Adeli. “Current methods in electrocardiogram characterization”. In: *Computers in biology and medicine* 48 (2014), pp. 133–149.

[2] J Willis Hurst. “Naming of the waves in the ECG, with a brief account of their genesis”. In: *Circulation* 98.18 (1998), pp. 1937–1942.

[3] Kresimir Friganovic et al. “Optimizing the detection of characteristic waves in ECG based on processing methods combinations”. In: *IEEE access* 6 (2018), pp. 50609–50626.

[4] Ali Isin and Selen Ozdalili. “Cardiac arrhythmia detection using deep learning”. In: *Procedia computer science* 120 (2017), pp. 268–275.

[5] Liping Xie et al. “Computational diagnostic techniques for electrocardiogram signal analysis”. In: *Sensors* 20.21 (2020), p. 6318.

[6] Jiapu Pan and Willis J Tompkins. “A real-time QRS detection algorithm”. In: *IEEE transactions on biomedical engineering* 3 (1985), pp. 230–236.

[7] Xueyu Wu et al. “Optimized pan-tompkins based heartbeat detection algorithms”. In: *2020 Chinese Control And Decision Conference (CCDC)*. IEEE. 2020, pp. 892–897.

[8] Shubha Kadambe, Robin Murray, and Gloria Faye Boudreaux-Bartels. “Wavelet transform-based QRS complex detector”. In: *IEEE Transactions on biomedical Engineering* 46.7 (1999), pp. 838–848.

[9] Mohamed Amine Belkadi and Abdelhamid Daamouche. “A robust QRS detection approach using stationary wavelet transform”. In: *Multimedia Tools and Applications* 80.15 (2021), pp. 22843–22864.

[10] Wei Zhao et al. “A simple and effective deep neural network based QRS complex detection method on ECG signal”. In: *Frontiers in Physiology* 15 (2024), p. 1384356.

[11] Norhafizan Ahmad et al. “Reviews on various inertial measurement unit (IMU) sensor applications”. In: *International Journal of Signal Processing Systems* 1.2 (2013), pp. 256–262.

[12] Jacob S Arlotti et al. “Benefits of IMU-based wearables in sports medicine: Narrative review”. In: *International Journal of Kinesiology and Sports Science* 10.1 (2022), pp. 36–43.

[13] Chenyu Gu et al. “IMU-based motion capture system for rehabilitation applications: A systematic review”. In: *Biomimetic Intelligence and Robotics* 3.2 (2023), p. 100097.

[14] Trupti Gujarathi and Kalyani Bhole. “Gait analysis using imu sensor”. In: *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*. IEEE. 2019, pp. 1–5.

[15] Aleksandar Subic. *Materials in sports equipment*. Woodhead Publishing, 2019.

[16] I Arun Faisal, T Waluyo Purboyo, and A Siswo Raharjo Ansori. “A review of accelerometer sensor and gyroscope sensor in IMU sensors on motion capture”. In: *J. Eng. Appl. Sci* 15.3 (2019), pp. 826–829.

[17] Vittorio MN Passaro et al. “Gyroscope technology and applications: A review in the industrial perspective”. In: *Sensors* 17.10 (2017), p. 2284.

[18] Luisa Mich. “Artificial intelligence and machine learning”. In: *Handbook of e-Tourism* (2020), pp. 1–21.

[19] *What is the history of artificial intelligence (AI)? | Tableau*. [Online; accessed 2025-02-25]. URL: https://www.tableau.com/data-insights/ai/history#history.

[20] Michael Haenlein and Andreas Kaplan. “A brief history of artificial intelligence: On the past, present, and future of artificial intelligence”. In: *California management review* 61.4 (2019), pp. 5–14.

[21] Konstantinos Theofilatos et al. “Predicting protein complexes from weighted protein–protein interaction graphs with a novel unsupervised methodology: evolutionary enhanced Markov clustering”. In: *Artificial intelligence in medicine* 63.3 (2015), pp. 181–189.

[22] Jeffrey A Larson, Michael H Johnson, and Sam B Bhayani. “Application of surgical safety standards to robotic surgery: five principles of ethics for nonmaleficence”. In: *Journal of the American College of Surgeons* 218.2 (2014), pp. 290–293.

[23] Pavel Hamet and Johanne Tremblay. “Artificial intelligence in medicine”. In: *metabolism* 69 (2017), S36–S40.

[24] Hatem Ibn-Khedher et al. “Next-generation edge computing assisted autonomous driving based artificial intelligence algorithms”. In: *IEEE Access* 10 (2022), pp. 53987–54001.

[25] Mohsen Soori, Behrooz Arezoo, and Roza Dastres. “Artificial intelligence, machine learning and deep learning in advanced robotics, a review”. In: *Cognitive Robotics* 3 (2023), pp. 54–70.

[26] Javier Andreu Perez et al. “Artificial intelligence and robotics”. In: *arXiv preprint arXiv:1803.10813* 147 (2018), pp. 2–44.

[27] Alexandara Harry. “Role of AI in Education.” In: *Interdiciplinary Journal & Hummanity (INJURITY)* 2.3 (2023).

[28] Neha Gupta et al. “Human activity recognition in artificial intelligence framework: a narrative review”. In: *Artificial intelligence review* 55.6 (2022), pp. 4755–4808.

[29] Alan McCabe and Jarrod Trevathan. “Artificial intelligence in sports prediction”. In: *Fifth International Conference on Information Technology: New Generations (itng 2008)*. IEEE. 2008, pp. 1194–1197.

[30] Christian Janiesch, Patrick Zschech, and Kai Heinrich. “Machine learning and deep learning”. In: *Electronic markets* 31.3 (2021), pp. 685–695.

[31] Muhammad Yaseen Khan et al. “Automated Prediction of Good Dictionary EXamples (GDEX): A Comprehensive Experiment with Distant Supervision, Machine Learning, and Word Embedding-Based Deep Learning Techniques”. In: *Complexity* 2021.1 (2021), p. 2553199.

[32] Steven J Rigatti. “Random forest”. In: *Journal of Insurance Medicine* 47.1 (2017), pp. 31–39.

[33] Pierre Geurts, Damien Ernst, and Louis Wehenkel. "Extremely randomized trees". In: *Machine learning* 63 (2006), pp. 3–42.

[34] Fangfang Lee Bryan Clark. *What is Gradient Boosting? | IBM*. [Online; accessed 2025-06-10]. URL: https://www.ibm.com/think/topics/gradient-boosting.

[35] Samir Touzani, Jessica Granderson, and Samuel Fernandes. "Gradient boosting machine for modeling the energy consumption of commercial buildings". In: *Energy and Buildings* 158 (2018), pp. 1533–1543.

[36] Guolin Ke et al. "Lightgbm: A highly efficient gradient boosting decision tree". In: *Advances in neural information processing systems* 30 (2017).

[37] Gongde Guo et al. "KNN model-based approach in classification". In: *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE: OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2003, Catania, Sicily, Italy, November 3-7, 2003. Proceedings*. Springer. 2003, pp. 986–996.

[38] Pranay Rishith. *Understanding K-Nearest Neighbors (KNN): Classification and Regression | Towards AI*. [Online; accessed 2025-06-10]. May 2023. URL: https://pub.towardsai.net/understanding-k-nearest-neighbors-a-simple-approach-to-classification-and-regression-e4b30b37f151.

[39] *PRISMA statement*. [Online; accessed 2025-06-17]. URL: https://www.prisma-statement.org/.

[40] Angel Ruiz-Zafra et al. "A comparative study on the suitability of smartphones and imu for mobile, unsupervised energy expenditure calculi". In: *Sensors* 15.8 (2015), pp. 18270–18286.

[41] Attila Reiss and Didier Stricker. "Creating and benchmarking a new dataset for physical activity monitoring". In: *Proceedings of the 5th international conference on pervasive technologies related to assistive environments*. 2012, pp. 1–8.

[42] Daniel Garcia-Gonzalez et al. "A public domain dataset for real-life human activity recognition using smartphone sensors". In: *Sensors* 20.8 (2020), p. 2200.

[43] Jennifer R Kwapisz, Gary M Weiss, and Samuel A Moore. "Activity recognition using cell phone accelerometers". In: *ACM SigKDD Explorations Newsletter* 12.2 (2011), pp. 74–82.

[44] Charikleia Chatzaki et al. "Human daily activity and fall recognition using a smartphone's acceleration sensor". In: *Information and Communication Technologies for Ageing Well and e-Health: Second International Conference, ICT4AWE 2016, Rome, Italy, April 21-22, 2016, Revised Selected Papers 2*. Springer. 2017, pp. 100–118.

[45] Oresti Banos et al. "Design, implementation and validation of a novel open framework for agile development of mobile health applications". In: *Biomedical engineering online* 14 (2015), pp. 1–20.

[46] S Chan Chang et al. "Capture-24: Activity tracker dataset for human activity recognition". In: (2021).

[47] Mohammad Malekzadeh et al. "Mobile sensor data anonymization". In: *Proceedings of the international conference on internet of things design and implementation*. 2019, pp. 49–58.

[48] Mi Zhang and Alexander A Sawchuk. "USC-HAD: A daily activity dataset for ubiquitous activity recognition using wearable sensors". In: *Proceedings of the 2012 ACM conference on ubiquitous computing*. 2012, pp. 1036–1043.

[49] Zhiwen Xiao et al. "A federated learning system with enhanced feature extraction for human activity recognition". In: *Knowledge-Based Systems* 229 (2021), p. 107338.

[50] Timo Sztyler and Heiner Stuckenschmidt. "On-body localization of wearable devices: An investigation of position-aware activity recognition". In: *2016 IEEE international conference on pervasive computing and communications (PerCom)*. IEEE. 2016, pp. 1–9.

[51] Jorge-L Reyes-Ortiz et al. "Transition-aware human activity recognition using smartphones". In: *Neurocomputing* 171 (2016), pp. 754–767.

[52] Lei Xiao et al. "A hybrid deep approach to recognizing student activity and monitoring health physique based on accelerometer data from smartphones". In: *Scientific Reports* 14.1 (2024), p. 14006.

[53] Yasin Kaya and Elif Kevser Topuz. "Human activity recognition from multiple sensors data using deep CNNs". In: *Multimedia Tools and Applications* 83.4 (2024), pp. 10815–10838.

[54] Md Zia Uddin et al. "A body sensor data fusion and deep recurrent neural network-based behavior recognition approach for robust healthcare". In: *Information Fusion* 55 (2020), pp. 105–115.

[55] Nhung Tran Thi Hong et al. "A low-cost real-time IoT human activity recognition system based on wearable sensor and the supervised learning algorithms". In: *Measurement* 218 (2023), p. 113231.

[56] Inyong Koo et al. "Contrastive Accelerometer–Gyroscope Embedding Model for Human Activity Recognition". In: *IEEE Sensors Journal* 23.1 (2023), pp. 506–513. DOI: 10.1109/JSEN.2022.3222825.

[57] Sai Siddarth Vakacherla et al. "Single accelerometer to recognize human activities using neural networks". In: *Journal of Biomechanical Engineering* 145.6 (2023), p. 061005.

[58] Laith Al-Frady and Ali Al-Taei. "Wrapper filter approach for accelerometer-based human activity recognition". In: *Pattern Recognition and Image Analysis* 30.4 (2020), pp. 757–764.

[59] Riktim Mondal et al. "A New Framework for Smartphone Sensor-Based Human Activity Recognition Using Graph Neural Network". In: *IEEE Sensors Journal* 21.10 (2021), pp. 11461–11468. DOI: 10.1109/JSEN.2020.3015726.

[60] Alexander Hoelzemann et al. "Hang-time HAR: a benchmark dataset for basketball activity recognition using wrist-worn inertial sensors". In: *Sensors* 23.13 (2023), p. 5879.

[61] Jia-Gang Qiu et al. "Research on motion recognition based on multi-dimensional sensing data and deep learning algorithms". In: *Mathematical Biosciences and Engineering* 20.8 (2023), pp. 14578–14595.

[62] Kieran Woodward et al. "A multi-sensor deep learning approach for complex daily living activity recognition". In: *Proceedings of the 2022 Workshop on Emerging Devices for Digital Biomarkers*. 2022, pp. 13–17.

[63] Mara Scattolini et al. "Leveraging Inertial Information from a Single IMU for Human Daily Activity Recognition". In: *2024 IEEE International Symposium on Medical Measurements and Applications (MeMeA)*. 2024, pp. 1–6. DOI: 10.1109/MeMeA60663.2024.10596886.

[64] Jozsef Suto et al. "Comparison of offline and real-time human activity recognition results using machine learning techniques". In: *Neural computing and applications* 32.20 (2020), pp. 15673–15686.

[65] Esma Uzunhisarcıklı, Erhan Kavuncuoğlu, and Ahmet Turan ̈Ozdemir. "Investigating classification performance of hybrid deep learning and machine learning architectures on activity recognition". In: *Computational Intelligence* 38.4 (2022), pp. 1402–1449.

[66] Neelakandan Mani, Prathap Haridoss, and Boby George. "Evaluation of a Combined Conductive Fabric-Based Suspender System and Machine Learning Approach for Human Activity Recognition". In: *IEEE Open Journal of Instrumentation and Measurement* 2 (2023), pp. 1–10. DOI: 10.1109/OJIM.2023.3301861.

[67] Lingfei Mo, Yaojie Zhu, and Lujie Zeng. "A multi-label based physical activity recognition via cascade Classifier". In: *Sensors* 23.5 (2023), p. 2593.

[68] Asif Iqbal et al. "Wearable Internet-of-Things platform for human activity recognition and health care". In: *International Journal of Distributed Sensor Networks* 16.6 (2020), p. 1550147720911561.

[69] Iwona Pajak et al. "Sports activity recognition with UWB and inertial sensors using deep learning approach". In: *2022 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. 2022, pp. 1–8. DOI: 10.1109/FUZZ-IEEE55066.2022.9882654.

[70] Muhammad Usman Sarwar et al. "Parciv: Recognizing physical activities having complex interclass variations using semantic data of smartphone". In: *Software: Practice and Experience* 51.3 (2021), pp. 532–549.

[71] Matthew N Ahmadi, Karin A Pfeiffer, and Stewart G Trost. "Physical activity classification in youth using raw accelerometer data from the hip". In: *Measurement in Physical Education and Exercise Science* 24.2 (2020), pp. 129–136.

[72] Adria Mallol-Ragolta et al. "Outer product-based fusion of smartwatch sensor data for human activity recognition". In: *Frontiers in Computer Science* 4 (2022), p. 796866.

[73] Junhui Huang, Sakdirat Kaewunruen, and Jingzhiyuan Ning. "AI-based quantification of fitness activities using smartphones". In: *Sustainability* 14.2 (2022), p. 690.

[74] Abdul Kadar Muhammad Masum et al. "Human Activity Recognition Using Smartphone Sensors: A Dense Neural Network Approach". In: *2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT)*. 2019, pp. 1–6. DOI: 10.1109/ICASERT.2019.8934657.

[75] Justyna Patalas-Maliszewska et al. "Inertial sensor-based sport activity advisory system using machine learning algorithms". In: *Sensors* 23.3 (2023), p. 1137.

[76] Ala Mhalla and Jean-Marie Favreau. "Wearable IoT Sensor Combining Deep Learning for Enhanced Human Activity Recognition in Indoor and Outdoor Settings". In: *International Symposium on Ubiquitous Networking*. Springer. 2023, pp. 43–53.

[77] Ferhat Bozkurt. "A comparative study on classifying human activities using classical machine and deep learning methods". In: *Arabian Journal for Science and Engineering* 47.2 (2022), pp. 1507–1521.

[78] Md Maruf Hossain Shuvo et al. "A Hybrid Approach for Human Activity Recognition with Support Vector Machine and 1D Convolutional Neural Network". In: *2020 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*. 2020, pp. 1–5. DOI: 10.1109/AIPR50011.2020.9425332.

[79] Sheikh Badar ud din Tahir et al. "Stochastic recognition of human physical activities via augmented feature descriptors and random forest model". In: *Sensors* 22.17 (2022), p. 6632.

[80] H Sharen et al. "WISNet: A deep neural network based human activity recognition system". In: *Expert Systems with Applications* 258 (2024), p. 124999.

[81] Agus Eko Minarno, Wahyu Andhyka Kusuma, and Hardianto Wibowo. "Performance comparisson activity recognition using logistic regression and support vector machine". In: *2020 3rd International conference on intelligent autonomous systems (ICoIAS)*. IEEE. 2020, pp. 19–24.

[82] Shilpa Ankalaki and M Thippeswamy. "The customized 1d cnn for sensor-based human activity recognition using various benchmark datasets". In: *Journal of Engineering Science and Technology* 17.4 (2022), pp. 2315–2335.

[83] Berke Ataseven et al. "Physical Activity Recognition using Deep Transfer Learning with Convolutional Neural Networks". In: *2022 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*. 2022, pp. 1–6. DOI: 10.1109/DASC/PiCom/CBDCom/Cy55231.2022.9928021.

[84] Fayez Alharbi, Lahcen Ouarbya, and Jamie A Ward. "Comparing sampling strategies for tackling imbalanced data in human activity recognition". In: *Sensors* 22.4 (2022), p. 1373.

[85] Sakorn Mekruksavanich, Ponnipa Jantawong, and Anuchit Jitpattanakul. "Ensemble Deep Learning Network for Enhancing Performances of Sensor-Based Physical Activity Recognition Based on IMU Sensor Data". In: *2024 5th International Conference on Big Data Analytics and Practices (IBDAP)*. 2024, pp. 150–155. DOI: 10.1109/IBDAP62940.2024.10689690.

[86] Muhammad Shoaib et al. "Fusion of smartphone motion sensors for physical activity recognition". In: *Sensors* 14.6 (2014), pp. 10146–10176.

[87] Mst. Alema Khatun and Mohammad Abu Yousuf. "Human Activity Recognition Using Smartphone Sensor Based on Selective Classifiers". In: *2020 2nd International Conference on Sustainable Technologies for Industry 4.0 (STI)*. 2020, pp. 1–6. DOI: 10.1109/STI50764.2020.9350486.

[88] Sakorn Mekruksavanich and Anuchit Jitpattanakul. "Rnn-based deep learning for physical activity recognition using smartwatch sensors: A case study of simple and complex activity recognition". In: *Mathematical Biosciences and Engineering* 19.6 (2022), pp. 5671–5698.

[89] Abdul Rehman Javed et al. "Analyzing the effectiveness and contribution of each axis of tri-axial accelerometer sensor for accurate activity recognition". In: *Sensors* 20.8 (2020), p. 2216.

[90] Sijie Ji, Xinzhe Zheng, and Chenshu Wu. "Hargpt: Are llms zero-shot human activity recognizers?" In: *2024 IEEE International Workshop on Foundation Models for Cyber-Physical Systems & Internet of Things (FMSys)*. IEEE. 2024, pp. 38–43.

[91] Marius Bock et al. "Investigating (re) current state-of-the-art in human activity recognition datasets". In: *Frontiers in Computer Science* 4 (2022), p. 924954.

[92] Muhammad Shoaib, Hans Scholten, and Paul JM Havinga. "Towards physical activity recognition using smartphone sensors". In: *2013 IEEE 10th international conference on ubiquitous intelligence and computing and 2013 IEEE 10th international conference on autonomic and trusted computing*. IEEE. 2013, pp. 80–87.

[93] Adel Alshamrani. "Continuous Human Activity Detection Using Multiple Smart Wearable Devices in IoT Environments". In: *International Journal of Computer Science & Network Security* 21.2 (2021), pp. 221–228.

[94] S Badar Ud Din Tahir. "Intelligent Media-Wearable Smart Home Activities (IM-WSHA)". In: *Accessed: Nov* 10 (2020), p. 2020.

[95] Anna Ferrari et al. "On the Personalization of Classification Models for Human Activity Recognition". In: *IEEE Access* 8 (2020), pp. 32066–32079. DOI: 10.1109/ACCESS.2020.2973425.

[96] Daniela Micucci, Marco Mobilio, and Paolo Napoletano. "Unimib shar: A dataset for human activity recognition using acceleration data from smartphones". In: *Applied Sciences* 7.10 (2017), p. 1101.

[97] Gustavo Aquino, Marly GF Costa, and Cicero FF Costa Filho. "Explaining one-dimensional convolutional models in human activity recognition and biometric identification tasks". In: *Sensors* 22.15 (2022), p. 5644.

[98] Ye Liu et al. "From action to activity: sensor-based activity recognition". In: *Neurocomputing* 181 (2016), pp. 108–115.

[99] Marc Bachlin et al. "Wearable assistant for Parkinson's disease patients with the freezing of gait symptom". In: *IEEE Transactions on Information Technology in Biomedicine* 14.2 (2009), pp. 436–446.

[100] Niall Twomey et al. "The SPHERE challenge: Activity recognition with multimodal sensor data". In: *arXiv preprint arXiv:1603.00797* (2016).

[101] Philipp M Scholl, Matthias Wille, and Kristof Van Laerhoven. "Wearables in the wet lab: a laboratory system for capturing and guiding experiments". In: *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 2015, pp. 589–599.

[102] Wallace Ugulino et al. "Wearable Computing: Accelerometers' Data Classification of Body Postures and Movements". In: *Advances in Artificial Intelligence - SBIA 2012*. Ed. by Leliane N. Barros et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 52–61. ISBN: 978-3-642-34459-6.

[103] Filippo Palumbo et al. "Human activity recognition using multisensor data fusion based on Reservoir Computing". In: *Journal of Ambient Intelligence and Smart Environments* 8.2 (2016). Cited by: 134; All Open Access, Green Open Access, pp. 87–107. DOI: 10.3233/AIS-160372. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84962580238&doi=10.3233%2fAIS-160372&partnerID=40&md5=e3e810bc9806d5b396e3fb810c708796.

[104] Pierluigi Casale, Oriol Pujol, and Petia Radeva. "Personalization and user verification in wearable systems using biometric walking patterns". In: *Personal and Ubiquitous Computing* 16 (2012), pp. 563–580.

[105] Lei Yin and Xiangyun Sun. "Textile-based sensors for human motion sensing: recent developments and future perspectives". In: *Nanocomposites* 11.1 (2025), pp. 79–98.

[106] *Monitor and Analyze Players' Load With Minimal Effort | KINEXON Sports*. [Online; accessed 2025-05-30]. URL: https://kinexon-sports.com/products/perform-imu/.

[107] *Try the Best Sports Vest For Player Tracking | KINEXON Sports*. [Online; accessed 2025-06-18]. URL: https://kinexon-sports.com/blog/sports-vests-vital-part-of-player-tracking/.

[108] *Smart Athlete Vest - Light Brain*. [Online; accessed 2025-05-30]. URL: https://lightbraincm.com/smart-vest/.

[109] Hexoskin. *Hexoskin Smart Shirts - Cardiac, Respiratory, Sleep & Activity Metrics*. [Online; accessed 2025-05-30]. URL: https://hexoskin.com/.

[110] *Catapult | Sports Technology | Unleash Potential*. [Online; accessed 2025-05-30]. URL: https://www.catapult.com/.

[111] *PPS#17 - Fato inteligente para treino desportivo :: TEXP@CT*. [Online; accessed 2025-06-13]. URL: https://texpact.pt/solucoes/artigos___solucoes/pps17___fato_inteligente_para_treino_desportivo-6bc71ca1.

[112] Wilmar Hernandez. "Improving the response of an accelerometer by using optimal filtering". In: *Sensors and Actuators A: Physical* 88.3 (2001), pp. 198–208.

[113] Stephen Butterworth et al. "On the theory of filter amplifiers". In: *Wireless Engineer* 7.6 (1930), pp. 536–541.

[114] Kazunori Ohkawara et al. "Real-time estimation of daily physical activity intensity by a triaxial accelerometer and a gravity-removal classification algorithm". In: *British Journal of Nutrition* 105.11 (2011), pp. 1681–1691.

[115] *scikit-learn: machine learning in Python — scikit-learn 1.7.0 documentation*. [Online; accessed 2025-06-18]. URL: https://scikit-learn.org/stable/.

[116] Moez Ali. *PyCaret: An open source, low-code machine learning library in Python*. Apr. 2020. URL: https://www.pycaret.org.

[117] Hervé Abdi and Lynne J Williams. "Principal component analysis". In: *Wiley interdisciplinary reviews: computational statistics* 2.4 (2010), pp. 433–459.

[118] Arif Priyatno and Triyanna Widiyaningtyas. "A SYSTEMATIC LITERATURE REVIEW: RECURSIVE FEATURE ELIMINATION ALGORITHMS". In: *JITK (Jurnal Ilmu Pengetahuan dan Teknologi Komputer)* 9 (Feb. 2024), pp. 196–207. DOI: 10.33480/jitk.v9i2.5015.

[119] Nursultan Zhantileuov and Serik Ospanov. "A Comparative Study of Supervised Machine Learning and Deep Learning Techniques with Feature Selection Methods for Classifying Parkinson's Disease Based on Speech Impairments". In: *2024 IEEE 4th International Conference on Smart Information Systems and Technologies (SIST)*. IEEE. 2024, pp. 124–129.

[120] Leo Breiman. "Random forests". In: *Machine learning* 45 (2001), pp. 5–32.

[121] Jerome H Friedman. "Greedy function approximation: a gradient boosting machine". In: *Annals of statistics* (2001), pp. 1189–1232.

[122] Evelyn Fix. *Discriminatory analysis: nonparametric discrimination, consistency properties*. Vol. 1. USAF school of Aviation Medicine, 1985.

[123] Rodger J Elble. “Gravitational artifact in accelerometric measurements of tremor”. In: *Clinical Neurophysiology* 116.7 (2005), pp. 1638–1643.

[124] *Welcome to LightGBM’s documentation! — LightGBM 4.6.0.99 documentation*. [Online; accessed 2025-06-14]. URL: https://lightgbm.readthedocs.io/en/latest/index.html.

# Appendix A

# Appendix

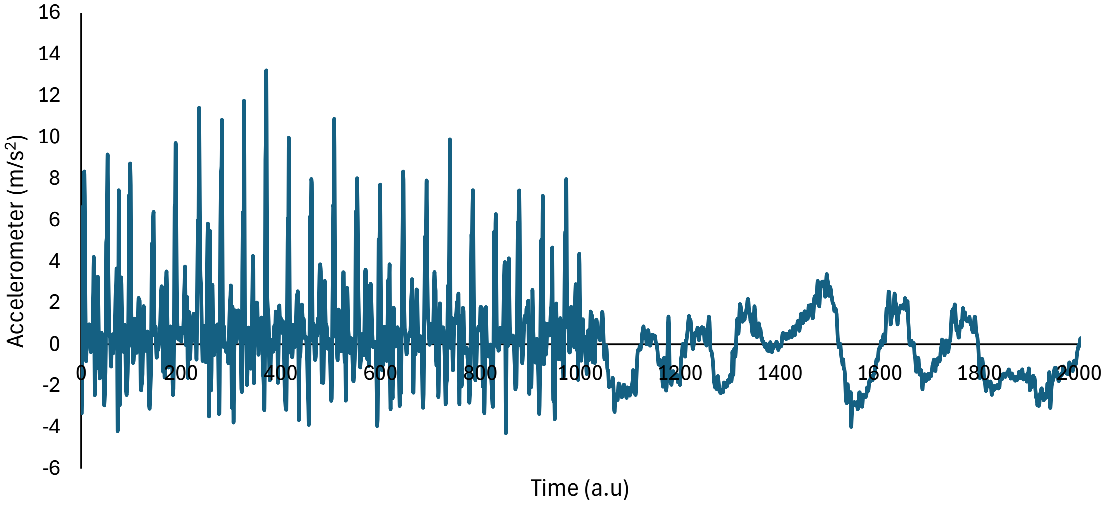



Figure A.1: Accelerometer data for participant 4 performing running activity wearing the IMU positioned on the chest