Letícia Maria Sousa Fernandes

Bachelor Degree in Biomedical Engineering Sciences

Learning Human Behaviour Patterns by Trajectory and Activity Recognition

Dissertation submitted in partial fulfillment
of the requirements for the degree of

Master of Science in
Biomedical Engineering

Adviser: Doutor Hugo Filipe Silveira Gamboa, Assistant Professor, NOVA School of Science and Technology, NOVA University of Lisbon

Examination Committee

Chairperson: Doutor Ricardo Nuno Pereira Verga e Afonso Vigário
Raporteur: Doutor Ricardo da Costa Branco Ribeiro Matias
Member: Doutor Hugo Filipe Silveira Gamboa

FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE NOVA DE LISBOA

October, 2019

Learning Human Behaviour Patterns by Trajectory and Activity Recognition

This document was created using the (pdf)LaTeX processor, based in the "novathesis" template[1], developed at the Dep. Informática of FCT-NOVA [2].
[1] https://github.com/joaomlourenco/novathesis [2] http://www.di.fct.unl.pt

Acknowledgements

The accomplishment of this important stage of my academic life would not be possible without the support of several people who were always with me during these past few years.

First of all, I would like to express my gratitude to my adviser, Professor Hugo Gamboa, for giving me the opportunity of developing my Master thesis under his guidance and for all his advice and insights. Thank you for introducing me to the wonderfulness of the world of data science and signal processing. I would like to thank *Associação Fraunhofer Portugal Research* for giving me the opportunity of developing my thesis on their facilities, for the amazing environment and for allowing me to participate in different projects that lead to my professional growth. I will be eternally grateful to the person who followed me up during this journey, Marília Barandas, thank you for being always available to clarify my doubts, for the transmitted knowledge about data science, for all the patience, guidance and for the unconditional support during the most stressful times of this thesis. Thank you! I am also grateful to Ricardo Leonardo for all the availability to clarify my doubts, for all the suggestions and for sharing deep learning knowledge. For everyone at the Lisbon office, thank you, especially to my master thesis colleagues for making this journey much more fun.

For those who lived with me at RUFS, thank you for all the great moments, adventures, dinners and parties that we experience together. To the friends that I carry always in my heart from my lovely home, thank you for being always there for me. To the person who always accompanied me during this past 5 years, that always had my back during the bad and good moments of this journey, always ready to cheer me up when I most needed, my study-mate, room-mate, cooking-mate, my deepest friendship, Ana Elisa Oliveira, my sincere thanks, without you this would, certainly, not be the same. I would also like to express my gratitude to Sérgio Gonçalves, for being always present in the most important times of my life, for all the support, care and dedication for making me a happier person.

To the people responsible for who I am today, my parents, you always did your best for giving me all the opportunities that I had during my whole life, thank you for transmitting me important values such as effort, dedication, respect and modesty. Thank you for all the love. Finally, to my two annoying sisters, thank you for making me the strong person that I am today, for the role-model, for always being there for me when I needed the most and for believing that I was capable of getting here.

Abstract

The world's population is ageing, increasing the awareness of neurological and behavioural impairments that may arise from the human ageing. These impairments can be manifested by cognitive conditions or mobility reduction. These conditions are difficult to be detected on time, relying only on the periodic medical appointments. Therefore, there is a lack of routine screening which demands the development of solutions to better assist and monitor human behaviour. The available technologies to monitor human behaviour are limited to indoors and require the installation of sensors around the user's homes presenting high maintenance and installation costs. With the widespread use of smartphones, it is possible to take advantage of their sensing information to better assist the elderly population. This study investigates the question of what we can learn about human pattern behaviour from this rich and pervasive mobile sensing data. A deployment of a data collection over a period of 6 months was designed to measure three different human routines through human trajectory analysis and activity recognition comprising indoor and outdoor environment. A framework for modelling human behaviour was developed using human motion features, extracted in an unsupervised and supervised manner. The unsupervised feature extraction is able to measure mobility properties such as step length estimation, user points of interest or even locomotion activities inferred from an user-independent trained classifier. The supervised feature extraction was design to be user-dependent as each user may have specific behaviours that are common to his/her routine. The human patterns were modelled through probability density functions and clustering approaches. Using the human learned patterns, inferences about the current human behaviour were continuously quantified by an anomaly detection algorithm, where distance measurements were used to detect significant changes in behaviour. Experimental results demonstrate the effectiveness of the proposed framework that revealed an increase potential to learn behaviour patterns and detect anomalies.

Keywords: Human Behaviour, Pattern Recognition, Anomaly Detection, Ambient Assisted Living, Probability Density Function, Clustering

Resumo

A população mundial está a envelhecer, o que aumenta a conscientização de imparidades neurológicas e comportamentais que podem surgir com o envelhecimento humano. Estas imparidades podem se manifestar através de condições cognitivas ou da redução da mobilidade, sendo dificilmente detetadas a tempo, quando dependentes de consultas médicas periódicas. Portanto, existe falta de monitorização da rotina humana, o que exige o desenvolvimento de soluções que permitam assistir e monitorizar o comportamento humano. As tecnologias disponíveis são limitadas a espaços interiores, e requerem a instalação de sensores no interior da casa dos utilizadores, o que apresenta elevados custos de manutenção e instalação. A ampla utilização dos telemóveis, tornou possível o aproveitamento dos seus sensores para assistir a população envelhecida. Este estudo investiga a questão de o que podemos aprender do comportamento humano a partir da informação rica dos sensores dos telemóveis. Ao longo de um período de 6 meses foram adquiridos dados de três rotinas humanas diferentes, através da análise da trajetória e do reconhecimento humano, compreendendo espaços interiores e exteriores. Foi desenvolvida uma ferramenta para modelar o comportamento humano que permite extrair características do movimento humano de uma forma não-supervisionada e supervisionada. A extração não supervisionada consiste na medição de propriedades da mobilidade, tal como, o comprimento do passo, os pontos de interesse, ou atividades da locomoção do utilizador, inferidas através de um classificador de atividades que não depende do utilizador. A extração supervisionada foi projetada para ser dependente do utilizador, visto que, cada utilizador tem comportamentos específicos que são comuns à sua rotina. Através dos padrões humanos aprendidos, o comportamento inferido foi quantificado de forma contínua, através de um algoritmo para a deteção de anomalias, onde uma medida de distância foi utilizada para detetar alterações significativas no comportamento humano. Resultados experimentais demonstraram a eficiência da ferramenta desenvolvida, revelando um elevado potencial para aprender padrões no comportamento humano e para detetar anomalias.

Palavras-chave: Comportamento Humano, Reconhecimento de Padrões, Deteção de Anomalias, Ambiente de Vida Assistida, Função Densidade de Probabilidade, Agrupamento

Contents

List of Figures

List of Tables

Acronyms

ADA AdaBoost.

ADLs Activities of Daily Living.

AP Access Point.

AUC Area Under the Curve.

BIC Bayesian Inference Criterion.

BSSID Basic Service Set Identifier.

CHMM Continuous hidden markov model.

DBSCAN Density-based spatial clustering of applications with noise.

DR Dead Reckoning.

DT Decision Tree.

DTW Dynamic Time Warping.

ED Euclidean Distance.

EMD Earth Mover's Distance.

GPS Global Positioning System.

HAR Human Activity Recognition.

HDBSCAN Hierarchical Density-based spatial clustering of applications with noise.

HMM Hidden Markov Models.

KDE Kernel Density Estimate.

KNN K Nearest Neighbor.

LCS Longest Common Sequence.

LDA Latent Dirichlet Allocation.

MFCC Mel Frequency Cepstral Coefficients.

ML Maximised Likelihood.

NARX Non-linear Autoregressive netwoRk with eXogenous.

NB Naive Bayes.

POI Points of Interest.

RF Random Forest.

RNN Recurrent Neural Networks.

ROC Receiver Operating Characteristics.

RSSI Received Signal Strength Indicator.

S-DTW Subsequence-DTW.

SVM Support Vector Machine.

TOI Times of Interest.

TSFEL Time Series Feature Extraction Library.

Chapter 1

Introduction

1.1 Motivation

The increase in life expectancy leads to the ageing population growing worldwide. According to United Nations [1], the global population over 60 years in 2017 was more than twice as large as in 1980, and the number of elderly people is expected to double again in 2050, as it can be seen in Figure 1.1. As a consequence, common health conditions associated with ageing, that affects human behaviour, such as physical declining, psychological and cognitive alterations are increasing. This is the case of people suffering from dementia that according to the World Alzheimer report of 2018 [2], 50 million people worldwide were living with dementia in 2018 and the estimation is to increase to 152 million in 2050.

Physical declining of elderly people is observed through the decrease of walking speed, mobility disability that is associated with falls, social isolation, difficulty in performing

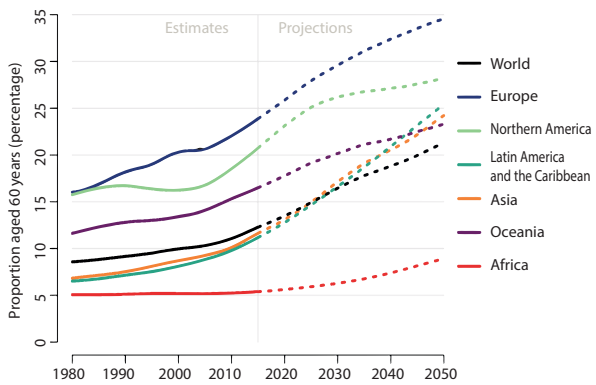


Figure 1.1: Percentage of population aged over 60 years by region, from 1980 to 2050. Adapted from [1].

activities of daily living such as cooking, using the toilet and dressing. On the other hand, cognitive alterations, that includes cognitive ageing, dementia and depression are more difficult to be detected in early stages of the disease [3]. For example, early symptoms of dementia that include memory impairment, may not be detected by doctors in periodical visits, given that there is a lack of routine screening. So the role of the caregiver is very important for the early diagnosis of these health conditions [4]. However, a significant portion of these people live home alone [5], and it may be difficult to either detect and monitor the disease leading to its progression. Thus, a reliable tool for learning more about the person's daily living for helping the diagnose and follow-up these impairments is needed. The assessment of human behaviour is the basis for understanding people's needs and problems, subsequently, helping them improve their lives. Human behaviour tends to promote well-defined motions which are repeated every day, suchlike trajectory patterns and sequences of performed activities. With the widespread of technology, specifically smartphones, it is possible to recognise human motions and to monitor human daily routines, since they possess multiple accurate sensors to better assist humans. Moreover, smartphones allow the monitoring of routines in a cheap and unobtrusive way.

1.2 Applications

The understanding of human behaviour patterns has a considerable impact in healthcare [6], however, it has other meaningful applications such as marketing analysis, security and even tourist management.

- **Health-Care**

  In healthcare, the analysis of human behaviour is important for understanding changes in behaviour that are associated with different health conditions such as dementia, depression, memory loss or even unhealthy daily habits. The quantification of these changes can assist doctors to diagnose diseases or even evaluate its progression.

- **Marketing**

  In the marketing point of view, companies intent to influence consumers to buy their products, to do so, they need to understand people's needs, what are the kind of products do they buy, at which specific times and places [7]. For the improvement of marketing strategies, it is important the understanding of consumers behaviour to describe the several types of consumers, predict their behaviour and thus to develop marketing strategies that are directly constructed to affect each particular behaviour.

- **Security**

  The analysis of human behaviour for security purposes has a significant role to guarantee the well being of society. This analysis allows to find patterns of normal

human behaviours and from these patterns detect suspicious behaviours potentially indicative of terrorism, robberies or even cyberattacks [8, 9].

- **Tourist Management**

  Understanding tourist behaviour by mobility patterns is a key goal in tourist management, as understanding the favourites places visited by tourists may be important for marketing purposes, urban planning and transportation [10].

1.3 Literature Review

The focus of this study is to learn human motion patterns on an everyday basis by human trajectory analysis and activity recognition, that are detected by several sensors embedded in smartphones. In the following, the related work regarding activity recognition, location technologies, human patterns and anomaly detection is reviewed.

1.3.1 Human Activity Recognition

In the context of activity recognition using smartphones, the available studies in the literature are mainly divided into simple and complex activities. Simple activities are focused on recognising locomotion activities such as *walking*, *standing*, *sitting*, *walking up* and *walking down*. The complex ones are more concerned in detecting activities such as *brewing coffee*, *cooking*, *taking shower*, *teeth brushing*, among others. For human behaviour analysis, both simple and complex activities will be addressed in this study.

Regarding simple activities, Ronao et al. [11], by stating that activities are hierarchical, proposed a two-stage Continuous hidden markov model (CHMM) using accelerometer and gyroscope data from a smartphone to recognise six human activities (*walking*, *walking up*, *walking down*, *sitting*, *standing*, *lying*). Although several classification methods were tested during this study, CHMM showed competitive performance and low errors, resulting in an accuracy of 91.76%. Chernbumroong et al. [12] proposed a study for the recognition of five human activities (*standing*, *lying*, *walking* and *running*) using a single wrist-worn accelerometer. By selecting a total of 13 features from temporal and frequency domain, the highest accuracy obtained was 94% using a Decision Tree (DT) classifier. Santos [13] focus on detecting 10 complex activities, suchlike *opening a door*, *brushing teeth* and *typing the keyboard*. The developed framework uses signals from accelerometer, gyroscope, magnetometer and microphone sensors, and the classifier was based on multiple hidden Markov models, one per activity. The developed solution was evaluated in the offline context, where it achieved an accuracy of 84±4.8%. Shoaib et al. [14] evaluated the performance of combining wrist and pocket positioning motion sensors for recognising both simple and complex activities with seven different windows sizes by using K Nearest Neighbor (KNN), DT and Naive Bayes (NB) classifiers. They concluded that for complex activities the recognition performance is increased with increasing window size and the three classifiers presented similar behaviours.

1.3.2 Location Technologies

For the purpose of detecting trajectories at the outdoor level the most common method is Global Positioning System (GPS). However, it is not a feasible solution for indoor location because inside buildings the GPS signal is attenuated by the building structure. In the last two decades, the effort from the scientific community to develop a robust and precise indoor location system resulted in a large number of solutions. Indoor location systems can be divided into infrastructure-free and infrastructure-based. The most common sources of information about infrastructure-based solutions are radio frequency signals, such as Bluetooth, ultrasounds or light [15]. Normally, the used infrastructure transmits a specific signal to the device being located. On the other hand, infrastructure-free solutions use opportunistic readings from signals that are pervasively available in the majority of environments, like a magnetic field, atmospheric pressure, ambient light, Wi-Fi or sound [16, 17, 18].

An example of an infrastructure-free solution is the study conducted by Guimarães et al. [16] that relies on a human motion tracking algorithms combined with an opportunistic sensing mechanism. The human motion tracking algorithm is based on multiple gait-model based filtering techniques that include sensor fusion, movement detection, direction of movement and floor changes detection. The evaluation process resulted in overall median localisation errors between 1.11m and 1.68m. Leornardo et al. [18] developed a framework for determining the user's location through the sound recorded by the user's device. The proposed algorithm for room-level location purposes used a Support Vector Machine (SVM) classifier and obtained an accuracy of 90%.

1.3.3 Motion Patterns

A pattern is defined as regularity in data, for instance, when data elements are repeated predictably. The search for patterns is a common subject in a variety of engineering and scientific areas [19]. The pattern discovery that is dedicated to learning motion patterns is the focus of this study since motion patterns can be used to infer about human behaviour. Common human behaviour can be found either in trajectory or activity recognition. In the following, some studies that find human motion patterns through trajectory or activity recognition are described.

Zheng et al. [20] proposed a heuristic method combining Dynamic Time Warping (DTW) and Earth Mover's Distance (EMD) to understand tourist mobility through the measurement of trajectory similarity. Trajectory similarity was calculated based on the weighted sum of spatial and temporal similarity, and types of tourist trajectory were obtained through KNN algorithm. The resulting method proved to be accurate and noise resistant. Shou and Di [21] created a methodological framework to analyse the similarity of activity patterns using multi-person multi-day GPS trajectories, which helps to analyze people's frequent travel patterns. Pattern similarity was measured using the Longest Common Sequence (LCS) and hierarchical clustering was applied to obtain clusters based

on a dissimilarity measure, without the need of a predefined cluster number. This work presents some limitations because temporal-spatial information is not considered. Mahmoud et al. [22] used Recurrent Neural Networks (RNN) to identify behavioural patterns, based on activity recognition, in an intelligent environment constituted by binary sensors from which data is converted to temporal sequences of activities. By comparing two models of RNN , Non-linear Autoregressive netwoRk with eXogenous (NARX) inputs and Elman network, it was concluded, by the authors, that NARX performed better than Elman network to predict and extract behavioural patterns. The study conducted by Farrahi and Gatica-Perez [23] discovers human routines, characterising both individual and group behaviours, using location data collected by mobile phones in an unsupervised manner. This study uses topic models that are probabilistic generative models for documents. An individual's day is represented by a bag of location sequences (or words) that is the histogram of the location sequence present in the day, these bags consider location transitions and fine-grain, coarse-grain time considerations. These bags are used as an input for topic models including Latent Dirichlet Allocation (LDA) that is used to find routine patterns characteristic of all days in the dataset, and author topic model that discovers routines of individual users considering the user identity.

1.3.4 Anomaly Detection

Anomalies are characterised as a non-conformity of the expected behaviour that follows a specific pattern. Anomalies detection is an important subject matter, and it is widely used for fraud detection, health-care, intrusion detection or even military surveillance [24]. Hence, besides learning motion patterns, some authors focus on detecting anomalies.

Suzuki et al. [25] proposed an unsupervised learning method to learn motion patterns and detect anomalies by the analysis of human trajectory recorded in a real store by cameras. Hidden Markov Models (HMM) were used to model spatial and temporal features of trajectory to detect patterns and anomalies. Through probabilistic distances between HMM the method detects anomalies and classifies several behaviour patterns. Trajectories were projected onto a low-dimensional space applying Multi-Dimensional Scaling and grouped using the k-means clustering method. The result of clustering showed trajectory patterns, and those trajectories whose maximum likelihood of clusters was smaller than a certain threshold was considered an anomaly. Forkan et al. [26] developed a context-aware change detection model using machine learning and statistical models. This study models behaviour through daily activity and vital signals. They created a HMM to detect anomalies in sequences of daily activities in an ambient assisted living. To detect behavioural changes in a user's lifestyle related to the time duration of activities or activities frequency a statistical model measuring Gaussian distribution of activities was used. To detect increasing or decreasing trend of vital signals a Holt's linear trend method was implemented. By combining anomalies from all domains they obtained a final decision either is a true anomaly or not using a fuzzy rule-based model categorised

in four stages including normal, warning, alert and emergency. A study conducted by Tomforde et al. [27] developed models to learn the user's behaviour in a health enabling living environments equipped with multiple sensors, by detecting the user's location and activities sequences. Two multinomial HMM were trained with the normal course of the user's days, the first relying on the user's location sequence and the second the user's activities sequence using the locations as hidden states. The resulting log-likelihood of the HMM were modelled into a Kernel Density Estimate (KDE) to signalise a deviation from the expected normal behaviour pattern.

An interesting technology that has been developed in the field of human behaviour modelling is Hive Link [1]. Hive Link is a recent innovating solution developed by Centrica Hive Limited that provides a home service that allows caregivers to check in and detect deviations on a person's routine. This smart service involves the prior installation of smart-plugs and sensors around the home, that use pattern detection to get to know the person's routine and detect any changes. The system takes seven days to learn a person's routine and has the capability of getting smarter with experience. The system also comprises a Hive app where caregivers can check the activity log and receive notifications if the person is not following their usual routine. Hive Link is an inspiring solution for this study because of its focus on finding out patterns and deviations from a person's routine. However, the solution that is going to be developed in this study presents some advantages compared to Hive Link. Firstly, the use of data from smartphone/wearable sensors to model behaviour patterns, decreases the maintenance costs, since, there is no need for installation of sensors around home. Secondly, the Hive Link solution is limited to indoors and the covered area by the installed sensors. This study extends the range of application to outdoors and uses a large number of metrics (both indoors and outdoors) that can be applied in different contexts such as health-care, security or management.

1.3.5 Summary

The widespread use of mobile devices is producing a huge amount of data, making the discovery of human actions, activities and interactions possible. The challenge now is no longer that of obtaining data, but that of using these vast amounts of data from different sources and recognising patterns that could give us a better understanding of human behaviour. Although there are some studies in the literature about modelling human behaviour, the people's behaviour changes are often hard to quantify. Moreover, the aggregation of both activity recognition and trajectory analysis remains relatively unexplained. Furthermore, most of the studies that found patterns by activity recognition rely on the installation of sensors around the home, which presents higher installation and maintenance cost compared to the use of smartphone sensors that are going to be used in this study. Finally, with this study, it will be evaluated the possibility to detect and quantify anomalies in humans routines by continuously learning their daily patterns.

[1] Available in https://www.hivehome.com/services/connected-care-hive-link (visited on 27/02/2019)

1.4 Objectives

The main goal of this study is to develop a solution for learning human behaviour patterns by the analysis of trajectory and activity recognition. For reaching this goal, models for learning human behaviour patterns, to classify and quantify changes in human behaviour considering the learned behavioural patterns are proposed. We also intend to explore the extraction of relevant features suitable for learning a wide variety of human behaviour patterns through an unsupervised manner, i.e. features that can be extracted without the need for any annotation effort, and a supervised manner, once more detailed information about the human daily living is required.

The main goals are the following: (1) conduct a deep exploration of how smartphone' sensors can be used to extract information about human behaviour; (2) design and acquire a complete and extensive dataset of human routines comprising different challenges, such as indoor or outdoor environment; (3) create an algorithm to extract human motion features through unsupervised and supervised methods; (4) develop a framework to learn and model human behaviour using the extracted features; (5) propose an unsupervised anomaly detection algorithm for an earlier detection of human behaviour changes.

By modelling human behaviour it will be possible to help in the early diagnosis of pathologies that cause changes in behaviour, providing timely treatment and improving quality of life.

1.5 Document Structure

The document structure is divided into five chapters as illustrated in Figure 1.2. The first chapter is the introduction starting with this study motivation, real-context applications, the literature review, and the study objectives. The second chapter addresses the theoretical background needed for the development of this study. The third chapter contains a detailed description of the implemented framework for learning human behaviour patterns. The fourth chapter regards the main results achieved by evaluating real case scenarios using the developed framework. The fifth and last chapter comprises these study conclusions and future work.

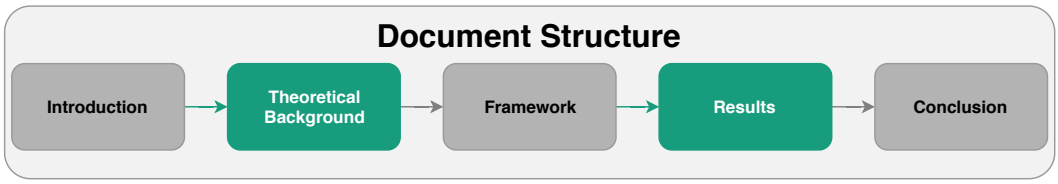


Figure 1.2: Document structure overview. The document comprises five chapters namely the introduction, theoretical background, framework, results and conclusion.

Chapter 2

Theoretical Background

This chapter contains the essential theoretical background to develop this project, it is composed by four sections. The first section includes the description of human behaviour, and motion patterns and anomalies detection concepts. The second section presents some notions of activity recognition and location technologies, followed by the description of smartphone sensors that are going to be used in this study. The third embraces machine learning concepts that are going to be used for learning human behaviour patterns. The fourth and last section comprises similarity measures.

2.1 Human Behaviour

Understanding human behaviour is not a trivial task, as it is stated by Skinner [28] "Behaviour is a difficult subject matter, not because it is inaccessible, but because it is extremely complex. Since it is a process, rather than a thing, it cannot easily be held still for observation.". Human behaviour is defined as the response of an individual to internal or external stimuli, involving cognition, emotion and executive functions.

Cognitive Functions: Cognition refers to functional proprieties of an individual that may be inferred from behaviour, namely receptive functions that corresponds to the individual capability to acquire, classify and integrate information, memory and learning functions referring to information storage and retrieval, thinking that involves mental organisation and reorganisation of information, and expressive functions that are defined as how the information is communicated or acted upon. Cognitive impairments are related to the degree of consciousness of the person, the attention level and the activity rate that affects the speed of mental processes and consequently the speed of motor responses [29].

Emotional Functions: Emotional or personality functions refers to human feelings

and motivations. Impairments in these functions are usually associated with brain injuries and the most severe impairments may be related to brain diseases. Emotional and personality changes are dependent on the severity of the brain impairments. Direct effects of brain injuries include disinhibition, diminution of anxiety and mild euphoria. Profound emotional and personality changes involve chronic frustration and radical changes in lifestyle, these changes are usually expressed by human depression [29].

Executive Functions: Executive functions correspond to the capabilities that enable humans to have independent, purposive and self-serving behaviour. Obvious defects of executive functions are the ones that are observable by naive persons, for example, emotional lability, irritability or excitability. The most serious defects are the ones that may be missed or not recognised by experts due to the lack of routine monitoring of these patients. These defects include the impaired capacity to initiate an activity, decreased or absent motivations and defects in planning and carrying out the sequence of activities that express high-level behaviours [29].

To assess human behaviour it is important to understand that human behaviour also relies on subject variability, where humans in the same situation can take different actions. Human behaviour variability may be influenced by cultural, ecological, socio-political and education contexts, and by biological and genetic transmissions [30]. Variability types are divided into variability within-subject and across-subject. Within-subject variability corresponds to a situation where an individual takes different actions at different times in the same situation. Across-subject variability refers to different actions taken by two different people in the same circumstance [31].

Human behaviour variability increases the complexity of this study because humans do not always behave in the same way. Therefore, one human may have multiple behaviour patterns and the behaviour patterns found for one subject may not have the same meaning for defining if the behaviour is or not normal comparing to other subjects. It only has meaning comparing to the subject behaviour pattern. Thus, the model for finding human behaviour patterns should be learned for each subject, because each subject will have its specific behaviour patterns.

The assessment of human behaviour, in this study, is going to be done through the analysis of motion patterns that derive from responses involving cognitive, emotional and executive functions describing human behaviour.

2.1.1 Motion patterns discovery

Motion patterns are repeated observations of motion that can be used to infer about human behaviour, including patterns in trajectories and activities. Human trajectory patterns can be assessed by the similarity between trajectories, based on location, velocity between locations and permanence time in a specific place [32]. Common features used in literature for trajectory analysis are subject Points of Interest (POI) and Times of Interest (TOI) [33]. Measuring trajectory similarity through POI has the purpose of finding

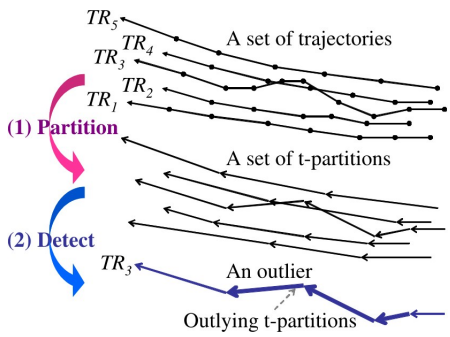


Figure 2.1: Representation of a set of trajectories being partitioned into a set of t-partitions. The outlying t-partitions are identified based on the distance from neighbouring trajectories [36].

specific interests of the subject being analysed in his/her normal routine. This type of analysis can be done in a more restrictive environment, such as a home, using indoor location methods. For instance, the time passed in the kitchen can give us information about personal interest in cooking. The motion patterns discovery can also be done on a wider scale, such as a commuting routine, using for this purpose a combination of indoor and outdoor location methods. Clustering methods are frequently used to divide trajectories into clusters of similar trajectory considering spatial or spatio-temporal similarity [34].

Motion patterns are also found in human activities, by analysing the sequence of performed activities, their duration and the number of times each activity is executed [26]. The pursuit for activity patterns is based on efficient methods in pattern data mining such as HMM to evaluate the probability of activities sequences and statistical models to evaluate duration and activities occurrence.

2.1.2 Anomaly detection

Once motion patterns are defined, it is possible to detect anomalies on those patterns that are characterised as a pattern deviation. Atallah and Yang [35] define anomalies as a deviation from learned models of behaviour and also state that classes distance can provide information about the anomaly level. Anomalous trajectories can be detected, for example, through distance [36] and clustering-based methods [37]. Distance-based outliers were defined by Knorr et al. [38] as "An object O in a dataset T is a DB(p, D)[2]-outlier if at least fraction p of the objects in T lies greater than distance D from O". An example of a distance-based outlier trajectory is shown in Figure 2.1.

Clustering-based methods cluster trajectories into groups by first measuring trajectory similarity and defining the clustering method. Anomalous clusters are the ones that contain only one trajectory [37].

Considering a trained HMM by activities or indoor locations sequences, the log-likelihood output can be modelled through a statistical distribution and an anomaly will

[2]DB(p, D) is shorthand notation for a Distance-Based outliers detected using parameters p and D [38].

be given by the behaviour that generates a HMM log-likelihood output with a probability value that is too different compared with the expected behaviour [27].

2.2 Motion detection using pervasive sensing

Learning human motion patterns on everyday basis compromises activity recognition methods and trajectory analysis. In this section, a brief description of Human Activity Recognition (HAR) and location technologies is introduced, as well as the sensors used for this detection.

2.2.1 Human Activity Recognition

HAR is an important and challenging research area in computer vision and machine learning that studies gestures and human motions through the use of sensors for the recognition of activities. It contains many important applications in health-care, life-care, smart environments, and homeland security [39]. The existing studies of HAR suggest the existence of two main classes of activity recognition, namely simple and complex activities. Simple activities are characterised by different body posture and locomotion of humans, comprising more repetitive motions, such as *walking*, *standing*, *lying*, *sitting*, *walking up* and *walking down* [11]. Complex activities are constituted by multiple simple activities and a specific function involving cognitive functioning [40].

Activities of Daily Living (ADLs) belongs to the set of complex activities and includes daily activities such as *bathing*, *toileting*, *brushing teeth*, *eating* and *cooking*. With the recognition and analysis of these activities, it is possible to infer about cognitive and physical capabilities of elderly people [41]. Thus, pathological behaviour related to physical activity can be detected, allowing early diagnosis of diseases and intervention procedures [42].

There are different approaches for HAR, some studies recognise human activity using cameras [43], external and wearable sensors [44]. HAR using cameras has obvious privacy issues associated because not everyone is willing to be continuously monitored by a camera. Intelligent homes using a wireless sensor network allows to observe what goes on in the house and infer human activities from sensor data [44], an issue associated to this method is maintenance of the sensors which are usually expensive, moreover, nothing can be done to recognise human activity if the person is out of the sensors range [45]. In recent years, smartphones usability has increased and because of the availability of various sensors in these devices, for being unobtrusive and also an affordable asset, there has been a shift towards smartphones to perform HAR [46, 47].

2.2.2 Location Technologies

Outdoor location positioning uses GPS to calculate users exact location by using signal information transmitted through a satellite-based navigation system [48]. Regarding indoor location, GPS based approaches are less effective because GPS signal is attenuated and

scattered by roofs, walls and other objects inside buildings [49]. Indoor location solutions created to overcome this problem are divided into two main categories: Infrastructure-free and infrastructure-based. Radio frequency signals such as Bluetooth, where beacons are used to transmit a specific signal to the device being located, are one of the most common applications of infrastructure-based solutions [50]. Although Wi-Fi signals depend on infrastructure, they can be considered as an infrastructure-free solution, due to its wide implementation inside buildings. Inertial sensors, magnetic field and atmospheric pressure are commonly used for infrastructure-free solutions [16].

The indoor location has been a useful tool for many applications such as tracking of medical equipment's or visitors in hospitals, locate books in libraries, shopping mall navigation, fire emergency rescue and assistant living systems for elderly people. For the analysis of human trajectory, the main location technologies used in this study are the GPS, Wi-Fi and Dead Reckoning (DR) approach.

2.2.2.1 Global Positioning System

GPS is a satellite-based navigation system that allows calculating user's location expressed by latitude, longitude and altitude together with timing information. GPS is constituted by three segments: the space segment, the control segment and the user segment. The space segment consists of a constellation of 24 to 32 satellites [51], where each GPS satellite transmits a microwave radio signal composed by two carriers frequencies, two digital codes and a navigation message. The carriers and codes are used to determine the distance between the GPS receiver and the satellite and the navigation message contains the coordinates of the satellites as a function of time. The control segment is a worldwide network of tracking stations that track GPS satellites providing multiple control information [52]. The user segment includes all users with a GPS receiver. User's location is determined by employing a triangulation process that considers distances from GPS receiver to three GPS satellites along with the satellites' locations. If a fourth satellite is available, altitude information can also be determined. GPS enabled smartphones are accurate within a 5-meter radius under open sky [53]. However, GPS accuracy is reduced near buildings, tunnels, bridges because GPS signal is attenuated and scattered by roofs, walls and other objects [49].

2.2.2.2 Wi-Fi Technology

Wi-Fi devices are used for location estimation due to its wide implementation inside buildings. Taking into account that all Wi-Fi Access Point (AP) has a unique ID called Basic Service Set Identifier (BSSID) that is transmitted periodically to any Wi-Fi client, it is possible to estimate client's position by mainly 3 technologies: Proximity, Triangulation and Scene Analysis (Fingerprinting):

- **Proximity** is the most simple method that uses AP's location with the strongest signal to estimate the user's position. It is a simple, fast real-time positioning

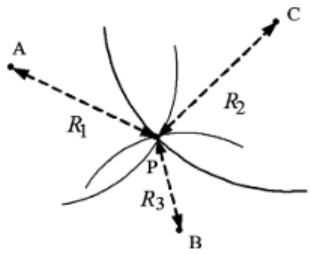


Figure 2.2: Lateration technique. P position is obtained based on the distance from P to three reference points A, B, C [55].

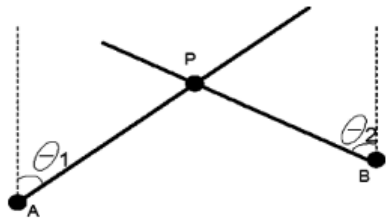


Figure 2.3: Angulation technique. P position can be found by the intersection of several pairs of angle direction lines [55].

method, however, it has limited accuracy, since it returns the position within the range of the nearest AP, and in the case of multi-floor buildings, it can not provide accurately in which floor the user is located due to signals overlap [54].

- **Triangulation** uses geometric properties of triangles to estimate the user's position. It includes lateration (Figure 2.2) and angulation (Figure 2.3) techniques. Lateration estimates position by measuring the user's distance to multiple reference points. On the other hand, angulation locates the user by measuring angles relative to multiple reference points [55].

- **Scene analysis** includes algorithms based on fingerprints. This method is divided into two stages: online and offline stages. The offline stage comprises features collection of a scene, where fingerprints store the signal strength of each location coordinate, as it is presented in Figure 2.4. The online stage allows estimating location by comparing currently observed signals strengths with the closest a priori location fingerprints, collected during the offline stage [55]. For fingerprinting methods to correctly estimate the position it is needed a sufficient amount of collected information beforehand, these acquisitions are time-consuming and error prone [56]. The main challenge of this technique remains in the fact that Received Signal Strength Indicator (RSSI) can be affected by diffraction, reflection and scattering during its propagation [55]. Therefore, the search for a more accurate and efficient indoor position system still continues.

2.2.2.3 Dead Reckoning technology

DR is an inertial navigation estimation technology that uses data collected from the accelerometer, magnetometer and gyroscope to estimate user's current position by using a previously determined position, heading and velocity [16, 57]. The accelerometer is used to detect steps and step length. The magnetometer is used to determine the Earth magnetic north to be used as a reference to determine movement direction. Gyroscope is

Figure 2.4: Wi-Fi RSSI map of an office building. Signal intensity is represented by colours, red colour means more intense [16].

usually used together with a magnetometer to improve the estimation of movement direction mainly in indoor environments since ferromagnetic materials cause disturbances in magnetometer readings [58].

2.2.3 Smartphone Sensors

To recognise human activity and location, smartphones are going to be used since they contain several sensors capable of measuring physical quantities in our surrounding. In the context of this study, the most relevant sensors are described bellow.

- **Accelerometer** measures acceleration in m/s^2. Smartphones use a tri-axial accelerometer sensor that can measure linear acceleration, relative to the Earth's, in three orthogonal axes determining the acceleration parts of each spatial direction (x,y and z-axis), as it can be seen in Figure 2.5. The obtained signal contains a dynamic component caused by smartphone movement and a static component that is gravity acceleration force.

- **Gyroscope** provides angular velocity information in rad/s in three orthogonal axis. Orientation over time can be calculated by the integration of gyroscope data resulting in rotation angles around x, y and z-axis, namely pitch, roll and yaw (Figure 2.5).

- **Light Sensor** provides the measurement of surrounding illumination intensity in SI lux units.

- **Microphone** records a sound signal using a transducer that converts a sound wave, that consists in a pressure wave, into an electrical signal proportional to its pressure [59].

- **Magnetometer** provides the orientation of the device relative to the Earth's magnetic north pole by measuring the strength of the local magnetic field along three orthogonal axes in μT. The local magnetic field obtained combines the geomagnetic field and magnetic field from the environment.

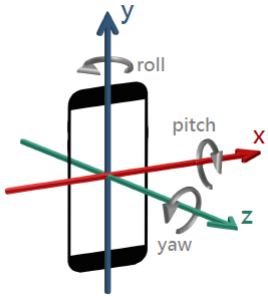


Figure 2.5: Smartphones coordinate system [61].

- **Barometer** pressure sensor provides the altitude of the device relative to its initial position by measuring the atmospheric pressure, in hPa or mbar, to infer about altitude above the sea [60].

- **Wi-Fi** is a wireless networking technology, that contains devices called AP that use radio waves to establish a communication between the Wi-Fi device and a wired network, typically the Ethernet. Smartphones contain a Wi-Fi receiving module that allows the connection to a wired network, from which we can extract the RSSI, BSSID and the frequency over which the client is communicating with the AP in MHz.

The previous section detailed the main areas and the types of sensors that can be used to obtain relevant information that reflects a change in subjects or surrounding. However, developing methods that can accurately model the true nature of human behaviour remains a challenge [35]. Using the previous knowledge of HAR and location technologies, behaviour can be modelled, through which patterns and deviations to those patterns can be detected.

2.3 Machine Learning

Machine learning is the field of computer science derived from artificial intelligence that gives the capability to computer programs to learn without being explicitly programmed. According to Tom M. Mitchel [62] machine learning definition is "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E". Machine learning includes several algorithms based mainly in four types of learning: Supervised learning, Unsupervised learning, Semi-supervised learning and Reinforcement learning.

- **Supervised learning** uses a labelled dataset to train the classifier, which means, the input, sample x, of the classifier has the output, prediction y, already defined. After the training procedure, the classifier is capable of predicting new outputs.

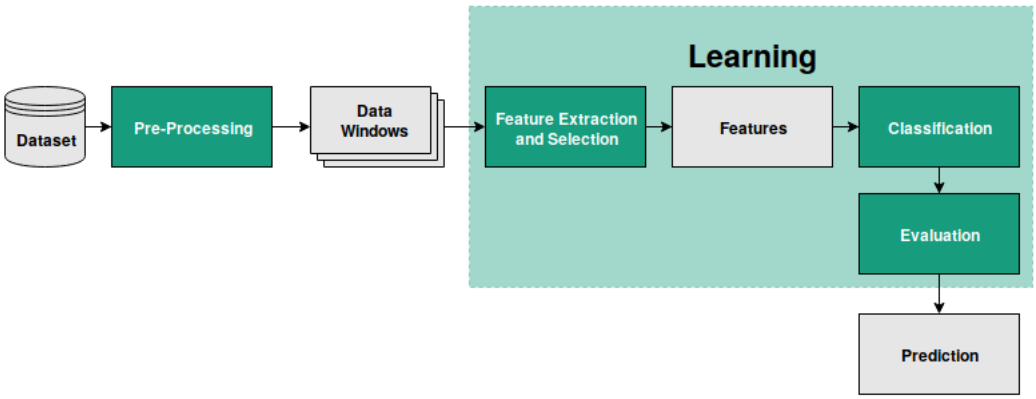


Figure 2.6: Machine learning classification architecture. The grey objects represent the variables inputs/outputs from the processes coloured in green.

- **Unsupervised learning** trains the classifier with an unlabelled dataset, finding patterns in the dataset relying on data similarity.
- In **Semi-supervised learning** the classifier is trained using a labelled and unlabelled dataset, containing more unlabelled data than labelled.
- **Reinforcement learning** trains the classifier that learns by interacting with a dynamic environment using the experience gathered to optimise its performance. In other words, by a trial and error method, it must reach a certain goal in the interacting environment such as winning a game.

Figure 2.6 represents the machine learning classification architecture. The dataset is segmented into equal-sized data windows from which features are extracted, followed by the implementation of a machine learning classifier, a validation procedure, and lastly, the prediction result.

2.3.1 Pre-processing

Before employing a machine learning algorithm, data needs to be processed to enhance relevant signal properties. In the case of a continuous stream of data, segmentation techniques are usually used to process this data. Data is segmented into equal-sized windows, whose length can be defined according to the recognition problem at state. Additionally, data windows may or not overlap depending if it is important to detect transitions between activities [63] or to enhance relevant features properties that may be overshadowed by segmenting the signal into a fixed window size.

Determining the appropriate window size is a common challenge in the scientific community [64, 65]. Shorter windows may be useful for the recognition of repetitive activities or activities with short duration [66] and it has the advantage of reducing energy consumption and faster recognition. On the other hand, the short window size is not

appropriate for the recognition of long complex activities that are not so repetitive. Thus, longer complex activities need to be recognised using a larger window size that allows enhancing relevant features of the activity [67]. Hence, it is important the selection of the window size, depending on the recognition problem.

After data is segmented into windows it is important to properly balance the dataset. An imbalanced dataset contains more samples of some classes than others. This is a common problem when working with standard machine learning algorithms because these algorithms expect a balanced dataset. When a complex and imbalanced dataset is provided, they tend to choose the classes with more samples and ignore the small ones [68]. Several solutions were developed to solve this problem at the data level including re-sampling methods and the algorithm level involving the adjustment of the cost of the various classes [69, 70].

2.3.2 Feature Extraction

Feature extraction is an important stage of machine learning algorithms, it is when each data window is transformed into a low dimensional subspace preserving most of the relevant information [71]. Features are typically extracted from time, spectral and statistic domains. A python library developed by Fraunhofer Portugal, named Time Series Feature Extraction Library (TSFEL)[3], extracts a total of 43 features including 12 from the temporal domain, 18 from the spectral domain and 13 from the statistical domain. A detailed description of TSFEL is presented in Appendix A. After feature extraction, features need to be normalised to a common scale so that new extracted features can contribute equally to the classifier. In this work, features were normalised by a Z-score normalisation, resulting in a zero mean and standard deviation of one, as it can be seen in Equation 2.1:

$$Z_n = \frac{X_n - \mu}{\sigma} \tag{2.1}$$

Where, X_n is the feature, μ the mean and σ the standard deviation. The mean and standard deviation from the training set are parameters that need to be learned to be used for the normalisation of the new data for prediction.

2.3.3 Feature Selection

Feature selection algorithms are one of the most important in a classification process because they allow to remove features with redundant, misleading and irrelevant information, reducing computation time, overfitting, complexity and improving the learning process of the classifier. Feature selection methods can be subdivided into Filters, Wrappers and Embedded methods.

[3] Available in https://github.com/fraunhoferportugal/tsfel (visited on 03/09/2019)

- **Filters:** Filter methods, have low computational cost and time, they do not rely on the classifier being used, and evaluates features based on general characteristics of the data [72].

- **Wrappers:** Wrappers methods are dependent on the machine learning classifier being used, having a better performance than filter methods. However, they present higher computational cost and time, because each feature set needs to be evaluated with the trained classifier [73].

- **Embedded:** These methods use both advantages of wrappers and filter methods.

2.3.4 Classification

In a machine learning classification process the training procedure can be divided into supervised and unsupervised learning methods.

2.3.4.1 Supervised Learning

This subsection describes the commonly used classifiers in supervised recognition problems.

Decision Tree constitution comprises nodes and a hierarchy of branches. There are three types of nodes, the root or decision node, the internal or chance node and the leaf or end node. The root node consists in the choice that leads to the subdivision of all samples in two or more exclusive subsets. The internal node corresponds to one of the possible choices available, its top edge is connected to the parent node and the bottom to the sun node. The outcomes of root and internal nodes are named branches. At each node, if-then-else rules are applied and after all the decision events, the leaf node is reached representing the final result [74].

K-Nearest Neighbors algorithm can be divided into mainly two stages, the first consists in determining the nearest neighbours and the second, according to the determined nearest neighbours, to decide the corresponding class. The nearest neighbours are selected by computing the distance, for example, the euclidean distance [75], between an unlabelled sample and a training labelled dataset with a class label. The nearest neighbours will be those at the shortest distance. The most straightforward approach to decide the corresponding class is to assign the majority class of the nearest neighbours.

Random Forest is a combination of decision trees, where each tree node contains a set of features randomly selected from the whole set of attributes, as employed in the bagging method [76]. The classification result arises from majority voting.

AdaBoost classifier combines many weak classifiers and it is based on an iterative algorithm [77]. It first starts with an unweighted training set, according to the classification results, the weight of the miss-classified samples, in the next iteration, is increased, i.e boosted. This way, the next classifier has no longer equal-weighted samples. The

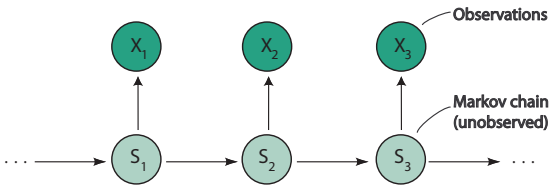


Figure 2.7: Hidden Markov Models chain, where S refers to hidden states and X to observations. Adapted from [79].

procedure continues, each classifier gets a score, and the final classifier will be a linear combination of the classifier's scores at each stage.

Naive Bayes classifier assigns the most likely class, y, for a sample , x, according to Equation 2.2:

$$\hat{y} = \underset{y}{\operatorname{argmax}} P(y) \prod_{i=1}^{x} P(x_i|y) \tag{2.2}$$

Where $\hat{y}$ is the assigned class, x is a feature sample, and $P(y)$ is estimated using the *Maximum A Posteriori* estimation and the likelihood of features is assumed to be Gaussian:

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} exp(-\frac{(x_i - \mu_y)^2}{2\pi\sigma_y^2}) \tag{2.3}$$

Where σ_y and μ_y are estimated by maximum likelihood. This classifier works by assuming, in a naive way, the independence between features given a class, according to Bayes rule (Equation 2.4), where y is the predicted class, and $(x_1, x_2, ..., x_n)$ is the feature vector.

$$P(y|x_1, x_2, ..., x_n) = \frac{P(y)P(x_1, x_2, ..., x_n|y)}{P(x_1, x_2, ..., x_n)} \tag{2.4}$$

Hidden Markov Models are effective to infer about hidden states (S) of a system that gives rise to a time series of observations. It is constituted by a set of hidden states, which variables are discrete, called Markov Chain, observations generated by some process whose state is hidden from the observer, by the probability of transitioning from one state to another, including self-transitions and by emission probabilities (see Figure 2.7) [78].

HMM satisfies Markov property, well described by Ghahramani [80], which states that "an n-th order Markov process is one in which S_t given S_{t-1} ... S_{t-n} is independent of S_τ for τ<t-n.". This means that the Markov chain bases its state choice only in transition probabilities from the previous state.

According to Rabiner [81], hidden states sequence can be estimated from a given model $\lambda = (A, B, \pi)$, where A corresponds to state transition probability matrix, B refers to emission probability matrix and π the initial state distribution vector.

However, to properly estimate the hidden states the following three basic problems are addressed:

- Problem 1: Viewed as an evaluation problem, how do we compute the probability of a given sequence and the model, $P(O|\lambda)$, where O is a sequence of observations and λ a model. In other words, how well a given model matches a given observation sequence.
- Problem 2: Attempt to uncover the hidden states of the model, given the observation sequences O and the model λ, how do we find the optimal state sequence.
- Problem 3: Optimisation of the model parameters $\lambda = (A, B, \pi)$ to maximise $P(O|\lambda)$.

2.3.4.2 Unsupervised Learning

Unsupervised learning comprises clustering methods and allows to model probability densities of input data.

Clustering methods divide data into groups according to their similarity, without the need of knowing previous knowledge of the group definitions. Each group of similar data is called a cluster. Clustering techniques satisfies two conditions, the objects in each group need to be similar to each other and different from objects contained in other clusters. Clustering techniques are divided into hierarchical methods, partitioning methods and density-based algorithms.

- **Hierarchical Methods:** In hierarchical methods, clusters are divided hierarchically, not in a single step. Thus, it is possible to obtain different partitions of data, from only one cluster containing all objects, to x clusters containing only one object [82]. Hierarchical methods are subdivided into agglomerative and divisive hierarchical clustering [83]. In agglomerative hierarchical clustering, initially each object represents one cluster, then clusters are grouped to originate the desired cluster structure. Meanwhile, in divisive hierarchical clustering, initially all objects belong to one single cluster and then the cluster is subdivided into smaller clusters until the desired cluster structure is obtained. The result of the hierarchical clustering methods is a dendrogram that illustrates the grouping or divisions made at each successive stage of the clustering analysis, an example is shown in Figure 2.8.
- **Partitioning Methods:** Partitioning methods start with an initial partitioning, where the number of clusters is pre-selected by the user. For the adequate partitioning, iterative optimisation of the cluster centres is performed [83].
- **Density Based Methods:** Density-based methods are used for discovering clusters with arbitrary shape. The objects are grouped according to a specific density function. In this method, it is necessary to define the minimum number of objects contained in a given neighbourhood radius, and the cluster keeps growing as long

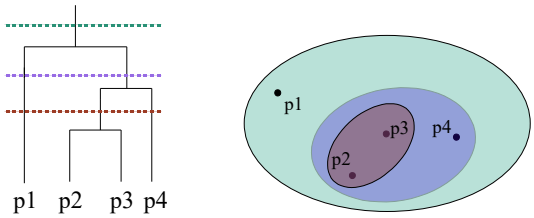


Figure 2.8: Representation of a dendrogram and corresponding clusters. The left part of the figure illustrates the dendrogram, whose dashed coloured lines represent the divisions giving origin to the corresponding coloured cluster on the right part of the figure. Adapted from [84].

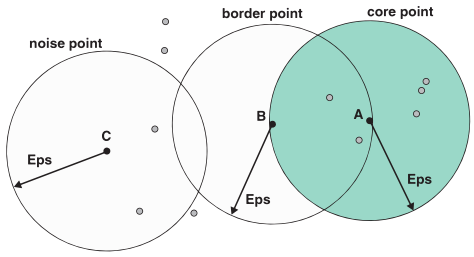


Figure 2.9: Representation of a core, border and noise points according to DBSCAN, considering $Minpts <= 7$. A point is a core point since its density region, coloured in green, contains 7 points, B point is a border point as it is on the edge of a density region and C point is a noise point since is neither a border or core point. Adapted from [84].

as the density, number of objects, in the neighbourhood exceeds a given parameter. Density-based methods are useful to find outlier objects [83]. Common density based methods include Density-based spatial clustering of applications with noise (DBSCAN) and Hierarchical Density-based spatial clustering of applications with noise (HDBSCAN).

1. **DBSCAN** is an effective method for separating a dataset with different density regions. The density of a specific point of the dataset is given by the number of objects within a given radius from that point, this radius is an input parameter in DBSCAN methods called *Eps*. The minimum number of objects inside a region, *minPts*, is also an input for DBSCAN algorithm [84]. DBSCAN algorithm works as described in the following.

   Initially, each point in the dataset is going to be classified as:

   - **Core point** if it is inside a density region, namely if the number of points within a given radius *Eps* exceed the *minPts* (see Figure 2.9).
   - **Border point** if it is on the edge of the density region (see Figure 2.9).
   - **Noise point** if it is in a sparsely occupied region (see Figure 2.9).

   Next, noise points will be eliminated. An edge will separate all core points that are within *Eps* of each other into density-based clusters. Finally, each border

point will be assigned to one of the clusters of its associated core points.

This algorithm has the advantage of not being necessary to define the final number of clusters, allowing the discovery of clusters with arbitrary shape and size and for being a noise-resistant method. However, it also presents some disadvantages, regarding the search of the optimal *Eps* value, some problems in the DBSCAN method when working with clusters whose density varies widely and if the dataset is constituted by high-dimensional data.

2. **HDBSCAN** is a hierarchical version of DBSCAN that is capable of dealing with clusters of different densities. Moreover, it is an improvement of DBSCAN as *Eps* parameter is not required. It has an input parameter *minPts*, and for a given *minPts* value the HDBSCAN algorithm computes the distance required to obtain that *minPts*, and produces a clustering tree containing all clusters obtained by DBSCAN, hierarchically. The clustering tree includes nodes that denote the transition of the different density levels that are hierarchically tested to optimise the value that provides the best results [34].

For modelling the probability density function of datasets whose function is unknown, the density estimation used in this study is KDE.

Kernel Density Estimate was firstly defined by Rosenblatt [85] and Parzen [86], the kernel estimate is given by the sum of the kernel function K placed at each point of the dataset, as it is defined in Equation 2.5.

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - X_i}{h}\right) \tag{2.5}$$

Where n is the number of points in the dataset, h is the bandwidth (also called smoothing parameter) and K is the kernel function centred on X_i with width h. The kernel function K needs to satisfy:

$$\int_{-\infty}^{+\infty} K(x)dx = 1 \tag{2.6}$$

This condition requires the normalisation of the kernel estimate that is ensured dividing it by n. This way it is guaranteed that the probability density function has a total probability of one. Moreover, K is usually a symmetric function and it is non-negative since probabilities are non-negative.

Examples of symmetric kernel functions are Epanechnikov, Biweight, Triangular, Gaussian and Rectangular [87], illustrated in Figure 2.10. Since these functions are symmetric they will have little impact on the choice of the estimator. Thus, in this study it will be used the Gaussian function, given by Equation 2.7:

$$K(x) = \frac{1}{\sqrt{2\pi}} e^{\frac{-x^2}{2}} \tag{2.7}$$

One of the challenges on the implementation of this algorithm is the choice of the optimal h parameter if it is too big the probability density function will be spread, not

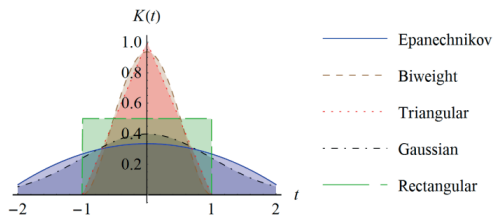


Figure 2.10: Illustration of symmetric kernel functions namely Epanechnikov, Biweight, Triangular, Gaussian and Rectangular, whose labels lines are coloured with the corresponding distribution colour [87].

revealing structural features. If h parameter is too little, the probability density function will be too narrow, showing spurious features [88]. Hence, the division by h inside and outside the kernel function in Equation 2.5 is to guarantee that the growth of K equals one.

There are automatic methods for calculating the optimal smoothing parameter including the rule of thumb, maximal smoothing principle, cross-validation methods, plug-in methods, well described in Turlach review [88].

The rule of thumb defined by Silverman [89] will be the method for calculating the optimal smoothing parameter in this study. This method assumes that data is normally distributed with a standard deviation, σ, and h will be the value that minimises the integrated mean square error, given by Equation 2.8.

$$h = 1.06\hat{\sigma}n^{-\frac{1}{5}} \tag{2.8}$$

2.3.5 Validation

Validation is one important stage of machine learning algorithms, to properly build a classifier. Datasets used in the machine learning algorithm are usually divided into three sets: a training set, a testing set and a validation set. The subdivision of the dataset is important to avoid that the classifier ever sees the evaluation data, preventing the occurrence of overfitting [90].

Model validation techniques include k-fold cross-validation, split sets, leave one out and bootstrapping.

- **K-fold cross-validation:** In K-fold cross-validation the dataset is subdivided into equal-sized K folds, the classifier is trained in k iterations using one fold to validate the results and the rest for training.

- **Split sets:** Split sets consists in dividing a percentage of the dataset for training and the remaining for performance evaluation.

- **Leave one out:** Leave one out validation, presents a higher computational cost, because it leaves out one sample for validation and trains with all the other, having

Table 2.1: Confusion Matrix, where TP - True positive, TN - True negative, FP - False positive, FN - False negative.

		Predicted Labels	
		Positive	Negative
True Labels	Positive	TP	FN
	Negative	FP	TN

the number of iterations equal to the dataset size. On the other hand, this technique returns more accurate results.

- **Bootstrapping:** Bootstrapping uses a dataset of size n to create bootstrap samples, with n instances, by resampling the dataset iteratively with replacement, using one bootstrap sample for testing and the rest for training [91].

After the classifier is trained, several metrics can be used to evaluate its performance [92], described in the following.

Confusion matrix is an N×N matrix that allows visualising the classifier performance, it is constituted by N rows indicating the true labels of the samples being classified, and N columns that correspond to the predicted labels. An example is represented in Table 2.1.

One of the most common metrics for evaluating performance is the **accuracy** that corresponds to the number of successful classifications divided by the total number of classifications.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{2.9}$$

Sensitivity transmits the ability of the classifier to classify correctly all true labels, and it is calculated by the number of successful classification from one specific class divided by the total number of successful classifications.

$$Sensitivity = \frac{TP}{TP + FN} \tag{2.10}$$

Specificity indicates how well the negatives are detected and corresponds to the number of true negatives divided by the total number of negatives. The lower the specificity, higher are the false positives.

$$Specificity = \frac{TN}{TN + FP} \tag{2.11}$$

Precision is the number of true positives divided by all the predicted labels.

$$Precision = \frac{TP}{TP + FP} \tag{2.12}$$

F1-score relies on the weighted average of the precision and sensitivity, returning a number on a scale between 0 and 1, where 1 is the best score.

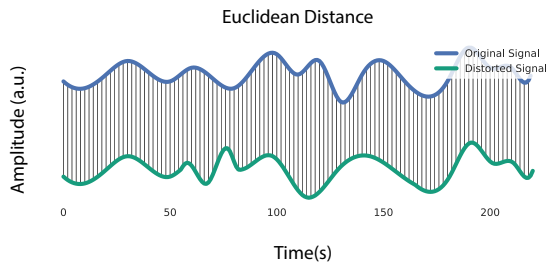


Figure 2.11: Distance measurement of two time series using the Euclidean Distance. Adapted from [61].

$$F1 = \frac{2 \times (precision \times sensitivity)}{precision + sensitivity} \tag{2.13}$$

Receiver operating characteristics curve also allows a visual analysis of the classifier's behaviour as it illustrates the performance of the true positive rate against the false-positive rate regarding different threshold values. Receiver Operating Characteristics (ROC) is reduced to a single value by calculating the Area Under the Curve (AUC). Ideally, a classifier has an AUC of one.

2.4 Similarity distances

In the context of human motion patterns, the analysis of similarities between time-series is a valuable tool to extract human routines through distance metrics. In the following subsections, a description of the distances addressed in this study is presented.

2.4.1 Euclidean Distance

Euclidean Distance (ED) can be defined as:

$$D(X,Y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{2.14}$$

Where X and Y are two time series given by $X = x_1 ... x_n$ and $Y = y_1 ... y_n$.

This is a simple approach to measure distance similarity (see Figure 2.11) and its principal advantage is the linear time complexity that is held. ED main limitation consists on the requirement of the two analysed time-series must have the same length, otherwise, addition, removal of point or the search of a partial alignment using a sliding window is required [93].

2.4.2 Dynamic Time Warping

DTW can measure the similarity between two time series, warping them non-linearly to align non-linear changes in time domain, as it is shown in Figure 2.12. Given two

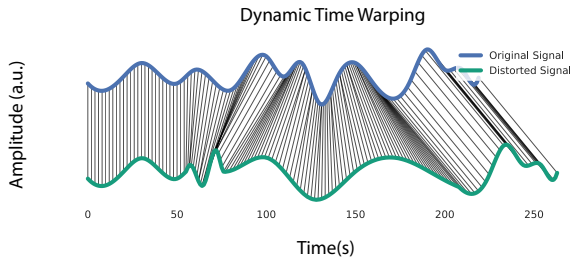


Figure 2.12: Similarity measurement using Dynamic Time Warping. Adapted from [61].

sequences $X := (x_1, x_2, ..., x_N)$ and $Y := (x_1, x_2, ..., x_M)$, an accumulated cost matrix with dimensions $N \times M$ is computed, where the first column is given by Equation 2.15 and the first row by Equation 2.16, where c denotes the local cost, or local distance, that is given by the absolute distance between x and y (with $x \in X$ and $y \in Y$) [94]. Equations 2.15 and 2.16 correspond to the cumulative distance (or cumulative cost) along the first column, row, respectively.

$$D(n,1) = \sum_{k=1}^{n} c(x_k, y_1) \; for \; n \in [1:N] \tag{2.15}$$

$$D(1,m) = \sum_{k=1}^{m} c(x_1, y_k) \; for \; m \in [1:M] \tag{2.16}$$

Hereafter, the remaining accumulated cost matrix positions can be calculated by Equation 2.17, that corresponds to the cumulative distance between each point of the two sequences [94].

$$Cd(i,j) = c(i,j) + min \begin{cases} Cd(i-1,j-1) \\ Cd(i-1,j) \\ Cd(i,j-1) \end{cases} , i \in [2:N] \; and \; j \in [2:M] \tag{2.17}$$

After the accumulated cost matrix is computed, DTW finds the optimal warping path, which means, the path with the minimum cumulative distance, by satisfying the following three conditions [94].

Boundary conditions - the first and last point of the two trajectories should match. Thus, it is a guarantee that all sequence is considered.

Monotonicity conditions - the alignment does not go back in time, ensuring that there are not replications in the alignment.

Continuity conditions - there are no omitted samples in time, to ensure that all samples are considered.

The DTW distance will be given by the accumulated distance at [N, M] accumulated cost matrix position.

2.4.3 Subsequence - Dynamic time Warping

Subsequence-DTW (S-DTW) is a modification of the classical DTW, defined by Muller [94], that finds matching subsequences between a sequence $X := (x_1, x_2, ..., x_N)$ and $Y := (x_1, x_2, ..., x_M)$, where the length of $Y(M)$ is much larger than the length of $X(N)$. The goal is to find the subsequence that optimally matches Y, given by $Y(a^* : b^*) = (y_{a^*}, y_{a^*+1}, ..., y_{b^*})$, where $1 \leq a^* \leq b^* \leq M$, that corresponds to the subsequence that minimises the DTW distance to X (i.e. Equation 2.18).

$$(a^*, b^*) = \underset{(a,b):1 \leq a \leq b \leq M}{\text{argmin}} \quad (DTW(X, Y(a : b))) \tag{2.18}$$

To obtain a^* and b^* the cost matrix needs to be modified. In the classic DTW, the distance in the first column is given by Equation 2.15 and the first row by Equation 2.16, which means they are defined by the cumulative distance (or cumulative cost) given by c, that is referred as a local distance between the two points [94].

In S-DTW the first column is given by 2.15 but the first row is given by $D(1, m) = c(x_1, y_k)$ for $k \in [1 : M]$. Hence, all points $(1 : M)$ are possible candidates to starting points for the optimal matching subsequence. Subsequently, the cost matrices is computed, and the optimal matching subsequence ending point in b^* is given by Equation 2.19.

$$b^* = \underset{b \in [1:M]}{\text{argmin}} D(N, b) \tag{2.19}$$

This way, the optimal matching sequence can end in any position along the sequence Y. Once, the last position of the subsequence is found, a track-back dynamic programming procedure finds the optimal first position of the optimal matching function, that is given by a^*. After the sequence is found, all positions around (N, b) will be set to a large value and the search for the next best end position starts [95].

Chapter 3

Framework for Learning Human Patterns Behaviour

Human behaviour analysis presents unique challenges to learn and understand human patterns that can not be addressed with conventional approaches. This chapter presents a framework for modelling human motion patterns behaviour. The key insight is to exploit the human motion by trajectory analysis and activity recognition, effectively capturing both indoor and outdoor environment.

An overview of the framework is presented in Figure 3.1 where each section is represented with different colours.

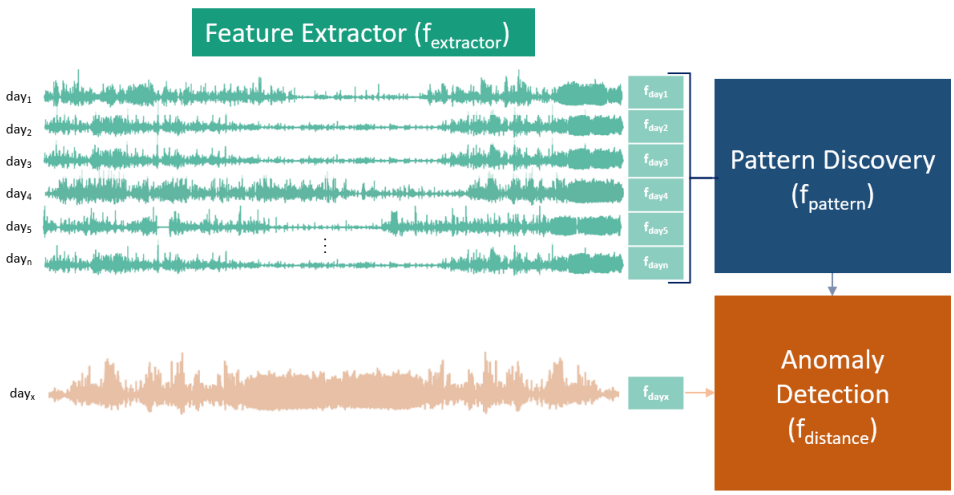


Figure 3.1: Representation of framework overview, first features are extracted from signals coloured in green (1st Section), that are used to define a pattern (2nd Section), then from a new day, features are extracted and the distance to the pattern will determine whether the day is or not anomalous (3rd Section).

The first section describes the methods applied for the feature extraction along each day, the second section uses the extracted features from a set of days to learn patterns, and the last section describes the process for anomaly detection using the previously detected patterns and features from a specific day.

3.1 Feature Extractor

The extraction of relevant features for the human motion pattern analysis can be simplified to the Equation 3.1 where day_i represents signals acquired from smartphone sensors along a day and $f_{extractor}$ is a set of the methods and models used to extract features that are represented by f_{day}. The feature extractor can be applied to unsupervised and supervised methods depending on the available information.

$$f_{day_i} = f_{extractor}(day_i) \tag{3.1}$$

3.1.1 Unsupervised Feature Extraction

This section describes the extracted features from human behaviour in an unsupervised manner, specifically all features that may be extracted without any previous annotation. Extracted features can be grouped by the type of information source used and are divided into the outdoor trajectory, DR and simple activities.

3.1.1.1 Outdoor Trajectory

Outdoor trajectories are acquired through GPS sensor embedded in smartphones. As mentioned in 2.2.2.1, GPS accuracy is reduced near buildings, walls and other objects that scatter and attenuate this signal, leading to inaccurate results. Figure 3.2a represents one trajectory that is clearly affected by GPS inaccuracy.

To use GPS signal to extract features such as walking velocity or distance, the correction of GPS outlier points must be done.

The Kalman filter is a well-known method and is usually referred in literature for noise reduction in GPS signal [96]. Kalman filter is a recursive and mathematical algorithm that processes inaccurate observation input data and generates a statistically optimal estimate of the next state by employing a prediction and observation model, where the prediction model estimates the next state considering the last measurement and the movement dynamics, assuming constant acceleration. The observation model retains the real measurements. Therefore, the first approach for correcting GPS outlier points was the implementation of a Kalman Filter, however, as the GPS measurements were too noisy, the Kalman filter was not very efficient in correcting these measurements as it can be observed in Figure 3.2b.

(a) Raw signal. (b) Kalman filter approach. (c) Algorithm 1 approach.

Figure 3.2: Process for correcting GPS trajectories. Figure 3.2a represents the raw GPS signal. Figure 3.2b the outlier's correction using a Kalman filter, where the original signal is coloured in blue and the corrected in green. Figure 3.2c corresponds to the outlier's removal using Algorithm 1.

Thus, instead of correcting GPS outlier points, an approach that relies on removing these points was applied.

Given a GPS trajectory, Tr, where each point Tr_i, is denoted by latitude, longitude and time and $i \in [0, 1, ..., len(Tr)]$, the first step was to convert the GPS coordinates from degrees to meters to estimate the maximum velocity between consecutive points and remove the ones with an inconsistent velocity. The threshold velocity for removing the GPS outlier points was empirically tested resulting in a threshold of 400 m/min ($\simeq$ 6.7 m/s).

The implementation is presented in Algorithm 1 and the obtained results can be observed in Figure 3.2c.

Algorithm 1 GPS outlier removal

INPUT: $Tr \Leftarrow$ GPS raw signal in meters
OUTPUT: $Trp \Leftarrow$ GPS signal without outliers
$d_i = euclidean\_distance(Tr_i, Tr_{i+1}) \Leftarrow i \in [0, 1, ..., len(Tr)]$
$v_i = \frac{d_i}{\Delta t}$
$id\_del = where(v_i > threshold)$
while $len(id\_del) > 0$ **do**
    Delete GPS points with i $\in id\_del$
    $d_i = euclidean\_distance(Tr_i, Tr_{i+1})$
    $v_i = \frac{d_i}{\Delta t}$
    $id\_del = where(d_i > threshold)$
end while

Human behaviour tends to promote well-defined motions which are repeated every day, to access human behaviour motion patterns through GPS trajectories, some features were extracted from the preprocessed GPS signal measurements. GPS signal outputs include timestamped geographic coordinates such as longitude and latitude, altitude and velocity. From these outputs, the following features were computed:

- **Mean and Maximum Velocity (m/s):** From the velocity obtained directly by the GPS outputs, the mean velocity is calculated. The maximum velocity corresponds to the 95 percentile of the velocity to reduce the influence of outlying measurements.
- **Mean and Maximum altitude variation (m):** The altitude variation is extracted from the altitude GPS measurements and subsequently the mean and maximum altitude variations are calculated.
- **Walking Distance (m):** Walking distance is calculated by the sum of the euclidean distance between GPS coordinates.
- **Walking Time (min):** Walking time is computed by the difference between the last and the first timestamps of GPS measurements in minutes.

3.1.1.2 Dead Reckoning

DR is an inertial navigation estimation technology used to estimate the user's current position by using a previously determined position. To learn mobility patterns that can not be measured using GPS measurements, this technology was used to extract features from the step detection and its estimated length. The DR algorithm implemented in this framework was developed by Fraunhofer AICOS [16]. The output features from this algorithm were:

- **Number of steps:** The detection of steps is done using the accelerometer signal and a sum of the detected steps during the whole acquisition is computed.
- **Mean step length (m):** Once a step is detected, the step length is estimated based on Weinberg's method [16], and the mean step length considering the whole acquisition is computed.

3.1.1.3 Simple Activities

For the recognition of simple activities such as *walking*, *standing*, *walking up* and *walking down*, a machine learning classifier was implemented. Although a supervised machine learning classifier was used, this classifier was included in unsupervised feature extraction since it is user-independent. The recognition of human locomotion is essential for understanding the walking difficulty that can be characterised by having more or fewer altitude variations or even stops. The implementation of this classifier undergoes several stages that are commonly applied in activity recognition algorithms, described in section 2.3 and relies only on accelerometer and barometer sensors. Both sensors signals were resampled to 30 Hz. The choice of the resampling frequency relied on the study conducted by Figueira [97], that chooses 30 Hz as the ideal sampling rate to provide a low power consumption and good accuracy to monitor human movements. Since this framework focus is the evaluation of human behaviour that needs to be continuously monitored, it is

desirable that the developed algorithms allow the minimum power consumption, to save the smartphone's battery, and simultaneously, fulfil a good performance.

Besides the tri-axial raw data from the accelerometer, the acceleration magnitude was also calculated using the following Equation 3.2.

$$magnitude = \sqrt{x^2 + y^2 + z^2} \tag{3.2}$$

Where x, y, z are the acceleration in each direction.

Afterwards, acceleration and pressure data are segmented into equal-sized 5 seconds windows without overlap, from which are extracted a total of 17 features from the temporal, spectral and statistical domains, included on TSFEL. After feature extraction, a normalisation using Equation 2.1 was applied. Regarding the feature selection algorithm, a feed-forward feature selection from wrapping methods [98] was used with a 10-fold cross-validation method.

Regarding the extracted features using this locomotion classifier, its predictions are used to calculate the percentage of time each activity is being performed. This percentage time is calculated by Equation 3.3.

$$t_{activity}(\%) = \frac{\Delta t_{activity}}{\Delta t_{route}} \times 100 \tag{3.3}$$

Where $\Delta t_{activity}$ represents the activity duration along all route and Δt_{route} corresponds to the route duration. Thus, the output features are:

- **Walking (%):** Different walking speeds were considered for training.
- **Standing (%):** This activity includes both sitting and standing activities.
- **Walking Up (%):** Walking through stairs (up) and also ramps with high elevation are included in this activity.
- **Walking Down (%):** Walking down includes walking through stairs (down) and also very steep descents.

3.1.2 Supervised Feature Extraction

In this section, features from location and activities are extracted in a supervised way, which means, that there is previous knowledge about the user being studied that can be used as input to the algorithms for extracting more personalised information about the routine performed.

3.1.2.1 Indoor Location

For the human behaviour trajectory evaluation in the indoor environment, a technology for indoor positioning is required. As the main solutions for indoor positioning

need installation of equipment or depend on fingerprinting processes, that can be time-consuming, for this study a room-level indoor location solution with a fast deployment was chosen. This solution relies on Wi-Fi RSSI measurements to recognise in which room the user is located. The algorithm used was developed by Fraunhofer and its training process is as simple as starting a data recording in each room separately. Using the unique IDs from AP and the corresponding signal strength, a statistical classifier is trained and the prediction step is based on the highest probability.

The output of the algorithm includes the labels of the predicted rooms over time. Thus, it is possible to extract some relevant features for finding patterns in human behaviour, namely:

- **TOI (min):** Given by the time spent in each room.
- **Number of entries:** The number of times the user goes into each room.

3.1.2.2 Complex Activities

The recognition of complex activities involves a deeper knowledge about the user being studied.

For this reason, depending on the user and also on the characteristics of the routine being analysed, a personalised training process is required. For this training process, a set of activities is selected, and the user must perform each activity several times beforehand. Alternatively, during his/her routine the annotation of activities can be done and used for training after a few days.

For the recognition of the complex activities, data from the accelerometer, gyroscope, magnetometer, barometer and microphone smartphone sensors are acquired. Similar to simple activities, data is resampled to 30 Hz and the magnitude of tri-axial sensors is calculated. A resample exception is applied to microphone since a sampling frequency of 8000 Hz is needed to detect small sound variations.

The complex activities classifier follows machine learning steps described in section 2.3. Additionally to TSFEL features, the ratio between the range of 0.6 Hz and 2.5 Hz frequencies with all frequency bands was added. This range of frequency was chosen because it is the range of frequencies of human movements [99]. This feature may be interesting for the recognition of these activities since the smartphone is placed on the wrist, thus it will be even more sensitive to human movements.

Independently of the train classifier, features that result from the prediction of each activity are extracted, namely:

- **Duration of activity$_i$ (min):** Time that activity i takes to be performed.
- **Number of activity$_i$:** Number of times that activity i is performed.

These features are useful for understanding the user's pattern in the execution of daily activities and may provide information about the user's cognitive behaviour. A long duration may indicate the difficulty of the user in executing a given activity. The missing activities from the user routine may also suggest the inability to perform them or the lack of interest in some activities that used to be part of the routine.

3.1.2.3 Hidden Markov Models

According to the literature, HMM is an effective method for finding patterns [25, 100, 101, 102]. This method infers about hidden states given a time series of observations and returns the log-likelihood of each time series of observations. This framework uses HMM to infer about the probability of a given sequence of observations including sequences of locations and activities performed by humans, that arise from the output of indoor location method described in subsection 3.1.2.1 and complex activity recognition explained in subsection 3.1.2.2.

Since the input is a discrete sequence of observations, a multinomial HMM was trained and tested recurring to hmmlearn library[4]. The number of hidden states (k) is a parameter that needs to be defined to implement HMM. The search for the optimal number of hidden states uses the Bayesian Inference Criterion (BIC), well-described in [103], that combines a Maximised Likelihood (ML) and a penalty term. The basis of BIC is that a model is penalised according to the number of parameters it contains, which means that more parameters lead to a complex model, increasing the likelihood and consequently the penalty term. The optimal number of states is given by:

$$k_{BIC} = \text{argmax}\, BIC(k) \tag{3.4}$$

Where,

$$BIC(k) = logP(O|\lambda_k) - \frac{N_k}{2} log(n) \tag{3.5}$$

The term $logP(O|\lambda_k)$ is the log-likelihood estimate of the model with k states, N_k is the number of free parameters, given by Equation 3.6, and n the number of observations.

$$N_k = [k \times (k-1)] - Total\ number\ of\ zeros\ in\ transition\ matrix \tag{3.6}$$

Thus, a set of numbers of hidden states was tested, in each iteration the BIC value is stored and the chosen k is the one that maximises BIC (see Equation 3.4).

Recurring to HMM the following features were extracted:

- **Activity sequence Log-likelihood:** Log-likelihood of each activity sequence performed.
- **Location sequence Log-likelihood:** Log-likelihood of each location sequence performed.

[4]Available in https://github.com/hmmlearn/hmmlearn (visited on 03/09/2019)

3.1.2.4 Activity search

The activity search is an alternative for the recognition of an activity without implementing a machine learning classifier. Supposing that a specific activity needs to be monitored in a person's daily routine, for instance, taking the medication activity, it is possible to apply the activity search algorithm for this recognition. Powering the training process of a machine learning classifier this algorithm relies on the application S-DTW for the activity search.

S-DTW is a modification of the classical DTW described in subsection 2.4.3. It is a useful method for comparing two sequences X and Y with too distinctive lengths. Considering that the length of X is much larger than the length of Y, the goal here is to find a subsequence in X that corresponds to the minimum DTW distance between all subsequences computed in X and sequence Y.

S-DTW is effective in finding subsequences in time series, however, when the sequence that is being searched has high variability, the subsequence with the minimum DTW distance may not be the correct one. To overcome this difficulty, a modification to S-DTW was developed to select the subsequence using a different criteria. The standard S-DTW only returns one subsequence, that corresponds to the minimum DTW distance. It is noticeable that a small variation in the subsequence that we are searching for, can lead to a high cost in the accumulated cost matrix, and this variability is frequent, since activities are not always performed in the same way. Thus, a criteria based on the minimum median value of the cost matrix path was implemented. The median value is more suitable in this case because it is less sensitive to noise. Furthermore, to avoid choosing the wrong subsequence due to the signal size, an additional condition was imposed (see Equation 3.7):

$$0.5 \times \Delta t_{ref} \leq \Delta t_{query} \leq 1.5 \times \Delta t_{ref} \tag{3.7}$$

Where Δt_{ref} is the duration of the sequence that is being searched and Δt_{query} is the duration of the returned subsequence. This way, it is more likely to find the correct subsequence.

The output of the activity search is a Boolean value with the following extracted feature:

- **Activity x:** Boolean value for activity x, where 1 indicates that the activity was found and 0 the activity was not found.

3.2 Pattern Discovery

The pattern discovery step plays a key role for modelling patterns in human behaviour. Using the notation from Figure 3.1, the pattern discovery can be described by Equation 3.8. Where f_{day_i} represents the obtained extracted features from Equation 3.1 of day i,

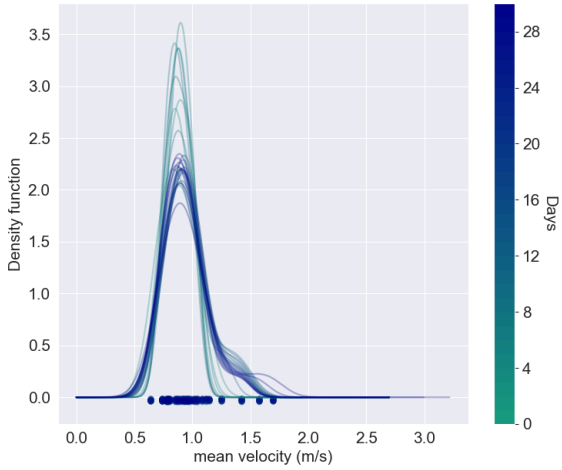


Figure 3.3: Example of a feature modelled through a KDE. The corresponding values are assigned with the points bellow.

$f_{pattern}$ is a function composed by models to learn motion patterns and $Pattern$ is the learned patterns using n days.

$$Pattern = f_{pattern}(f_{day_1}, f_{day_2}, ..., f_{day_n}) \tag{3.8}$$

Hereafter, two methods for learning patterns using the extracted features from unsupervised and supervised feature extraction, are proposed, namely: probability density functions and clustering approaches.

3.2.1 Kernel Density Estimation

The first approach for learning human patterns is to model the previously extracted features into a probability density function. As the probability density function of the extracted features is unknown, a KDE was used to model each extracted feature in section 3.1.

This method was designed to be independent of the feature being used and all features from the Feature Extractor step were equally modelled with KDE. Thus, the process of adding more features to model a specific routine can be easily introduced in this framework without changing the pattern discovery method. Depending on the intrinsic characteristics of each feature for a specific user, the modelling process may need more or fewer days to learn the feature pattern. Naturally, the pattern will become more robust with the addition of more days. In Figure 3.3 an evolution of the KDE for mean velocity through the number of days is presented. It is possible to verify that after approximately 16 days the KDE stabilise and adding more days do not make significant differences to the distributions.

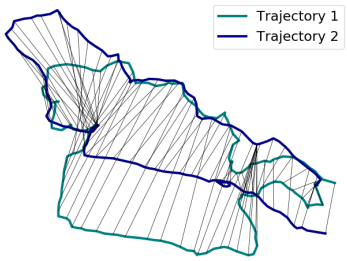


Figure 3.4: Trajectory similarity representation using DTW distance.

3.2.2 Clustering

Although the majority of features can be modelled using KDE, other patterns can be learned using spatial information. For example, we can learn that a specific activity occurs always in the same geographic location. For this purpose, clustering methods are adequate for finding patterns in an unsupervised way, relying on data similarity. Hereafter, cluster analysis is applied in trajectories, POI and sensory signatures.

3.2.2.1 Trajectory

Trajectory similarity is measured using DTW, it is a robust distance measure for calculating spatial similarity since it can warp trajectories non linearly and have a good performance when measuring the similarity of trajectories with different lengths and sampling rates. This is ideal for dealing with human trajectories, as humans travel with different speeds and do not always walk the same distance. Hence, the Pierre-Rouanet DTW algorithm [5] is adapted for the calculation of DTW in two dimensional data, as GPS trajectories are two-dimensional. The detailed algorithm is shown in Algorithm 2, and an illustration of this distance feature which was applied to two outdoor trajectories is presented in 3.4.

Algorithm 2 DTW in two dimensional trajectories

1: **INPUT:** Two trajectories P and Q in meters
2: **OUTPUT:** $distance_{P\_Q} \Leftarrow$ DTW distance between P and Q in meters
3: $euclidean\_norm = \sqrt{(p_x - q_x)^2 + (p_y - q_y)^2} \Leftarrow$ where p_x, p_y, q_x, q_y are the points of trajectories P and Q in x, y axis respectively.
4: $distance_{P\_Q} = dtw(P, Q, norm = euclidean\_norm)$

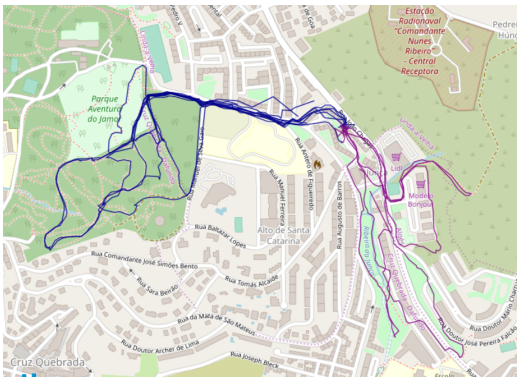
Once trajectory distance is measured, it is required an automatic method that groups trajectories according to their spatial similarity. Since the final number of clusters is unknown, DBSCAN and HDBSCAN are two suitable clustering methods. For this study,

[5] Available in https://github.com/pierre-rouanet/dtw (visited on 03/09/2019)

HDBSCAN, from hdbscan clustering library[6], was chosen as it automatically finds the optimal *Eps* value and is capable of dealing with clusters of different densities.

The input of HDBSCAN algorithm is a precomputed distance matrix using DTW in two-dimensional data. Additionally, the only required parameter is the minimum cluster size that was set to three, which means, that one group needs to contain at least three samples to be considered a cluster.

An example of this clustering process using an outdoor routine of 11 days is presented in Figure 3.5a, where 2 clusters were obtained.



(a) Representation of two trajectory clusters in blue and purple using a routine of 11 days.

(b) Representation of one POI in blue for a routine of 11 days.

3.2.2.2 Points of interest

POI are locations of interest, in other words, it is where the user stays for a specific time in different routes. In this study, a POI is defined as the location where the person stands for the minimum time of 1 minute and occurs in at least 3 different days. An algorithm based on the prediction of the simple activities classifier was developed (see section 3.1.1.3), for the assessment of these locations. This algorithm uses the standing predictions of the classifier combined with time and spatial information, to detect the locations where the user stayed at least one minute. A POI is also defined within a radius of 50 meters. As the final number of clusters is unknown and minimum radius for considering the points of the same cluster (i.e. 50 meters) is well-known, it was implemented the DBSCAN clustering method, for the POI search. The DBSCAN input is the pre-computed distance matrix between coordinate points using the euclidean distance between coordinate points. Moreover, the required parameters of DBSCAN were empirically tested obtained 50 for *Eps* and 3 for *minpts*. An example is shown in Figure 3.5b, where one POI was identified in a routine of 11 days.

3.2.2.3 Sensory signatures

Humans have predictable behaviours that can be translated into sensor signatures. By assimilating sensor data from routines, a practical scheme that employs unsupervised

[6]Available in https://hdbscan.readthedocs.io/en/latest/index.html (visited on 07/09/2019)

learning to extract unique sensor signatures from human routine was done. The idea of sensory signatures was to find motion patterns in a more abstract level. Thus, sensory signatures that were repeated in more than one day, within human routines, were searched. This is a non-obvious and complex question that leads to the implementation of several approaches, it was tested on the accelerometer signal as it provides significant information about human movements. Several attempts using different methods, ranging from signal entropy analysis, S-DTW, deep learning, among others, were performed. Although the preliminary results were promising, the obtained results turn out to be unsatisfactory to make reliable conclusions about these methods.

In order to simplify this search, as the signals regions with more entropy were also characterised by having a high acceleration, a search for regions with high acceleration (higher than 11 m/s) was done. Subsequently, the found regions were mapped through their corresponding GPS coordinates. This way, if a signal with high acceleration was found in the same location, it was more likely referred to the same activity. Similarly to 3.2.2.2, the GPS coordinates of the signals regions with acceleration higher than 11 m/s^2 were extracted from all days. Then, DBSCAN clustering method was applied with parameters Eps and $minpts$ equal to 50 and 3 respectively. This means that clusters contain at least 3 locations, considering different routes, whose signal has an acceleration higher than 11 m/s^2 within a radius of 50 meters.

3.3 Anomaly Detection

Once motion patterns were defined, the next step of this framework aimed to detect anomalies on those patterns. Using the notation from Figure 3.1, the anomaly detection can be described by Equation 3.9, where $Pattern$ is the learned patterns from section 3.2, f_{day_i} represents the extracted features for day_i, $f_{distance}$ is a method based on a distance measure to detect deviations from the $Pattern$ and Δ_{day_i} is the distance value.

$$\Delta_{day_i} = f_{distance}(Pattern, f_{day_i}) \tag{3.9}$$

To describe the anomaly detection a detailed fluxogram is present in Figure 3.6. This figure summarises both human behaviour learning and anomaly detection. First, features were extracted from each day and the behaviour pattern is only defined after a predefined number of days (day_{learn}). When day_{learn} was reached, the pattern was learned and the threshold to detect an anomaly was defined. The following days were evaluated by measuring the distance of each day to the pattern. To predict an anomaly more robust, the anomalous decision was done through the evaluation of a predefined number of consecutive days (day_{anom}). Therefore, only if the mean distance along the day_{anom} days was above a threshold an anomaly was detected.

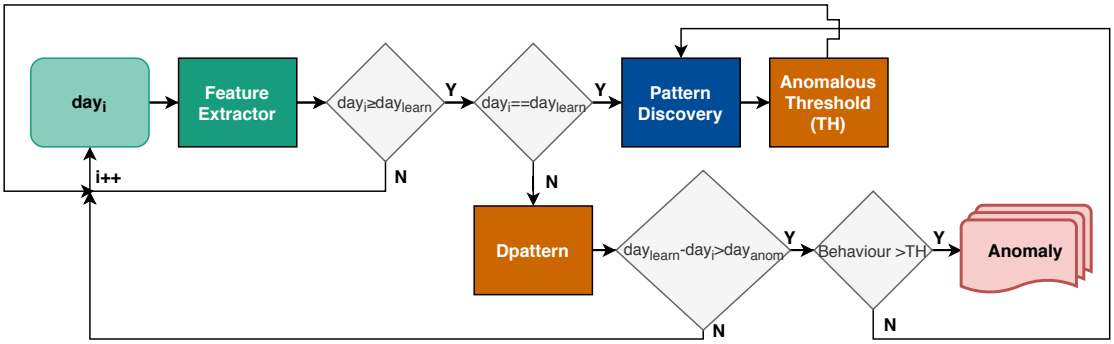


Figure 3.6: Learning patterns continuously and anomaly detection fluxogram.

The distance to the pattern and the threshold for detecting an anomaly were defined according to the pattern being evaluated.

The threshold was initially defined considering the learned behaviour regarding day_{learn}. Once the threshold was defined, the behaviour distance accounting the following day_{anom} days was computed and if this distance was lower than the threshold, then the threshold was updated considering the new behaviour and the process continues, otherwise an anomaly was detected and the pattern model would not learn the day covered on day_{anom} days with a higher distance value.

Since patterns were defined as probability density functions and clustering, the detection of anomalies will be described considering those two methods.

3.3.1 Anomalies in distributions

Assuming that a specific feature is well modelled through KDE, we can use the density value from KDE to assess the feature probability given a predefined feature value.

To transform density values to distance values, we normalised each distribution by its maximum value. Therefore, the KDE distance is a scale from 0 to 1 and we can assess the level of deviation from pattern through a quantitative measure.

In Figure 3.7 a KDE distribution modelling a specific feature is represented. For evaluating the anomalous level of a new measure, the intersection between the distribution and the new measure was obtained. For example, the green line corresponds to a likely measure according to that distribution, since the interception with the distribution returns a value above the anomalous threshold. The distance to the pattern is given by $1 - KDE_{distance}$, thus in this example, a value of 0.12 is obtained. In contrast, the red line refers to an unlikely measure with a distance to the pattern of 0.88 which is considered as an anomalous value.

This method provides a quantitative measure of the anomaly considering the distribution pattern and it is useful for evaluating the anomaly level of a specific user's feature.

To combine several features and return an overall level of anomaly a global distance

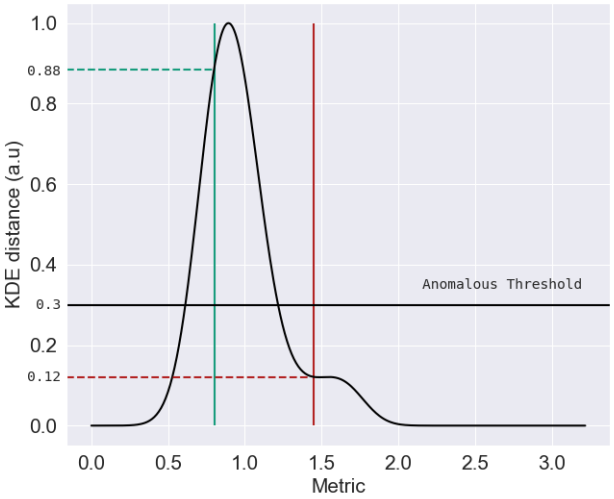


Figure 3.7: KDE distribution, where the x-axis is the evaluated feature and y is the KDE distance. The green line is a likely feature value, the red line in an unlikely feature value and the black horizontal line corresponds to the anomalous threshold.

($Dpattern$) was computed by a weighted arithmetic mean where depending on the importance of each feature, different weights can be assigned. $Dpattern$ was computed by Equation 3.10, where d_{fi} is the distance to the pattern of each feature i in a total of n features and w_{fi} are the corresponding weights, thus we can define a weight for each feature to measure the anomalous behaviour.

$$D_{pattern} = \frac{\sum_{i=0}^{n} d_{fi} \times w_{fi}}{\sum_{i=0}^{n} w_{fi}} \tag{3.10}$$

The current $Behaviour$ (Figure 3.6) is given by:

$$Behaviour_j = \frac{\sum_{x=j}^{j+day_{anom}} D_{pattern_x}}{day_{anom}} \tag{3.11}$$

The anomalous threshold was defined by Equation 3.12, where all behaviours correspond to normal behaviour only.

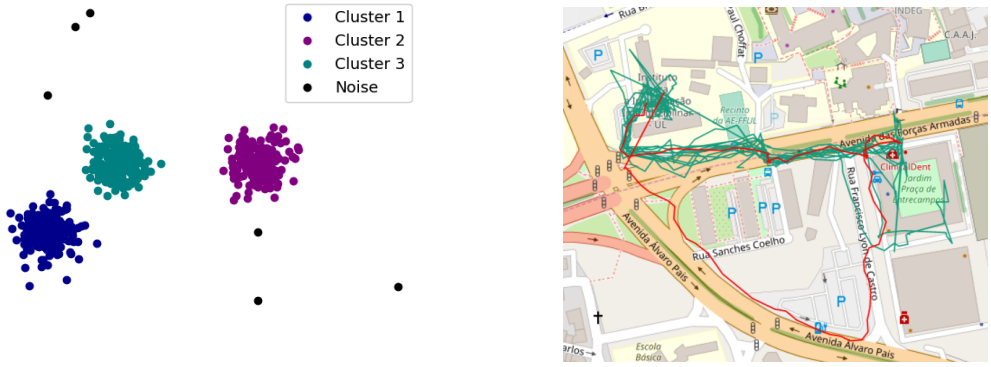
$$TH = 1.1 \times max(Behaviour_0, Behaviour_1, ..., Behaviour_m) \tag{3.12}$$

By defining the threshold as more 10% of the maximum behaviour distance, only the distances to behaviour that are higher than 10% of the maximum behaviour distance of the trained model would account as anomalies, thus the model becomes less sensitive to false positives.

3.3.2 Anomalies in clustering

This framework comprises clustering methods that are sensitive to noise, namely DBSCAN and HDBSCAN. Hence, anomalies in clustering were identified as the resulting noise data of clustering methods, as shown in Figure 3.8a. Additionally, the search for anomalies within a cluster was also done using distance metrics. Trajectory clusters were separated through spatial similarity, therefore, within a cluster, it is important to detect trajectories that are distinguishable, such as a person getting lost while trying to reach a specific location. Figure 3.8b shows an example of a trajectory cluster, where the red trajectory is visually different from the remaining. For the detection of these anomalies, the output of the clustering method combined with a threshold-based approach was used to detect an anomaly if the maximum distance of the new trajectory with all trajectories inside the cluster was higher than a threshold. In this clustering method, $Dpattern$ is defined as the DTW distance between all trajectories within a cluster as in Equation 3.13, being Tr a trajectory of a set of N trajectories.

$$Dpattern = DTW(Tr_i, Tr_j) \; for \; i \; and \; j \; \in [0, N] \tag{3.13}$$



(a) DBSCAN clustering results

(b) HDBSCAN trajectories results.

Figure 3.8: Clustering approaches illustration. Figure 3.8a represents the DBSCAN result, finding three clusters and noise points represented in green, blue, purple and black, respectively. Figure 3.8b illustrates an anomalous trajectory coloured in red within a cluster of similar spatial trajectories coloured in blue.

Chapter 4

Results and Discussion

This chapter comprises the results of the framework for learning human patterns behaviour tested on three different datasets involving the daily living of two different users. First, a description of signal acquisition and datasets is presented followed by the results obtained by each dataset, regarding the extracted features, motion patterns and anomaly detection according to case studies.

4.1 Signal Acquisition

The signal acquisition was done using the Recorder mobile application developed by Fraunhofer AICOS. The application extracts the data perceived not only by smartphone sensors but also by external sensors connected to the smartphone. The application allows selecting the sensors from which data should be recorded, as it can be seen in the first image of Figure 4.1. In this study, the accelerometer, gyroscope, magnetometer, barometer, sound, Wi-Fi and location sensors were recorded. The Recorder application stores the data according to a predefined folder structure. Each user has an identification folder, which contains multiple folders organised by each day of acquisition, named with the corresponding date. The acquisition folder contain text files, properly identified by sensors names, with the sensing information acquired during the acquisition.

During the data recording, the user was asked to behave as naturally as possible, for extracting the user's behaviour most realistically, and to perform his/her routine without any recording interruptions. After the acquisition is finished, data is automatically saved in the smartphone.

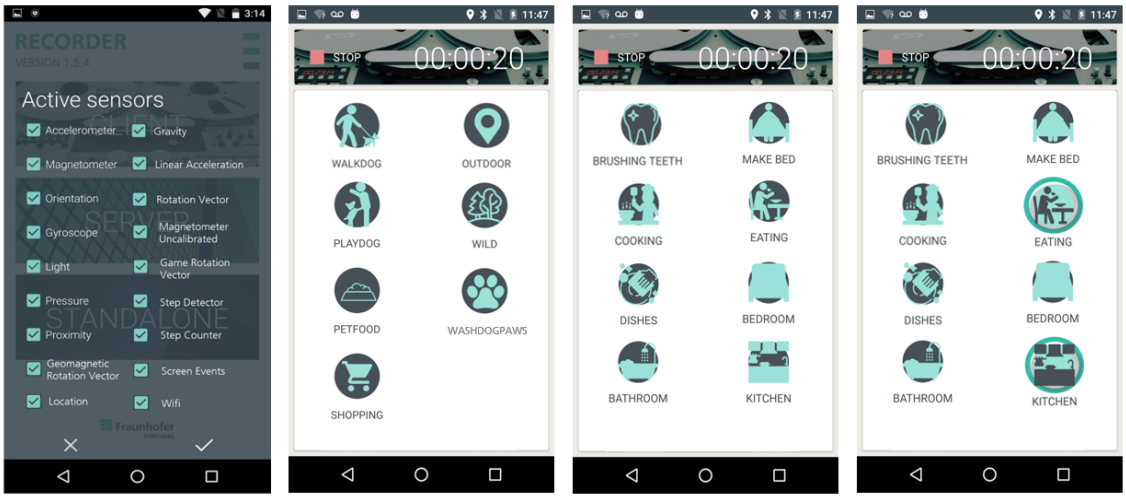


Figure 4.1: Recorder application mobile layout. From left to the right, the first image corresponds to the layout for activating the sensors, the second and third images present customised layouts for the user's routine, and the last image illustrates the annotation of eating activity and kitchen location.

4.1.1 Annotation Procedure

Once the data recording starts, different layouts appear on the screen according to the user's routine (see Figure 4.1). These layouts were designed for the scope of this study, and customised for each user, according to his/her routine, to provide an attractive, and easy annotation during the acquisition.

For validation purposes, the user was asked to annotate the start and end of each complex activity performed, and every time the user entered or left a specific location. To stamp the beginning of an activity/location, the user only needed to click on the corresponding button and a green light colour will indicate the activity/location that was taking place (see the fourth image of Figure 4.1). When the user stopped performing an activity or left a specific location, the button should be clicked again for stamping the end of the activity/location.

4.1.2 Sensor Placement

Since the data recording of human routines is an exhaustive and long procedure, the sensor placement was carefully chosen to minimise the user's discomfort. Furthermore, sensor placement should be sensitive enough to perceive human motions in order to recognise predefined complex activities that need to be monitored. Hence, it was decided to place the smartphone on the user's wrist of the dominant hand, hold by a band support as shown in Figure 4.2. In real life application, the smartphone should be replaced by a bracelet or smartwatch comprising the needed sensors to perceive human motion behaviour.

Figure 4.2: Sensors placement illustration with Recorder mobile application on going.

4.1.3 Ethical and privacy considerations

Smartphone sensors allow to record data of naturalistic behaviour, objectively, continuously and in an unobtrusive way. In this study, the collected data, using a smartphone, was processed and analysed to learn human pattern behaviour by the recognition of activities and trajectory of the user at study. The collected information may lead to the identification of the user, so it is important to consider ethics and to safeguard user data privacy.

The following ethical and privacy considerations were considered relevant based on Harari et al. [104] study. The study participants should consent to enrol voluntarily in this study, and they should be duly informed about the nature of the collected data. They also should authorised the use of the sensing application on his/her smartphone. The participants should be informed in a clear and perceptible manner on how the application works and on the data storage practices being implemented. The goals of the study must be explicit and the user should have the opportunity to receive a copy of the acquired data. Participants data should be discussed carefully so that privacy is respected. At any time, the user has the right of deleting or removing their personal data and withdraw from the study. According to Boase [105] data must be stored in a safe place to avoid the access of unauthorised people, avoiding the exposure of participants information. Data should also be anonymous and removed all identifying information.

A copy of the informed consent filled by the participants of this study is presented in Appendix B.

4.2 Human mobility on daily walks dataset

This dataset was recorded by User 1 to extract mobility patterns during the daily walks. The User 1 dataset comprises the outdoor daily walks of the user with his/her dog. During the walks the user frequently performed the following complex activities: *washing the dog paws*, *feeding the dog* and *playing with the dog*. These activities may also be characteristic

Figure 4.3: *Playing with the dog*, *washing the dog paws* and *feeding the dog* activities acquisition time in seconds with the corresponding repetitions inside the brackets.

of the user's mobility, so it is important to recognise them. Hence, the user was asked to perform a set of complex activities repetitions separately from the routine, to implement a classifier for their automatic recognition on the routines. The activities acquisition time is presented in Figure 4.3. Thus for validation purposes, the user also had to annotate these activities during the daily walks.

Despite the normal routine days acquired in this dataset, it also comprises days with planned anomalies reflecting a human that starts to express reduced mobility. Reduced mobility is a serious disability concerning elderly people, associated with falls that may have critical consequences. The planned anomalies were directly related to mobility, thus, User 1 during the anomalous days started to walk slowly and to perform shorter trajectories in less challenging paths.

The data recording comprises a total of 61 days of acquisition, 49 normal days and 12 anomalous days, over a period of 6 months. The dataset overall information is summarised in Table 4.1. A set of the recorded days faced some issues during the acquisitions, such as missing sensor data file or inaccuracies on GPS data points. Both issues limited the extraction of some relevant features. Less frequently, wrong and missing annotations also occurred. Despite some excluded features, all acquisition days were used for pattern discovery. However, for anomaly detection only 38 days, 31 normal and 7 anomalous, were reliable for the detection of anomalies, since only those contained all the information needed for the anomaly behaviour detection procedure.

4.2.1 Feature Extractor

Feature extractor contains the results from unsupervised and supervised feature extraction regarding the performance of the implemented machine learning classifiers, routine activity sequences and activity search algorithm.

4.2.1.1 Locomotion classifier

To recognise basic activities on daily walks such as *standing*, *walking*, *walking up* and *walking down*, five different machine learning classifiers were trained. As stated in subsection 3.1.1.3, only accelerometer and barometer sensors were used for this recognition. A total of 17 features were extracted, namely mean values, standard deviations, maximum frequency magnitudes and maximum peaks from the magnitude and tri-axial accelerometer signals, and linear regression from the barometer. A feed-forward feature selection was

Table 4.1: Dataset information regarding the user ID, activities performed, activated sensors, acquisition device, device position and recording days and hours.

Dataset Information		
User ID	1	
Activities	Simple	Complex
	Standing, Walking, Walking up, Walking down	Feeding the dog, Washing the dog paws, Playing with the dog
Sensors	Accelerometer (200Hz), Gyroscope (200 Hz), Magnetometer (100 Hz), Barometer (5 Hz), Sound (8000 Hz) , Location (1 Hz), Wi-Fi (0.3 Hz)	
Acquisition Device	Samsung Galaxy S6	
Device Position	Dominant Wrist	
Recording Days	Normal	Anomalous
	49	12
Recording Hours	Normal	Anomalous
	~30 hours	~3 hours

Table 4.2: Accuracy of machine learning classifiers classifiers for simple activities recognition (*walking*, *standing*, *walking up* and *walking down*) using a test set of 30% of the input samples. The classifier with highest accuracy (DT) is highlighted in bold.

Classifier	Accuracy (%)
DT	**90.2%**
KNN	89.0%
RF	87.8%
NB	84.1%
ADA	63.0%

applied using 10-fold cross-validation, and the train and test sets were randomly split using a ratio of 70% / 30% of the input samples. The obtained results of the 5 classifiers are presented in Table 4.2, with the accuracy of the high-performance classifier, highlighted in bold. The corresponding normalised confusion matrix is represented in Figure 4.4.

The DT classifier ended up with an accuracy of 90.2%, using the following selected features: Standard deviation of acceleration magnitude, barometer linear regression, mean y-axis acceleration, the total number of peaks of x-axis acceleration and standard deviation of y-axis acceleration. By the analysis of the confusion matrix, the *standing* class is well recognised. However, *walking*, *walking up* and *walking down* classes present some confusion, which is expectable since all activities rely almost on the same movements. One of the main differences between these activities can be assessed by the pressure that tends to vary when *walking up* or *walking down*, and this classification scenario is given by the barometer linear regression feature.

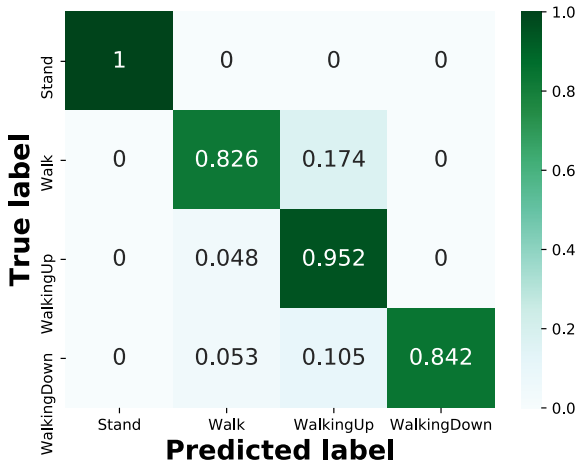


Figure 4.4: Normalised confusion matrix of locomotion recognition activities using DT classifier.

4.2.1.2 Routine activities classifier

Even though locomotion activities are meant to evaluate human's mobility, more complex activities that may be characteristic from the human's routine being studied, may also bring relevant information for detecting reduced mobility. For instance, the detection that a specific activity was no longer performed. For this purpose, three daily complex activities performed in the human routine at study were selected, namely *washing the dog paws*, *feeding the dog* and *playing with the dog*.

In order to train a classifier to recognise these 3 activities, the user was asked to perform the activities at least 6 times, while data recording. After this process, features from the segmented windows (5 seconds with 50% overlap) were extracted, and the best features selected through feed-forward feature selection method. TSFEL features were not all chosen for this recognition problem since it would increase the time complexity of the forward selection algorithm. From the sound, only Mel Frequency Cepstral Coefficients (MFCC) values and spectral features were extracted. From barometer, only features that did not rely on the absolute barometer value were extracted, since this value depends on the atmospheric pressure that varies along the days. From the accelerometer, magnetometer and gyroscope, spectral, statistical and temporal features were extracted from the three-axes and magnitude windows. Thus, a total of 128 features were extracted and used as input to a feed-forward feature selection algorithm. The validation process was a 10-fold cross-validation and all user activity repetitions were used to train the classifiers. The test set was composed by the activities performed and annotated during each day of the user routine. The obtained accuracy and F1-score using 5 different classifiers is presented in Table 4.3.

Afterwards, a post-processing was applied to the prediction result, correcting predictions by majority voting. Consider the segmented prediction labels, into windows, by a step of one. The first label of each window is assigned to the majority prediction of that

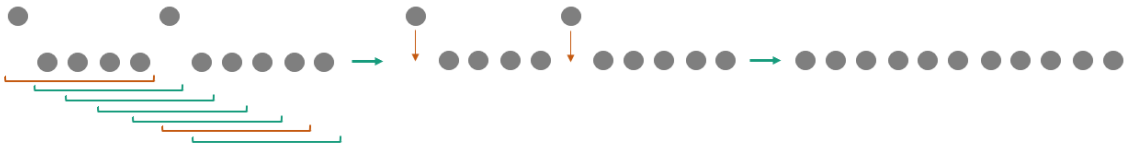


Figure 4.5: The grey circles represent predictions with the boundaries of the segmented predictions bellow. The orange boundaries represent the windows that contain a prediction that is reassigned due to the majority voting of the corresponding window. The process is analysed from the left to the right.

Table 4.3: Results of supervised learning classifiers tested in all days of user routine in terms of accuracy and F1-score before and after post-processing. The best results are in bold.

Classifier	Accuracy (%) Before PP	F1-Score (%) Before PP	Accuracy (%) After PP	F1-Score (%) After PP
DT	44.1%	40.4%	46.0%	43.4%
KNN	81.6%	82.1%	**94.0%**	**94.1%**
RF	70.4%	71.0%	83.1%	84.0%
NB	82.3%	82.1%	92.4%	92.2%
ADA	73.6%	74.0%	86.4%	86.7%

window, see an example in Figure 4.5. The application of this post-processing resulted in an accuracy improvement for all tested classifiers, as it can be seen in Table 4.3.

The classifier that achieved the highest accuracy and F1-score after post-processing was KNN with K equals to 5 and using the following selected features: Standard deviation of the acceleration magnitude (*Acc Mag std*), dominant frequency of y-axis magnetometer value (*Mag y dominant frequency*), sound MFCC 0 (*Sound MFCC 0*), mean y-axis acceleration (*Acc y Mean*), mean acceleration magnitude (*Acc mag Mean*), mean x-axis gyroscope value (*Gyro x Mean*), ratio between the range of frequencies between 0.6 and 2.5Hz and the all frequency band of the acceleration magnitude (*Acc Mag 0.6_2.5 and all band*), standard deviation of gyroscope magnitude (*Gyro Mag std*) and the ratio between the range of frequencies between 0.6Hz and 2.5Hz and the all frequency band of z-axis magnetometer value (*Mag z 0.6_2.5 and all band*). The horizon plot in Figure 4.6, illustrates the behaviour of the selected features on the three classified activities. The contribution of each feature can be assessed by the interpretation of horizon plot (Figure 4.6), where we can visualise some distinct characteristics, for instance, the *Mag y dominant frequency* is clearly different between *washing the dog paws* activity and the remaining activities since it presents two great positive peak values. Moreover, *play with the dog* activity can be distinguished from the remaining activities by the *Acc y Mean*, as it contains a great positive value. *Feeding the dog* is explicitly enhanced through its great positive *Gyro x Mean*. The combination of all these features leads to a good discrimination between activities, resulting in an overall accuracy of 94%. The normalised confusion matrix of KNN after post-processing is shown in 4.7.

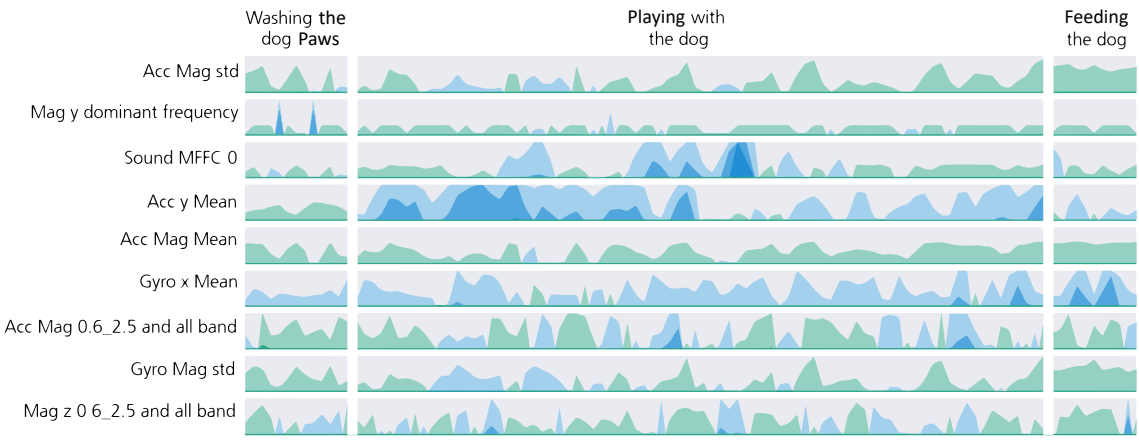


Figure 4.6: Horizon plot with selected features represented from top to the bottom. The x-axis contains the three activities, namely *washing the dog paws*, *playing with the dog* and *feeding the dog*. The range of y-axis corresponds to a third of the maximum value of each feature corresponding to the darkest values. The blue and green colours correspond to positive and negative values, respectively.

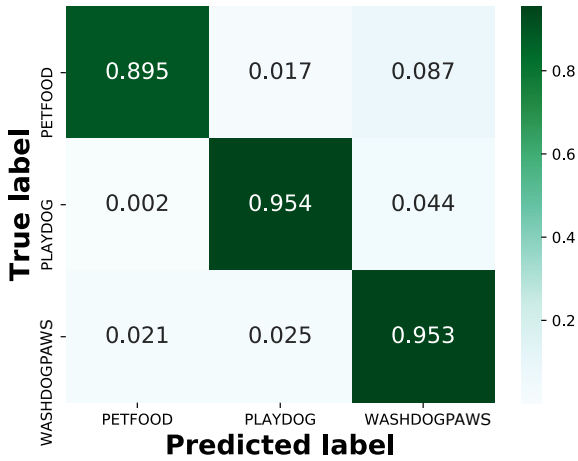


Figure 4.7: Normalised confusion matrix for routine activities classification, using KNN classifier, after post-processing.

Although the obtained results were satisfactory, it is important to note that in an online recognition process an activity spotting process is required. As activity spotting is a complex problem [106] and the attempt approaches, such as the creation of a rejection class or a None class, did not achieve successful results, this topic is identified as future work since the main goal of this study is the discovery of motion patterns.

4.2.1.3 Routine activities sequence

Based on the routine activity classifier, HMM are used to infer about the hidden states of a given sequence of observations, specifically a sequence of performed activities by the user. In this dataset, the user performs a maximum of three activities concerning *feeding the dog* (label 0), *playing with the dog* (label 1) and *washing the dog paws* (label 2). Here we

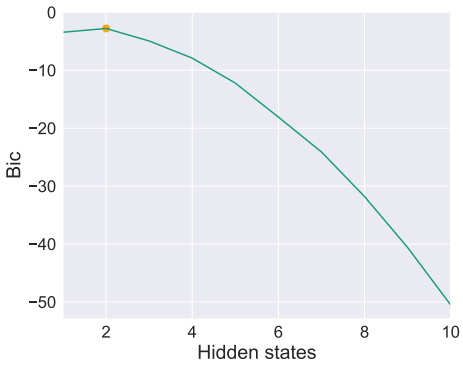


Figure 4.8: Representation of the BIC value by increasing the number of hidden states. The marker in orange corresponds to the number of hidden states that leads to a higher BIC value.

are interested not in the hidden states sequence but on the returned log-likelihood of the sequence of observations. For the implementation of HMM, it is necessary to define the number of hidden states in advance.

The number of hidden states was estimated through BIC, where the BIC value was calculated by testing the model using a sequence of observations with 10 different hidden states. According to the results of BIC shown in Figure 4.8, 2 hidden states are suitable for inferring about the likeliness of activity sequences, since it is the number of hidden states that leads to the largest BIC value.

Thus, a HMM was trained with all activity sequence from all days of the user routine, using both true (from annotations) and predicted activities from the trained classifier.

4.2.1.4 Activity search algorithm

The activity search algorithm was implemented to find a known activity from the user routine. As for the classification process of complex activities, a training step, where the user needs to acquire specific activities, several times, beforehand, is required, a simpler approach was developed. For this algorithm, the user only needs to perform the activity one time. For validation purposes, one repetition of *washing the dog paws* activity was selected. The search for this activity relies on the magnitude of the accelerometer signal as shown in the top of Figure 4.9. To reduce the computational cost, the signal is resampled to 30Hz, and a smoothing filter is applied to the resampled accelerometer magnitude signal, as it can be shown in the bottom signal of Figure 4.9.

The activity search algorithm, as explained in section 3.1.2.4, is based on a modification of S-DTW. The implemented algorithm, instead of finding only one sequence that ends on the minimum value of the last cost matrix line, returns several sequences corresponding to the ending points bellow the 10 percentile of the last cost matrix line. Using a routine day as an example, a total of 8 sequences were obtained (see Figure 4.10). As it can be seen from Figure 4.10, the subsequences 2, 3, 4, 6 and 7 have a duration of less than half of the search subsequence, being excluded by algorithm condition (see Equation 3.7).

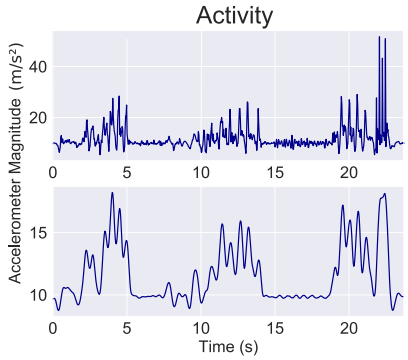


Figure 4.9: Illustration of the searched activity. The top signal is the raw accelerometer magnitude of the *washing the dog paws* activity, the bottom signal is the same signal after the application of a smoothing filter.

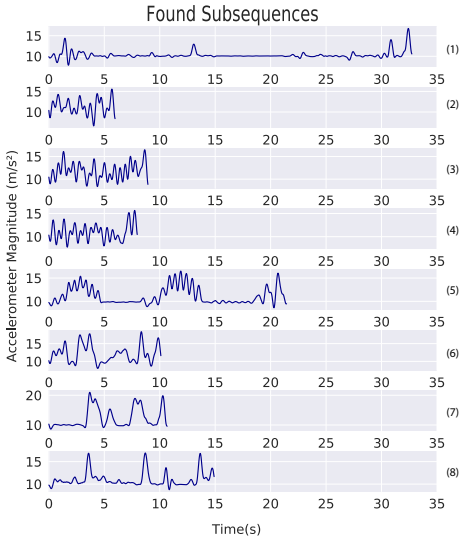


Figure 4.10: Illustration of the 8 found subsequences in the search for *washing the dog paws* activity.

Therefore, the remaining subsequences are compared by their median value of the cost matrix path, and the one with the lowest value is selected. In this example subsequence 5 is selected, that is the one that corresponds to the *washing dog paws* activity.

The activity search algorithm was tested in a total of 41 days and the results are represented in Table 4.4. As the selected activity is a common user's routine activity, the dataset is composed of much more days containing the activity than days without the activity. Although the obtained results seem to be unsatisfactory, it is important to note that regarding the 35 days containing the activity, in 31 days the correct activity was detected. Each acquisition day is composed of approximately 40 minutes from which approximately 5 subsequences are found and the correct activity is selected in 89% of the times. In this scenario, the classic S-DTW only correctly detects the activity in 20 days. On the other hand, the results for the 6 days without the activity were not satisfactory since only 1 day was a false negative. However, due to the annotation issues, it is possible

Table 4.4: Results of the activity search algorithm tested in 41 days. In 31 days the activity was correctly detected, in 4 days the activity existed but it was not detected, in 5 days there was no activity and a subsequence that did not match the searched activity was returned and in 1 day there was no activity and any activity was detected.

		Predicted	
		T	F
True	T	31	4
	F	5	1

that the 5 false-positive can actually contain the searched activity. Nevertheless, more data is necessary to test the algorithm robustness considering days without the selected activity and also create new S-DTW restrictions to avoid the false positive. Considering the classic S-DTW, its absence of restrictions does not allow to exclude a subsequence.

4.2.2 Pattern Discovery

The extracted features from trajectory and activity recognition can have significant information about the user's behaviour. Hence, in this section regarding the extracted features in all the recorded days, patterns including features modelled through KDE distribution and clustering methods were addressed.

The human mobility on daily walks dataset comprises a large outdoor component, from which almost all of the proposed features in section 3.1 were extracted. Features from activities, DR, outdoor environment and HMM log-likelihood for a set of activity sequences were modelled into a KDE distribution and clustering methods were addressed for finding similar trajectories, POI and sensory signatures.

4.2.2.1 Motion patterns using KDE

KDE was used to model features extracted from an unsupervised and supervised manner. In particular, duration of each recognised activity (Figure 4.11), locomotion time percentage (Figure 4.12), walking features (walking time, walking distance, number of steps, step length - Figure 4.13), velocity features (mean and maximum velocity - Figure 4.14), altitude features (mean and maximum altitude - Figure 4.14) and finally the HMM log-likelihood for a set of activity sequences (Figure 4.15).

Regarding the recognition of three complex activities, namely *washing dog paws*, *playing with the dog* and *feeding the dog*, Figure 4.11 shows the three activities duration distributions using the true and predicted labels. The activities duration was calculated based on consecutive predicted labels from the same activity. Individual predictions were discarded, since each predicted label corresponds to a window size of 5 seconds, and the complex activities have a duration longer than 5 seconds, as it can be confirmed by the true labels from Figure 4.11. This way, the values constituting the blue distributions were obtained.

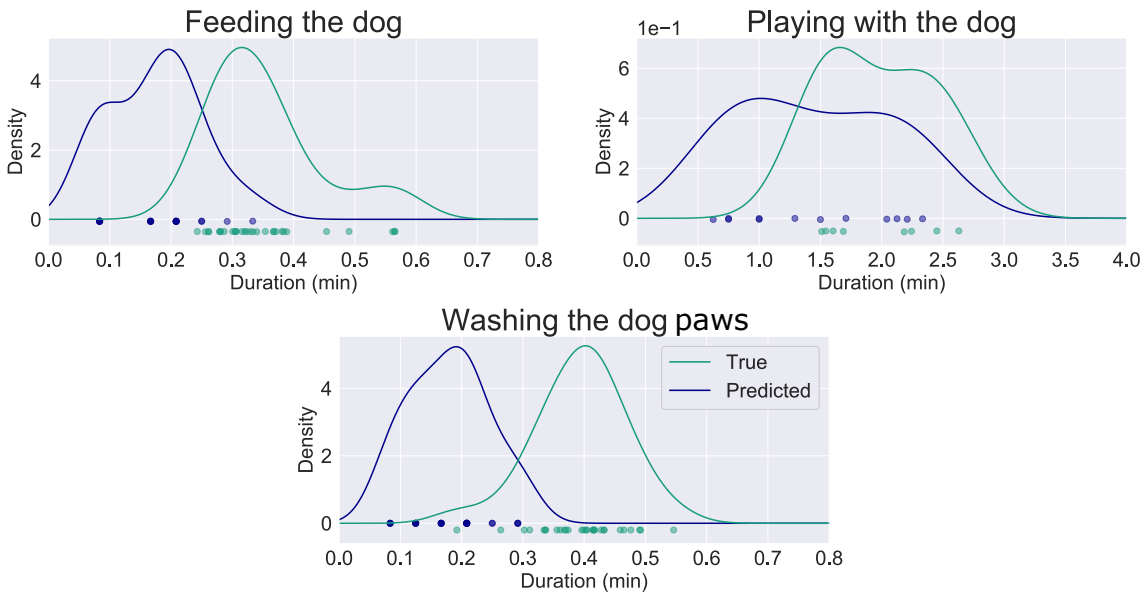


Figure 4.11: Representation of the *feeding the dog*, *playing with dog* and *washing the dog paws* activities duration distribution. The green and blue distributions are generated using the true and predicted activities labels.

In general, the predicted activities have lower duration compared to the true ones, which can be explained by the following reasons. Firstly, the post-processing performed to the classifier prediction results discard some predictions by a majority voting window, if the discarded predictions belong to the beginning or end of the activity, correct predictions may be discarded reducing the activity duration. Furthermore, the user's true duration tends to be longer than the actual activity since the user first annotates the start of the activity, then performs it and finally annotates the end of the activity. The time between the annotation and the actual execution of the activity may lead to longer true labels duration. The gap between true and predicted activities duration is not prejudicial since the classifier behaviour is consistent between activities and for further predictions, its behaviour will be similar. Focusing on the true labels, *feeding the dog* and *washing dog paws* activities have less variability than *playing with the dog*, which is an activity that follows a spread distribution that can take from 1 min to 3 min.

The modelled locomotion percentage time distribution is presented in Figure 4.12. Through the analysis of these distributions, it is obvious that the user's most common locomotion mode is *walking*, although the user also stops during the daily walk. Even not so likely, *walking up* and *walking down* also make part of this user routine, filling under 20% of the daily walking.

Focusing on Figure 4.13, it is possible to visualise that the user usually walks between 20 to 40 min, however, sometimes the user performs more challenging walks with a longer distance (more than 4000 m) and consequently taking more time (more than 40 min), as it can be seen on the top distributions of Figure 4.13. The number of steps follows the same behaviour of the distance walked by the user, which makes a good validation for

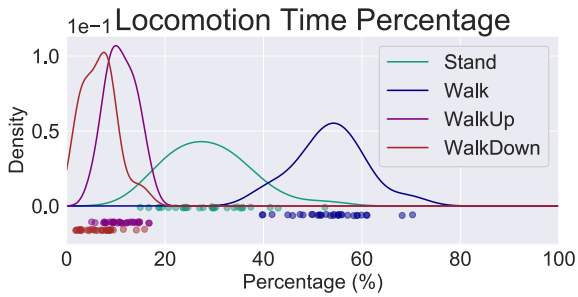


Figure 4.12: Percentage time distribution while *standing*, *walking*, *walking up* and *walking down* coloured in green, blue, purple and brown respectively.

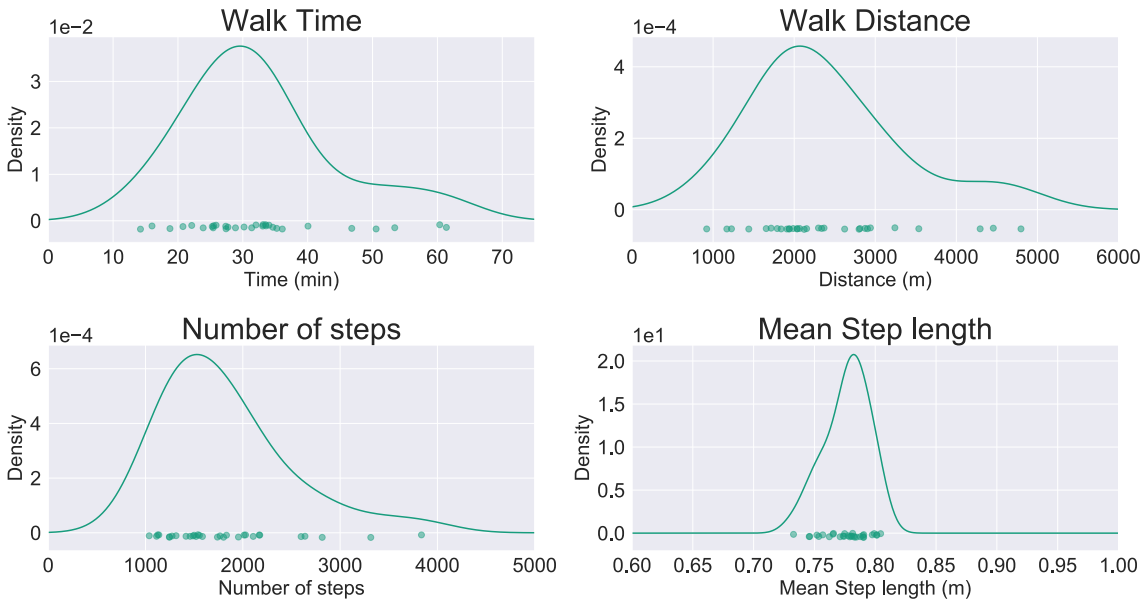


Figure 4.13: Illustration of the walk time, walk distance, number of steps and step length distributions from human mobility on daily walks dataset.

the estimated distance values between GPS measurements and step detection algorithm. The mean step length is in the range of 0.75 to 0.8 m.

Considering velocity features, it is possible to identify mainly two different behaviours depicted by two peaks in the distributions (Figure 4.14).

The altitude distributions in Figure 4.14 shows that the most frequent mean altitude variations conducted by the user are under 20 m and the maximum under 40 m, and it can even exceed the 100 m in a few days.

For HMM sequence probability assessment, the most intuitive way is the use of the returned log-likelihoods. However, as the returned likelihoods arise from summing all possible state sequences for a given sequence of observations that involve the product between all emission probabilities and all transition probabilities, including the start state distribution, the larger the sequence, the smaller will be the log-likelihood. Thus, it is difficult to infer about the likeliness of a given sequence since a small log-likelihood can either be a not likely sequence or a too large sequence. To overcome this problem, the

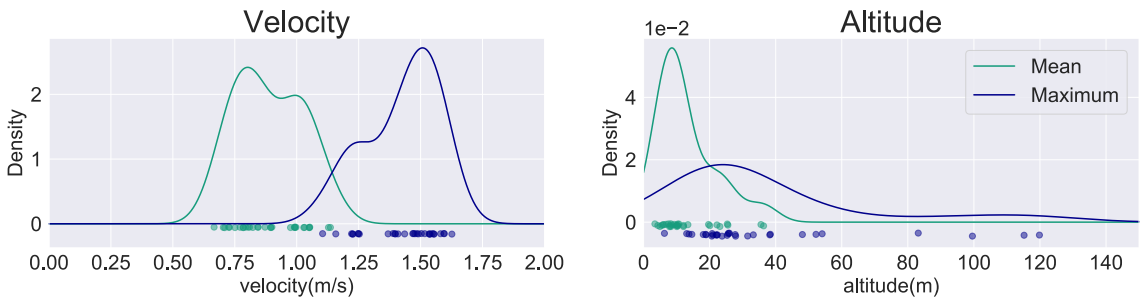


Figure 4.14: From the left to the right is represented the mean and maximum velocity distributions and the mean and maximum altitude variation distributions from human mobility on daily walks dataset. The mean and maximum distributions are coloured in green and blue respectively.

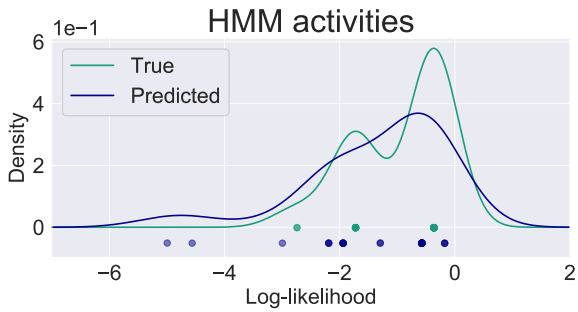


Figure 4.15: Log-likelihood distributions for true and predicted activities sequences represented in green and blue, respectively.

resulting log-likelihood was also modelled into a distribution using KDE, this way it is possible to infer about the probability of a given sequence considering the trained model, regardless of the sequence length.

Figure 4.15 shows two KDE distributions of the resulting log-likelihood of a trained HMM using a set of 31 activity sequences, the green and blue corresponds to the true and predicted activity sequences, respectively. Through the analysis of the distribution and the corresponding results in Table 4.5 it is verified that in both true and predicted activity sequences the most likely sequence is 2, 0, followed by 2, 1, 0, corresponding to the peak values in the true labels KDE distribution.

4.2.2.2 Motion patterns using clustering

Trajectory clusters: Trajectory clusters were found by first creating a distance matrix using DTW distance between all trajectories. This distance matrix was fitted by HDBSCAN to find trajectory clusters with a minimum cluster size of 3. The trajectory clusters results are shown in Figure 4.16. A total of 5 clusters were found in this dataset coloured in blue, brown, orange, purple and green. The black trajectories correspond to the ones that are not similar to any of the found trajectory clusters. Hence, the user presents a large variability on the performed walks, resulting in 5 clusters of spatial similar trajectories. Even inside the trajectory clusters, where trajectories are spatially similar, they do not

Table 4.5: Results of HMM training with the true and predicted activities. The first column indicates the day of acquisition, the second column the activity sequence, where 2 refers to *washing the dog paws*, 1 to *play with dog* and 0 *feeding the dog*, and the third column the log-likelihood that derives from the trained HMM using the corresponding sequences.

Day	Activity Sequence	Log Likelihood
1	2, 1, 0	-1.724
2	2, 0	-0.363
3	2, 0	-0.363
4	2, 0	-0.363
5	2, 0	-0.363
6	2, 1, 0	-1.724
7	2, 1, 0	-1.724
8	2, 0	-0.363
9	2, 1, 0	-1.724
10	2, 0	-0.363
11	2, 1, 0	-1.724
12	2, 0	-0.363
13	2, 0	-0.363
14	2, 0	-0.363
15	2, 0	-0.363
16	2, 1, 0	-1.724
17	0	-2.741
18	2, 1, 0	-1.724
19	2, 1, 0	-1.724
20	2, 0	-0.363
21	2, 0	-0.363
22	2, 0	-0.363
23	2, 1, 0	-1.724
24	2, 0	-0.363
25	0	-2.741
26	2, 0	-0.363
27	2, 1, 0	-1.724
28	2, 0	-0.363
29	2, 0	-0.363
30	2, 0	-0.363
31	2, 0	-0.363

(a) True activity results.

Day	Activity Sequence	Log Likelihood
1	2, 1, 0	-1.941
2	2, 0	-0.573
3	2	-0.177
4	2, 0	-0.573
5	2, 0	-0.573
6	2, 1, 0	-1.941
7	2, 1, 2, 1, 0	-4.564
8	2	-0.177
9	2, 1	-1.295
10	2, 0	-0.573
11	2, 1, 0	-1.941
12	2, 0	-0.573
13	2, 0	-0.573
14	1	-2.995
15	2, 0	-0.573
16	2, 1, 0	-1.941
17	0	-2.191
18	1, 2, 0	-4.997
19	2, 1	-1.295
20	2	-0.177
21	2, 0	-0.573
22	2, 0	-0.573
23	2, 1, 0	-1.941
24	2, 0	-0.573
25	0	-2.191
26	2, 0	-0.573
27	2, 1, 0	-1.941
28	2, 0	-0.573
29	2, 0	-0.573
30	0	-2.191
31	2, 0	-0.573

(b) Predicted activity results.

correspond exactly to the same path. This way, the behaviour of this user in terms of trajectories does not follow a specific pattern, but instead it is represented by mainly 5 patterns containing some variability.

Points of Interest: The user's points of interest were discovered by implementing a DBSCAN clustering method that considers within a radius of 50 m all stand locations that happened for at least 1 min. Thus, a total of four POI were found and their locations can be visualised in Figure 4.17. Since these POI were not annotated, it was asked to the user to validate the obtained POI. All POI represent meaningful locations to the user, being the blue POI the user's home, the orange POI the supermarket and the purple and green POI two gardens where the user usually stops to speak with other dogs owners. Hence, the found POIs are not just some locations where the user stops but are locations of interest to the user.

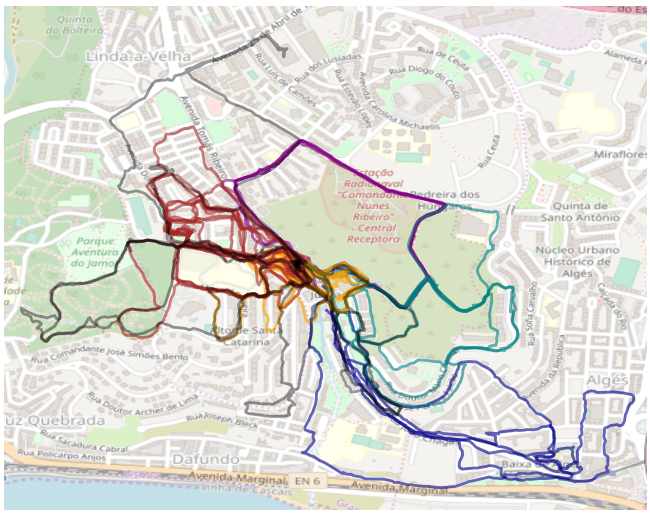


Figure 4.16: Representation of the trajectory clusters by spatial similarity. The found clusters are coloured in blue, brown, orange, purple and green. The black trajectories the ones that does not belong to any cluster.

Figure 4.17: Illustration of the locations that corresponds to the user POI. Four POI are identified, coloured in blue is the user's home, in green and purple, two gardens and in orange the supermarket.

Sensory signatures: Sensory signatures clusters are found if the signals with an acceleration higher than 11 m/s^2 occur within a radius of 50 m in at least 3 days. DBSCAN clustering method was implemented for this search and the resulting clusters are shown in Figure 4.18. Here two clusters can be observed, one corresponding to the user's home and the other to a garden that belongs to the trajectory patterns of this user. For validation purposes, the user was wondered about the activities performed in those locations, turns out that these two clusters correspond to locations where the user usually plays with the dog. It makes perfect sense, as *playing with the dog* activity involves rapid movements from the user, being characterised for having a high acceleration.

Figure 4.18: Representation of the location where sensory signatures occur.

4.2.3 Anomaly Detection

The previous subsection describes how patterns are discovered considering all normal days of a user's daily routine. However, using the rules from Figure 3.6 a number of 14 days was set to the minimum learning period to learn each feature pattern ($day_{learn} = 14$) and 5 consecutive days were used to predict an anomaly ($day_{anom} = 5$). For $Dpattern$ all feature weights were assigned to one, however, weight optimisation should be considered in future work.

Depending on the user and on the selected features, the learned pattern can not completely describe the real user behaviour, however, since the pattern model is updated daily, the patterns will become more robust along the days. Thus, anomalies will be detected considering the previous normal days.

The use case for anomalies detection, based on human mobility on daily walks dataset, concerns a user that starts to face reduced mobility. The extracted patterns that make sense for the detection of reduced mobility are the ones that are directly related to the user's mobility, namely the locomotion percentage time, the number of steps, mean step length, walking time, walking distance, mean and maximum velocity and mean and maximum altitude variation distributions.

In order to measure the distance to the pattern of each feature, according to the KDE distributions, the distributions were normalised by its maximum density value. The distance to the pattern is given by $1 - KDE_{distance}$, where the $KDE_{distance}$ corresponds to the intersection value of the feature and the normalised distribution.

As mentioned before, in this use case, a range of 5 days was considered to make a decision about an anomaly, however, the developed framework is flexible enough to change the range of days easily for testing other case studies that need a different range of days.

The distance of each day to the pattern was obtained and the results are represented in

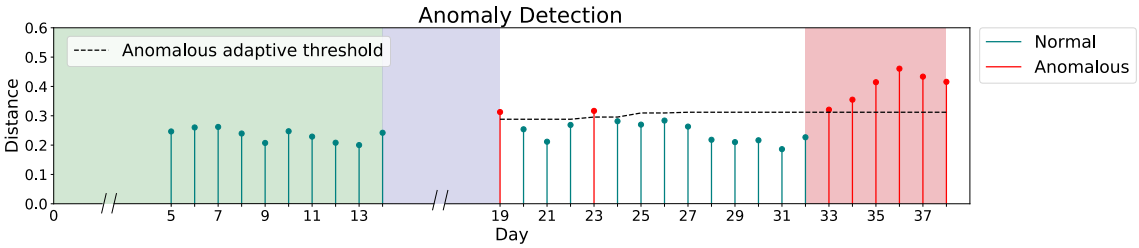


Figure 4.19: Representation of the anomaly detection results. The x-axis corresponds to the days of the user's behaviour and the y-axis to the distance to the pattern. The green and red stem correspond to normal and anomalous days, respectively. The streak line defines the anomalous adaptive threshold. The green region corresponds to the distances for first defining the pattern and the threshold, the blue region corresponds to the day_{anom} and the red region is the ground truth of the anomalous days.

Figure 4.19. The anomalous threshold was defined by equation 3.12, and it was updated daily according to normal behaviour.

As the behaviour is defined within a range of 5 days, each distance represented in Figure 4.19 corresponds to the mean distance to the pattern regarding 5 days. The anomalous threshold is only defined after the learned pattern, which occurs after 14 days. On day 19 the behaviour starts to be evaluated, considering the behaviour of the last 5 days, according to the previous pattern and the defined threshold. The threshold is continuously updated unless an anomaly is detected as well as the pattern. Comparing the detected anomalies and the ground truth (red region), day 19 and 23 were incorrectly assigned as an anomaly, perhaps the user had a distinct locomotion behaviour on the last 5 days that lead to a larger anomalous distance. For this specific user, the used day_{learn} (14 days) might be insufficient and more days should be considered for learning the correct user pattern. Day 32 was not correctly detected as an anomaly, which is acceptable since it considers the behaviour of the last 5 days that were normal, so the mean distance tends to be reduced.

The overall results were satisfactory, however, we identified key points that can lead to improvements in this framework and should be addressed in future work, namely:

- *Dpattern* weights optimisation for each feature.
- Weights parametrization based on the intrinsic characteristics of each feature, for instance, to evaluate the reduced mobility on daily walks an increase on the mean velocity should have a different weight than a decrease on the mean velocity.
- Automatically detection of the number of days needed to learn the pattern.

4.3 Morning daily living routine dataset

Morning daily living routine dataset aims to evaluate the behaviour motion patterns of the User 2, focusing on the indoor environment. Thus, in this dataset, we are interested

Figure 4.20: *Making the bed*, *cooking*, *eating*, *washing the dishes* and *brushing teeth* activities acquisition time in minutes and repetitions inside the brackets.

in knowing the activities performed by the user comprising complex activities of his/her daily living such as *making the bed*, *cooking*, *eating*, *washing the dishes* and *brushing teeth*, and the user's location in a room-level. Similar to human mobility on daily walks dataset (see section 4.2), the user was asked to perform a set of complex activity repetitions and to start data recording in each room separately for the room-level indoor location. The activity acquisition time is described in Figure 4.20. The eating activity has more recorded time than the remaining, since it is an activity with longer duration, however, the dataset was balanced before training.

Room-level indoor location relies on data recordings of complex activity recognition, separated by division. The bedroom used *making the bed*, the kitchen used *eating* and *cooking* and the bathroom used *brushing teeth* data recordings.

Similar to human mobility on daily walks dataset 4.2, this dataset also covers normal and anomalous days of the daily routine of this user over a period of 6 months. The use case for the anomaly detection in this dataset was a simulation of an user that starts to experience dementia impairment. For this purposes the anomalies planned on the user's routine are related to dementia behaviour [3], regarding absence of activities that reflect the user's difficulty or disinterest in performing the activity, activity sequences that are not common on the normal behavior of the user and increased stay in certain home divisions that may be indicative that the user is feeling depressive.

The dataset overall information is summarised in Table 4.6.

Considering the corrupted files and missing information, from 57 days of acquisition, only 48 days were analysed on anomaly detection, from which 34 were normal and 14 anomalous.

4.3.1 Feature Extractor

In morning daily living routine dataset, the feature extractor step of the proposed framework addressed the recognition of locomotion activities (previously described in subsection 4.2.1.1), the indoor location, location sequence, complex activities from the morning daily living routine and the complex activities sequence.

Since the locomotion activity classifier is user-independent, the detailed description of the training process was already explained in subsection 4.2.1.1.

Table 4.6: Dataset information regarding the user ID, activities performed, activated sensors, acquisition device, device position and recording days and hours.

Dataset Information		
User ID	2	
Activities	Simple	Complex
	Standing, Walking, Walking up, Walking down	Making the bed, Cooking, Eating breakfast, Washing the dishes, Brushing teeth
Sensors	Accelerometer (200Hz), Gyroscope (200 Hz), Magnetometer (50 Hz), Barometer (30 Hz), Sound (8000 Hz) , Location (0.05 Hz), Wi-Fi (0.3 Hz)	
Acquisition Device	LG Nexus 5	
Device Position	Dominant Wrist	
Recording Days	Normal	Anomalous
	40	17
Recording Hours	Normal	Anomalous
	~32 hours	~12 hours

4.3.1.1 Indoor location classifier

The activities of daily living occur in specific locations, therefore understanding the location where the user is along the day can provide insights about the activities being performed, or even help to find the user in an emergency. Moreover, location patterns can also bring useful information to understanding the user's behaviour. Therefore, a room-level indoor location classifier was implemented.

The indoor location relying on room level recognition used data records from each room of the user's house. The implementation of a statistical classifier regarding the Wi-Fi RSSI of the unique IDs from several AP lead to an accuracy of 93.6%. The corresponding normalised confusion matrix is in Figure 4.21, where the None class corresponds to every location where the classifier certainty is bellow 60%, this includes the outdoor, corridors and the entrance hall.

4.3.1.2 Morning activities classifier

For the recognition of complex activities, namely *making the bed*, *washing the dishes*, *cooking*, *eating* and *brushing teeth*, separated activities from routines were acquired for training the classifier and the accuracy and F1-score of various machine learning classifiers were evaluated. For the training process, the user was asked to record data while performing each of the morning activities. The train set was composed of 15 repetitions, of each activity, and the test set was composed by the annotated activities during 54 days of the morning routine. For this recognition problem, a window size of 20 seconds was chosen.

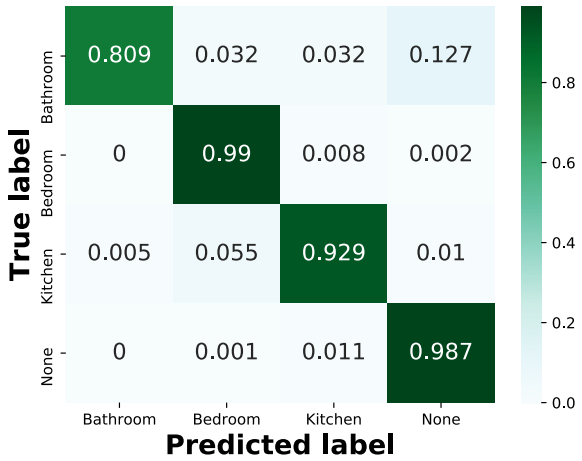


Figure 4.21: Normalised confusion matrix of a room-level indoor location recognition.

Table 4.7: Results of supervised learning classifiers tested in all days of the morning daily living routine dataset in terms of accuracy and F1-score before and after post-processing. The best results are in bold.

Classifier	Accuracy (%) Before PP	F1-Score (%) Before PP	Accuracy (%) After PP	F1-Score (%) After PP
DT	89.7%	90.0%	97.7%	97.7%
KNN	92.5%	92.4%	96.8%	96.9%
RF	90.6%	90.9%	**98.1%**	**98.1%**
NB	62.9%	49.2%	72.2%	61.0%
ADA	88.5%	87.9%	94.2%	93.9%

Additionally, an overlap of 30% is used between window sizes to enhance relevant features of the activity that may be overshadowed by partitioning the signal into fixed size windows. The features were extracted using TSFEL library, and a feed-forward feature selection using 10-fold cross-validation was applied. The KNN classifier was the one that achieved the highest accuracy (92.5%) comparing to DT, Random Forest (RF), NB and AdaBoost (ADA) (see Table 4.7).

Similar to the previous dataset (section 4.2), a post-processing was applied to the prediction labels. In this dataset, the post-processing was constituted by two stages. Firstly, predictions were accepted based on activities location by using the indoor location prediction from subsection 4.3.1.1 (see Figure 4.22). As some activities can only be performed in certain home divisions, all predictions that did not match their location are discarded. The second stage corrects predictions by majority voting as described in 4.2.1.2.

After the post-processing the RF classifier obtained the highest accuracy with a value of 98.1% (see Table 4.7) using the following features: mean y-axis acceleration (*Acc y Mean*), mean z-axis magnetometer value (*Mag z Mean*), and mean gyroscope magnitude (*Gyro mag Mean*). The normalised confusion matrix using RF after post-processing is presented in Figure 4.23.

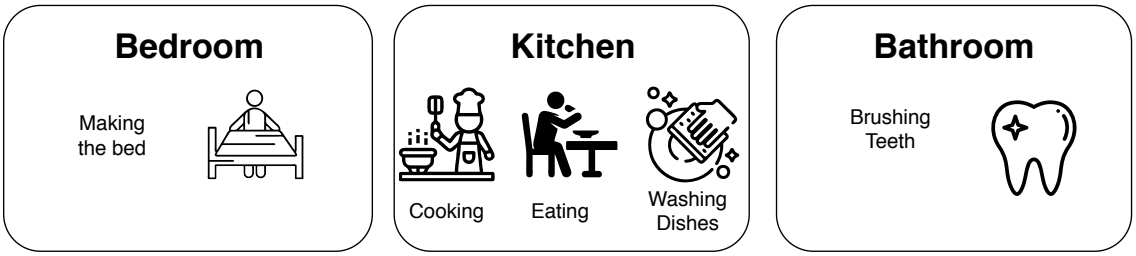


Figure 4.22: Locations represented by squares with the corresponding activities within.

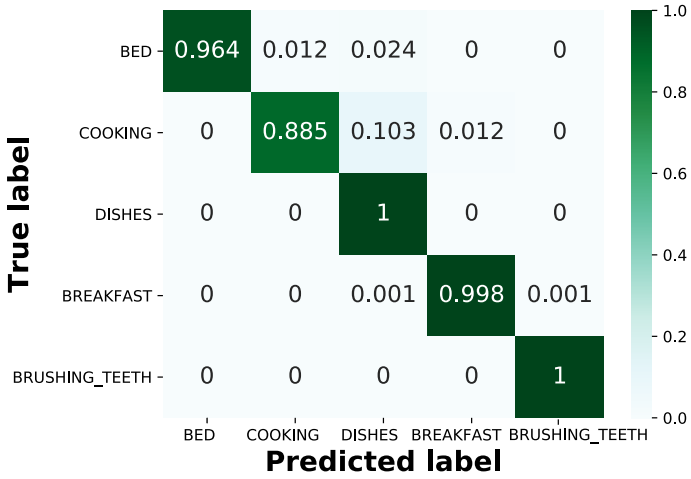


Figure 4.23: Normalised confusion matrix for morning activities classification, using RF classifier, after post-processing.

From the interpretation of horizon plot (4.24), it is easy to see the contribution of each feature for the recognition problem, for instance, *making the bed* activity can be distinguished from the remaining because it has a great negative *Acc y Mean* and great positive values of *Mag z Mean* and *Gyro mag Mean*. In contrast, *eating* activity has a positive *Acc y Mean* and negative *Mag z Mean* and *Gyro mag Mean*.

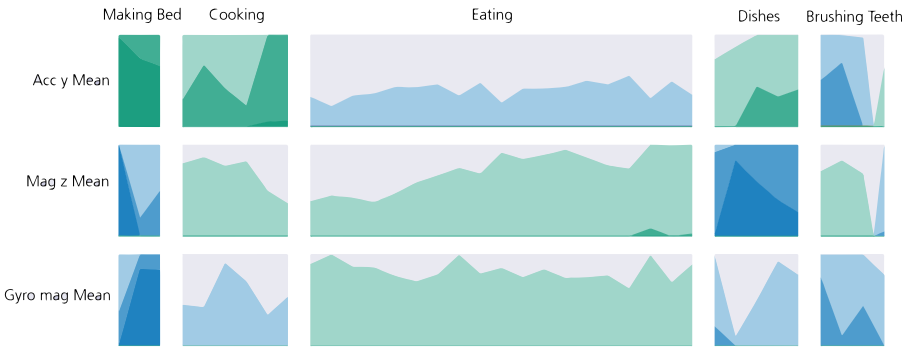


Figure 4.24: Horizon plot with selected features represented from top to the bottom. The x-axis contains the five activities, namely *making the bed*, *cooking*, *eating*, *washing the dishes* and *brushing teeth*. The range of y-axis corresponds to a third of the maximum value of each feature corresponding to the darkest values. The blue and green colours correspond to positive and negative values, respectively.

4.3.1.3 Morning activities and location sequence

HMM were implemented in this dataset to evaluate the probability of a given sequence of activities and locations performed by the user. The sequence of activities is important for the evaluation of the cognitive behaviour of the user, as a sequence that is too different from the usual sequences performed by the user may be an alarm situation. All 5 activities from morning activities classifier were used to learn activities sequence. The number of hidden states was estimated through BIC, resulting in 2 hidden states, since it was the number of states that lead to a larger BIC value. Thus, the model was trained using 2 hidden states and the activities sequences performed by the user during 34 days. Regarding location sequence, the number of hidden states was also 2 and the HMM was implemented from both true labels and predictions along 34 days.

4.3.2 Patterns Discovery

The morning daily living dataset was evaluated only on indoor environment. Thus, trajectory clusters, POI and signal signatures were not applied in this dataset.

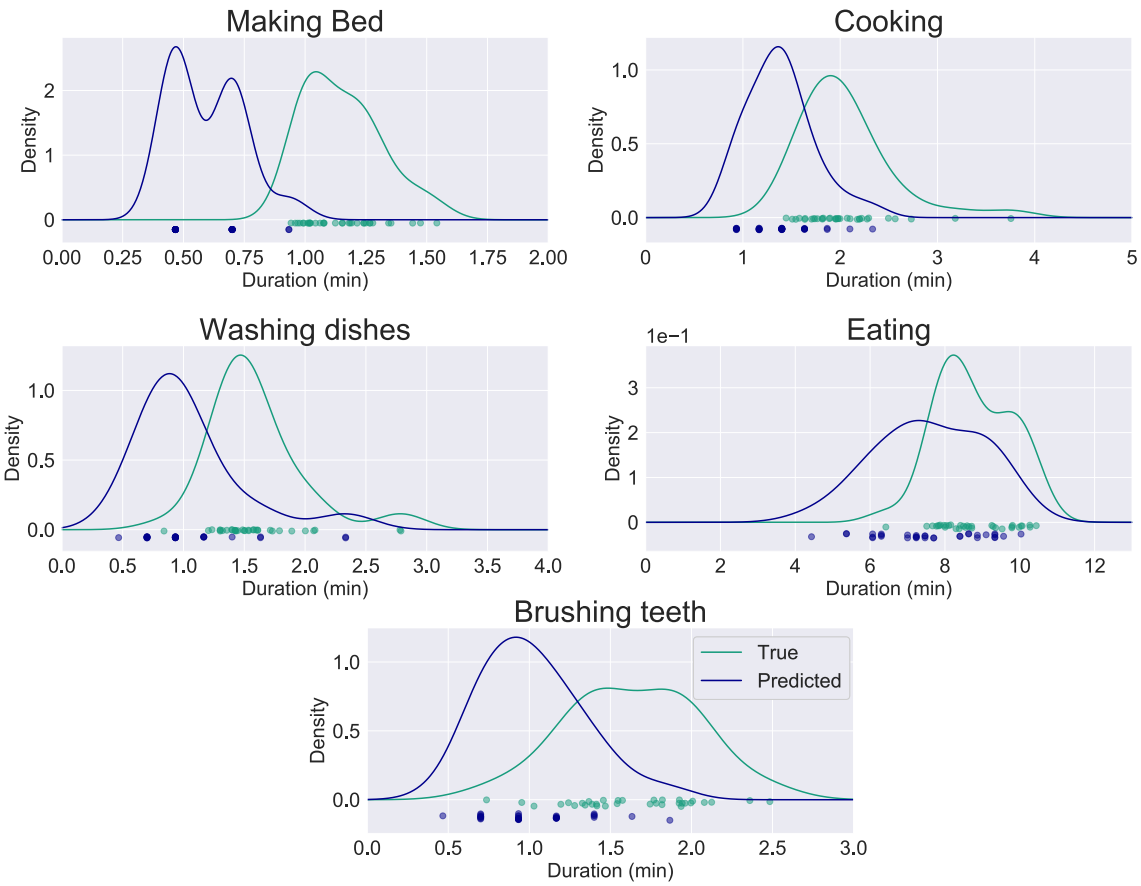


Figure 4.25: Five features modelled into a distribution using KDE, namely *making the bed*, *cooking*, *washing the dishes*, *eating* and *brushing teeth* duration. The green and blue distribution is generated using the true and predicted activities labels, respectively.

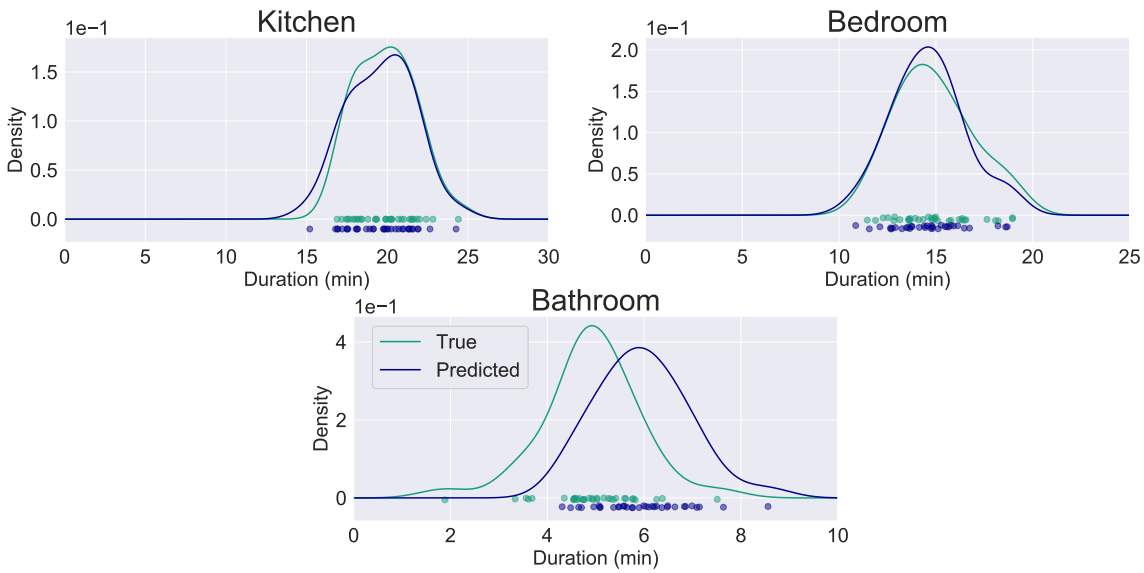


Figure 4.26: Illustration of the distribution of the time spent in each location from true and predicted labels, coloured in green and blue respectively.

The extracted features on this dataset include activities duration (Figure 4.25), time spent in each location (Figure 4.26), the locomotion percentage time (Figure 4.27) and walking features (Figure 4.28). These features were all modelled into a distribution using KDE, this way it is possible to learn the user's behaviour in terms of indoor environment.

Figure 4.25 represents five distributions including *making the bed*, *cooking*, *washing the dishes*, *eating* and *brushing teeth* duration. Focusing on the activity duration distribution, the predicted distributions present lower duration than true distributions, the reasons for this behaviour can be explained by the post-processing and annotation process as described in subsection 4.2.2.1.

Figure 4.26 contains the distributions corresponding to the time spent in each location, namely *kitchen*, *bedroom* and *bathroom*, from the true and predicted labels. Regarding the true labels, the location where more time was spent is in the *kitchen*, which makes senses since it is the location where the user performs more complex activities such as *cooking*, *eating* and *washing the dishes*.

Comparing both distributions, predicted distributions are in fact similar to the ones that use true labels, except for the *bathroom* distribution that is shifted to the right, thus with higher time spent in the bathroom.

Due to the type of complex activities performed in this dataset, it is expected that *standing* activity is the one with highest % of duration, as it can be confirmed by Figure 4.27. Observing the number of steps and step length distribution, it is possible to note that there are not much variability (see Figure 4.28).

The HMM activities in Figure 4.29 illustrates the log-likelihood of true and predicted activity sequences modelled using a KDE. Table 4.8 contains the resulting log-likelihoods of the trained HMM using activity sequences extracted from the classifier's true labels

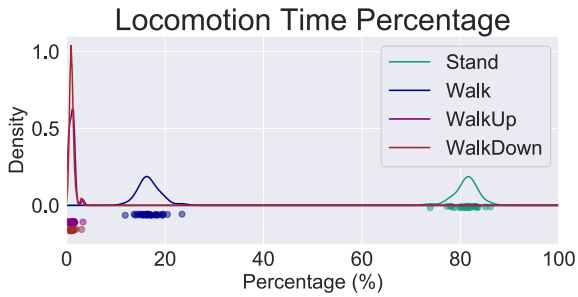


Figure 4.27: Percentage time distribution while *standing*, *walking*, *walking up* and *walking down* coloured in green, blue, purple and brown respectively.

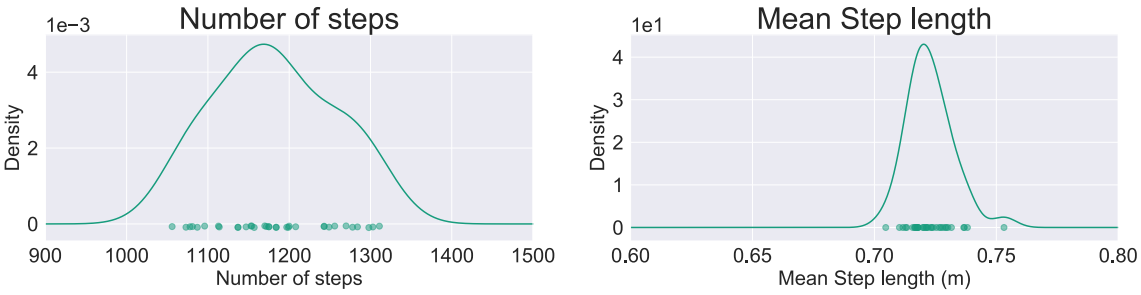


Figure 4.28: From the left to the right is represented the distributions containing the number of steps performed by the user and the mean step length distribution.

and prediction results. Activity 0 refers to *making the bed*, 1 to *cooking*, 2 to *washing the dishes*, 3 to *eating* and 4 to *brushing teeth*. By the analysis of the true distribution, it is noticeable two peaks with log-likelihood around -7 and -5 that correspond to sequences 0, 1, 3, 4, 2 and 0, 1, 3, 2, 4, respectively. Although the KDE using the true and predicted labels are not exactly the same KDE distribution, the most probable likelihood of the predicted distribution includes the same sequences. Therefore, the two most likely activity sequences, according to the true sequences, are in fact probable according to both models.

Focusing on the results of the true labels (Figure 4.29), the results of the HMM applied to locations are not very informative since the user does not follow a specific sequence pattern, there are some common sequences along the days, characterised by a higher density points in x-axis, but the overall distribution is spread.

Thus, for finding patterns in sequences using HMM it is necessary that the outcome distribution is well-defined, since a small variation in the log-likelihood refers to an alteration on the performed sequence that may have a pathological meaning, so a pattern in sequences modelled through HMM should follow a stable distribution.

4.3.3 Anomaly Detection

The morning daily living routine dataset aims to detect anomalies on users behaviour that reflect the behaviour of someone starting to experience dementia behaviour. For this purpose, only the features that may characterise this behaviour were accounted for,

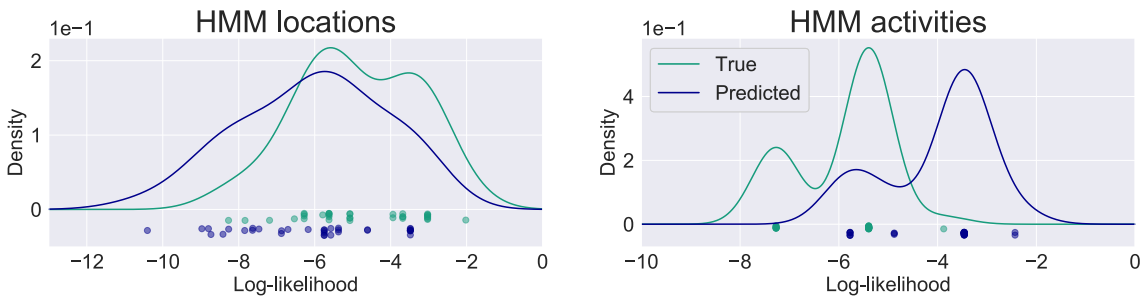


Figure 4.29: From the left to the right is represented the log-likelihood distributions for true and predicted location sequences and the log-likelihood distributions for true and predicted activities sequences represented in green and blue, respectively.

Table 4.8: Results of HMM training with the true (Table 4.8a) and predicted activities (Table 4.8b). The first column indicates the day of acquisition, the second column the activity sequence where activity 0 refers to *making the bed*, 1 to *cooking*, 2 to *washing the dishes*, 3 to *eating* and 4 to *brushing teeth*. The third column contain the log likelihood that derives from the trained HMM using the corresponding sequences.

Day	Activity Sequence	Log Likelihood
1	0, 1, 3, 4, 2	-7.272
2	0, 1, 3, 2, 4	-5.395
3	0, 1, 3, 2, 4	-5.395
4	0, 1, 3, 2, 4	-5.395
5	0, 1, 3, 2, 4	-5.395
6	0, 1, 3, 2, 4	-5.395
7	0, 1, 3, 2, 4	-5.395
8	0, 1, 3, 2, 4	-5.395
9	0, 1, 3, 4, 2	-7.272
10	0, 1, 3, 2, 4	-5.395
11	0, 1, 3, 2, 4	-5.395
12	0, 1, 3, 2, 4	-5.395
13	0, 1, 3, 2, 4	-5.395
14	0, 1, 3, 4, 2	-7.272
15	0, 1, 3, 4, 2	-7.272
16	0, 1, 3, 4, 2	-7.272
17	0, 1, 3, 4, 2	-7.272
18	0, 1, 3, 2	-3.876
19	0, 1, 3, 4, 2	-7.272
20	0, 1, 3, 4, 2	-7.272
21	0, 1, 3, 2, 4	-5.395
22	0, 1, 3, 2, 4	-5.395
23	0, 1, 3, 2, 4	-5.395
24	0, 1, 3, 2, 4	-5.395
25	0, 1, 3, 2, 4	-5.395
26	0, 1, 3, 2, 4	-5.395
27	0, 1, 3, 2, 4	-5.395
28	0, 1, 3, 2, 4	-5.395
29	0, 1, 3, 2, 4	-5.395
30	0, 1, 3, 4, 2	-7.272
31	0, 1, 3, 4, 2	-7.272
32	0, 1, 3, 4, 2	-7.272
33	0, 1, 3, 2, 4	-5.395
34	0, 1, 3, 2, 4	-5.395

(a) True activity results.

Day	Activity Sequence	Log Likelihood
1	0, 1, 3, 2	-2.432
2	0, 1, 3, 2, 4	-3.462
3	0, 1, 3, 2, 4	-3.462
4	0, 1, 3, 2, 4	-3.462
5	1, 3, 2, 4	-4.878
6	0, 1, 3, 2, 4	-3.462
7	1, 3, 2, 4	-4.878
8	0, 1, 3, 2, 4	-3.462
9	0, 1, 3, 4	-3.468
10	1, 3, 2, 4	-4.878
11	0, 1, 3, 2, 4	-3.462
12	0, 1, 3, 2, 4	-3.462
13	0, 1, 3, 2, 4	-3.462
14	0, 1, 3, 4, 2	-5.773
15	0, 1, 3, 4, 2	-5.773
16	0, 1, 3, 4, 2	-5.773
17	0, 1, 3, 4	-3.457
18	0, 1, 3, 2	-2.432
19	0, 1, 3, 4, 2	-5.773
20	0, 1, 3, 4, 2	-5.773
21	0, 1, 3, 2, 4	-3.462
22	0, 1, 3, 2, 4	-3.462
23	0, 1, 3, 2, 4	-3.462
24	0, 1, 3, 2, 4	-3.462
25	0, 1, 3, 2, 4	-3.462
26	0, 1, 3, 2, 4	-3.462
27	0, 1, 3, 2, 4	-3.462
28	0, 1, 3, 2, 4	-3.462
29	0, 1, 3, 2, 4	-3.462
30	0, 1, 3, 4, 2	-5.773
31	0, 1, 3, 4, 2	-5.773
32	0, 1, 3, 2, 4	-3.462
33	0, 1, 3, 2, 4	-3.462
34	0, 1, 3, 2, 4	-3.462

(b) Predicted activity results.

namely the activity sequence probability, duration of each activity, time spent in each location, number of entries per location and inactivity percentage time.

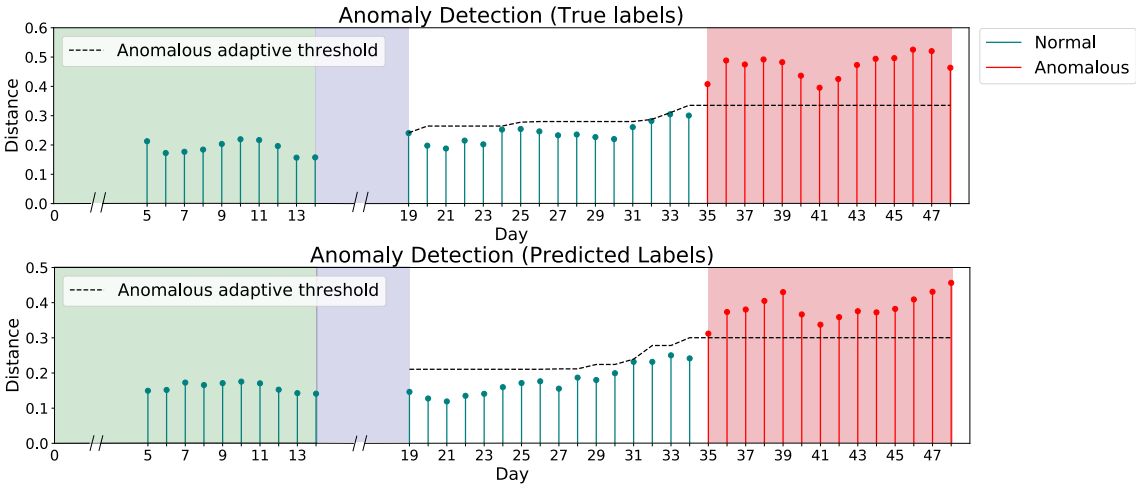


Figure 4.30: From the top to the bottom is represented the anomaly detection using the true and the predicted labels. The x-axis corresponds to the days of the user's behaviour and the y-axis to the distance to the pattern. The green and red stem correspond to normal and anomalous days, respectively. The streak line defines the anomalous adaptive threshold. The green region the distances for first defining the pattern and the threshold, the blue region corresponds to the day_{anom} and the red region is the ground truth of the anomalous days.

Using the fluxogram from Figure 3.6, day_{learn} was set to 14 and a range of 5 days (day_{anom}) for anomaly detection was used. The $Dpattern$ weights were set to one.

Figure 4.30 illustrates the results for anomaly detection along 48 days of the user's behaviour. Comparing the results of the anomaly detection using the true and the predicted labels, despite the overall distances are different, the developed algorithm shows promising results since it correctly predicts the anomalous days of the user behaviour that starts in day 35, using both true and predicted labels.

4.4 Lunchtime routine dataset

Lunchtime routine dataset was recorded for the analysis of human trajectories behaviour on the outdoor environment during the User 2 lunchtime. This dataset contains 47 normal and 10 anomalous days of acquisition, over a period of 6 months, the dataset information can be visualised in Table 4.9. The use case for anomaly detection on this dataset concerns a human that starts to experience loss of memory episodes, getting confused and getting lost when trying to reach a specific destination. The most important source of information for this dataset is GPS measurements. Unfortunately, these acquisitions had serious issues with GPS measurements due to missing GPS data and only 26 days were viable to be evaluated, from which 19 were normal days and 7 were anomalous.

For the feature extractor step, features from the locomotion classifier (subsection 4.2.1.1), walking features (such as walking time, step length or mean velocity), trajectory clusters and POI were extracted.

Table 4.9: Dataset information regarding the user ID, activities performed, activated sensors, acquisition device, device position and recording days and hours.

Dataset Information		
User ID	2	
Activities	Simple	
	Standing, Walking, Walking up, Walking down	
Sensors	Accelerometer (200Hz), Gyroscope (200 Hz), Magnetometer (50 Hz), Barometer (30 Hz), Sound (8000 Hz) , Location (0.05 Hz), Wi-Fi (0.3 Hz)	
Acquisition Device	LG Nexus 5	
Device Position	Dominant Wrist	
Recording Days	Normal	Anomalous
	47	10
Recording Hours	Normal	Anomalous
	~54 hours	~13 hours

4.4.1 Pattern Discovery

Similar to the previous datasets, KDE and clustering methods were used to model features from feature extractor step of the framework (section 3.1).

4.4.1.1 Motion patterns using KDE

The user's locomotion modelled using KDE is shown in Figure 4.31. From the interpretation of this figure, it is possible to verify that the *standing* and *walking* activities took the majority of the lunchtime. Additionally, the user walks are usually simple walks since they do not have many climbs or declines.

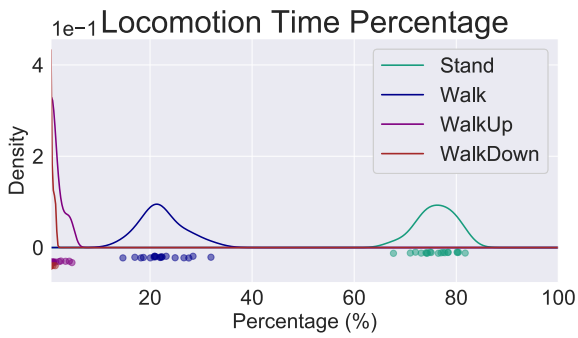


Figure 4.31: Percentage time distribution while *standing*, *walking*, *walking up* and *walking down* coloured in green, blue, purple and brown respectively.

Regarding walk features, the usual walks performed by the user takes around 25 min in a range of 1000 m to 2000 m. Since this dataset also contains an indoor component, the number of steps is usually higher compared to the walk distance outdoor. The user step length is very characteristic in the range of 0.7 m to 0.75 m. This information can be

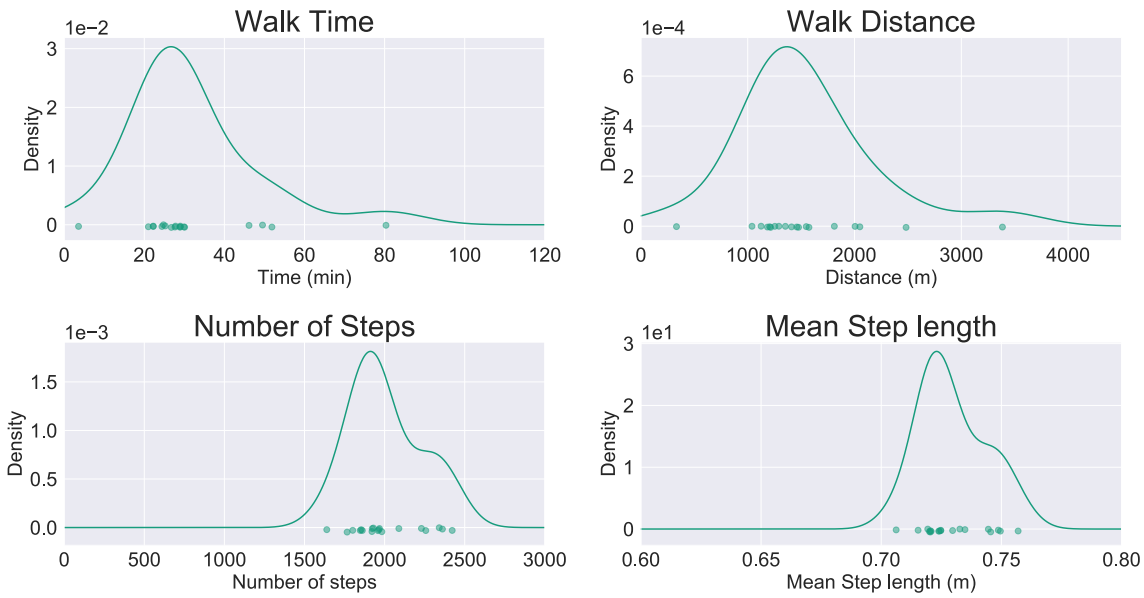


Figure 4.32: Illustration of the walk time, walk distance, number of steps and step length distributions from lunchtime routine dataset.

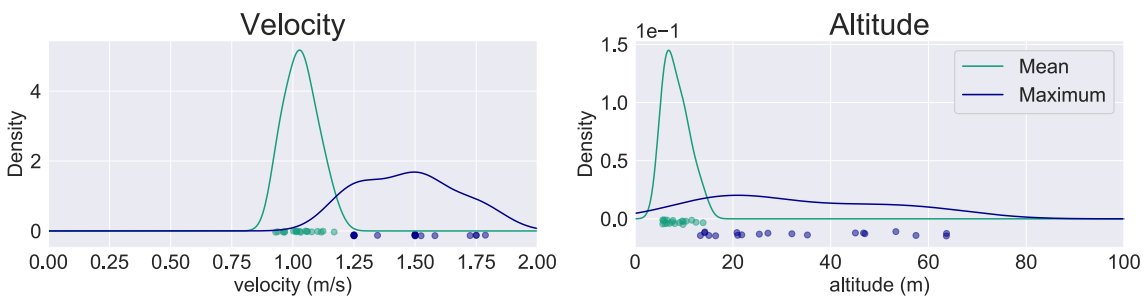


Figure 4.33: From the left to the right is represents the mean and maximum velocity distributions and the mean and maximum altitude distributions from lunchtime routine dataset. The mean and maximum distributions are coloured in green and blue respectively.

verified in the distributions illustrated in Figure 4.32. Considering the user velocity in Figure 4.33, the mean velocity of the user is very stable around 1 m/s.

As verified in the user's locomotion, the walks performed by the user do not present difficulty and this is in accordance with Figure 4.33, as the mean altitude variation is under 20 m and the maximum can exceed the 40 m.

4.4.1.2 Motion patterns using clustering

For trajectory clusters, the same reasoning applied for the discovery of trajectories clusters in subsection 4.2.2.2 was applied in this dataset. Hence, the resulting clusters are in Figure 4.34. The user contains two specific patterns in the trajectory since two clusters that correspond exactly to the same paths were found.

The discovery of POI of this user follows the same methods as in trajectory clusters

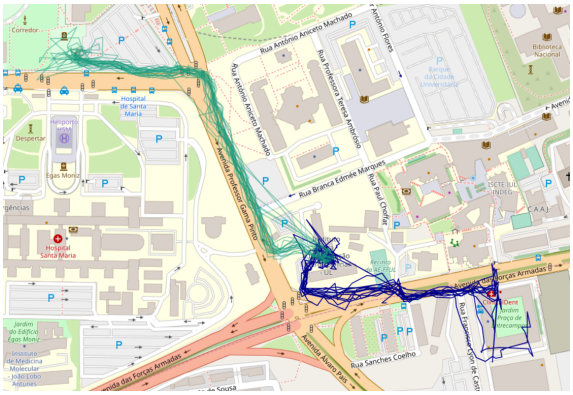


Figure 4.34: Representation of two trajectory clusters coloured in green and blue.

Figure 4.35: Illustration of the three user's POI, two coffees coloured in green and purple and the working place coloured in blue.

of subsection 4.2.2.2. Three locations of interest of this user were obtained, which are represented in Figure 4.35. The blue POI seems to be common to both trajectories and the origin of the trajectory, if we look closer, the specified location, after validation with the user, corresponds to its working place. The purple POI occupies a region containing a coffee where the user usually goes, such as the green POI.

4.4.2 Anomaly Detection

The use case of this dataset aims to detect anomalies when a person starts to experience loss of memory that leads to confusion to reach a specific destination, getting lost and performing different trajectory paths.

Considering 26 days of the lunchtime routine dataset, the first 14 days were used to train HDBSCAN clustering method, resulting in two clusters represented in Figure 4.36 by blue and green colour. By learning more days, the trajectories that were recognised as noise by HDBSCAN clustering method were considered anomalous trajectories, since they are spatially different from the user's trajectories patterns. For the trajectories that

were assigned to a cluster, the distances of each trajectory with all trajectories within the assigned cluster were computed, as described in subsection 3.3.2, and the ones that had a maximum distance higher than the threshold computed as explained in subsection 3.3.2 were considered anomalous trajectories. The anomalous trajectories are coloured in red in Figures 4.36 and 4.37.

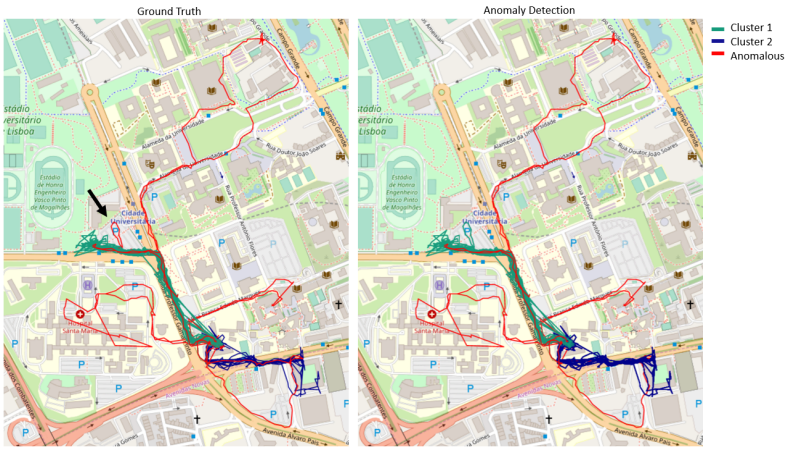


Figure 4.36: Anomaly trajectories detection from lunchtime routine dataset. The left image represents the ground truth of the clusters coloured in green, and blue and the anomalous trajectories in red, with the non detected anomalous trajectory pointed with a black arrow. The right image illustrates the results of the anomaly detection algorithm.

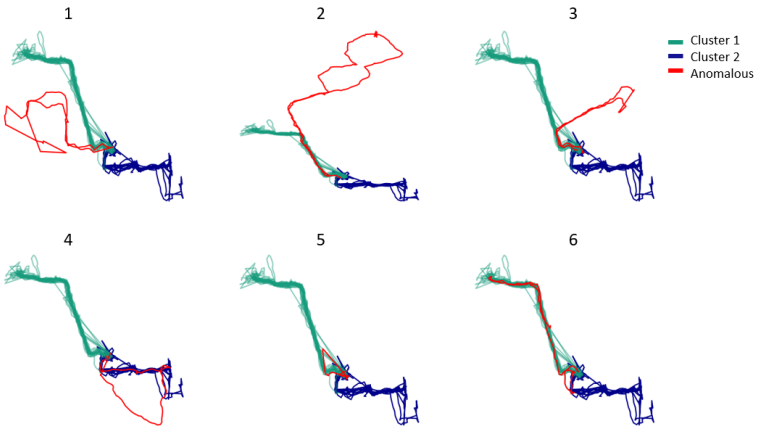


Figure 4.37: Representation of the 6 anomalous trajectories coloured in red together with the trajectory clusters coloured in green and blue.

Figure 4.36 represents the ground truth and overall results of the anomaly detection algorithm by continuously learning the trajectory patterns. Comparing the two figures, only one anomalous trajectory was incorrectly assigned to a cluster, this trajectory is highlighted using a black arrow. In Figure 4.37 it is more clear the visualisation of the anomalies, these are either characterised by trajectories that are spatially distinguishable

(trajectories 2 and 3), or trajectories that although are spatially similar present a large distance compared to the assigned cluster (trajectories 1, 4, 5, 6).

Even though, the algorithm did not properly recognise one of the anomalous trajectories, the results were still satisfactory.

Chapter 5

Conclusion and Future Work

5.1 Conclusions

The lack of routine screening of the ageing population, especially, for the people who live home alone is a serious concern since routine alterations and difficulty on performing certain daily activities are some of the common symptoms of cognitive impairments. Furthermore, home accidents and disorientation are also common issues regarding the elderly population that if it is not detected on time may lead to severe consequences. Thus, the main contribution of this study was the development of a framework for learning human pattern behaviour by trajectory analysis and activity recognition.

Smartphones and wearable sensors are an extraordinary useful source of information about human behaviour, due to their sensing capabilities and their ubiquity. Concerning the public available datasets regarding human behaviour, neither of the found datasets contain suitable data for a long period to draw conclusions about human routines. Therefore, a data collection deployment collecting multiple sources of smartphone' sensors data for over 6 months was performed, including location, inertial and environment data. Additionally, for validation purposes, an annotation process was considered during data recording and is also included in the database. During this long period, three datasets were designed for addressing challenges ranging from unsupervised human mobility analysis to the recognition of complex activities and their sequence in an indoor or outdoor environment.

The developed framework includes three main steps namely the human behaviour feature extraction, pattern discovery methods and anomaly detection. Acknowledging the machine learning classifiers implemented in this study the locomotion activities classifier reached the best accuracy of 90.2% using DT classifier, the complex activities classifiers implemented for human mobility on daily walks dataset and morning daily

living routine dataset achieved, respectively, an accuracy of 94.0% using KNN classifier and 98.1% using RF classifier. The indoor room-level location classifier for morning daily living routine dataset accomplished an accuracy of 93.6%.

The second part investigates the usage of the extracted features for understanding and discovery of human patterns. Similarly to the study conducted by Forkan [26], this study uses statistical models for the estimation of human behaviour patterns. However, instead of a Gaussian Distribution, this study uses a KDE to define the probability density function since behaviour features may not follow a Gaussian Distribution as it was verified on the behaviour features acquired during this study. Clustering methods were also used for learning human patterns based on spatial similarity.

Finally, an anomaly detection algorithm was introduced to detect abnormal behaviour, whereas abnormal is recognised by an adaptive threshold according to the distances of the current behaviour to the learned pattern. Experimental results demonstrate the effectiveness of the proposed framework that revealed an increase potential to learn behaviour patterns and detect anomalies considering different use cases. This study may be a key insight for monitoring elderly daily routines as well as marketing analysis, security and tourism management.

5.2 Future Work

Although the developed study revealed promising results there is still room for improvement that can be addressed in the future.

Firstly, considering the current framework, in an online context, methods for the activity spotting of the complex activity classifiers should be implemented.

Furthermore, signals from smartphone sensors present a huge potential and are sensitive enough to perceive human motions and activities, the search for more complex sensory signatures using a combination of different sources of information should be addressed in an unsupervised manner.

For trajectory analysis, the used similarity measure considered the hole trajectory. However, for a more detailed and robust trajectory analysis, the segmented trajectory should be analysed instead.

The distance to the pattern used for anomaly detection can be improved by a weight optimisation process, where more relevant features should have a higher weight. Depending on the type of feature, a weight parametrization should also be addressed. Moreover, new approaches for defining the anomalous threshold may be considered.

Finally, the smartphone used for data acquisition should be replaced by a smartwatch or bracelet containing the sensors embedded in a smartphone. On the end of each day, data should be sent automatically to a server where the developed framework in this study will receive as input the recorded data. Then, the user's behaviour patterns should be learned continuously and any deviation to the normal behaviour should be reported by a notification or text message to the caregiver or the user's doctor.

Bibliography

[1] United Nations. *World Population Ageing 2017 - Highlights (ST/ESA(SER.A/397)*. Tech. rep. New York: Department of Economic and Social Affairs, Population Division, 2017. URL: https://www.un.org/en/development/desa/population/publications/pdf/ageing/WPA2017{_}Highlights.pdf.

[2] C. Patterson. *The state of the art of dementia research : New frontiers World Alzheimer Report 2018*. Tech. rep. London, 2018. URL: https://www.alz.co.uk/research/WorldAlzheimerReport2018.pdf.

[3] E. Jaul and J. Barron. "Age-Related Diseases and Clinical and Public Health implications for the 85 Years Old and Over Population." In: *Frontiers in Public Health* 5 (2017), pp. 1–7. DOI: 10.3389/fpubh.2017.00335.

[4] A. Bradford, M. E. Kunik, P. Schulz, S. P. Williams, and H. Singh. "Missed and Delayed Diagnosis of Dementia in Primary Care: Prevalence and Contributing Factors." In: *Alzheimer Disease & Associated Disorders* 23.4 (2010), pp. 306–314. DOI: 10.1097/WAD.0b013e3181a6bebc.

[5] D. Evans, K. Price, and J. Meyer. "Home Alone With Dementia." In: *SAGE Open* 6.3 (2016), pp. 1–13. DOI: 10.1177/2158244016664954.

[6] J Sayers. "The World Health Report 2001: Mental Health : New Understanding, New Hope." In: *Bull World Health Organ* 79.11 (2001), p. 1085. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2566704/pdf/0042{_}9686{_}79{_}11{_}1085.pdf.

[7] N. Goodwin, J. A. Nelson, F. Ackerman, and T. Weisskopf. *Consumption and the Consumer Society*. 2008.

[8] D. G.S.A. R. Baskaran. "Automated human behavior analysis from surveillance videos : a survey." In: *Springer Science* (2012). DOI: 10.1007/s10462-012-9341-3.

[9] M. Evans, L. A. Maglaras, Y. He, and H. Janicke. "Human Behaviour as an aspect of Cyber Security Assurance." In: *Security and Communication Networks* 9.17 (2016), pp. 4667–4679. DOI: 10.1002/sec.1657.

[10] M. González, C. Hidalgo, and A. Barabási. "Understanding individual human mobility patterns." In: *Nature* 453 (2008), pp. 779–782. DOI: 10.1038/nature06958.

[11] C. A. Ronao and S. B. Cho. "Human activity recognition using smartphone sensors with two-stage continuous hidden markov models." In: *2014 10th International Conference on Natural Computation, ICNC 2014*. IEEE, 2014, pp. 681–686. ISBN: 9781479951505. DOI: 10.1109/ICNC.2014.6975918.

[12] S. Chernbumroong, A. S. Atkins, and H. Yu. "Activity classification using a single wrist-worn accelerometer." In: *SKIMA 2011 - 5th International Conference on Software, Knowledge Information, Industrial Management and Applications*. 2011, pp. 21–25. ISBN: 9781467302463. DOI: 10.1109/SKIMA.2011.6089975.

[13] M. Abreu, M. Barandas, R. Leonardo, and H. Gamboa. "Detailed Human Activity Recognition based on Multiple HMM." In: *Proceedings of the 12th International Joint Conference on Biomedical Engineering Systems and Technologies - Volume 4*. 2019, pp. 171–178. DOI: 10.5220/0007386901710178.

[14] M. Shoaib, S. Bosch, O. D. Incel, H. Scholten, and P. J. Havinga. "Complex human activity recognition using smartphone and wrist-worn motion sensors." In: *Sensors* 16.4 (2016), pp. 1–24. ISSN: 14248220. DOI: 10.3390/s16040426.

[15] M. Er Rida, F. Liu, Y. Jadi, A. A. A. Algawhari, and A. Askourih. "Indoor location position based on bluetooth signal strength." In: *Proceedings - 2015 2nd International Conference on Information Science and Control Engineering, ICISCE 2015*. 2015, pp. 769–773. ISBN: 9781467368506. DOI: 10.1109/ICISCE.2015.177.

[16] V. Guimarães, L. Castro, S. Carneiro, M. Monteiro, T. Rocha, M. Barandas, J. Machado, M. Vasconcelos, H. Gamboa, and D. Elias. "A motion tracking solution for indoor localization using smartphones." In: *2016 International Conference on Indoor Positioning and Indoor Navigation, IPIN 2016*. IEEE, 2016, pp. 1–8. ISBN: 9781509024254. DOI: 10.1109/IPIN.2016.7743680.

[17] G. V. Zàruba, M. Huber, F. A. Kamangar, and I. Chlamtac. "Indoor location tracking using RSSI readings from a single Wi-Fi access point." In: *Springer Science* 13.2 (2007), pp. 221–235. ISSN: 10220038. DOI: 10.1007/s11276-006-5064-1.

[18] R. Leonardo, M. Barandas, and H. Gamboa. "A framework for infrastructure-free indoor localization based on pervasive sound analysis." In: *IEEE Sensors Journal* 18.10 (2018), pp. 4136–4144. ISSN: 1530437X. DOI: 10.1109/JSEN.2018.2817887.

[19] A.K. Jain, R.P.W. Duin, and J. Mao. "Statistical Pattern Recognition: A Review." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22.1 (2000), pp. 4–37. DOI: 10.1109/34.824819.

[20] W. Zheng, R. Zhou, Z. Zhang, Y. Zhong, S. Wang, Z. Wei, and H. Ji. "Understanding the tourist mobility using GPS: How similar are the tourists?" In: *Tourism Management* 71 (2019), pp. 54–66. ISSN: 02615177. DOI: 10.1016/j.tourman.2018.09.019. URL: https://doi.org/10.1016/j.tourman.2018.09.019.

[21] Z. Shou and X. Di. "Similarity analysis of frequent sequential activity pattern mining." In: *Transportation Research Part C: Emerging Technologies* 96 (2018), pp. 122–143. ISSN: 0968090X. DOI: 10.1016/j.trc.2018.09.018. URL: https://doi.org/10.1016/j.trc.2018.09.018.

[22] S. M. Mahmoud, A. Lotfi, and C. Langensiepen. "Occupancy Pattern Extraction and Prediction in an Inhabited Intelligent Environment using NARX Networks." In: *Proceedings - 2010 6th International Conference on Intelligent Environments*. 2010. ISBN: 978-1-4244-7836-1. DOI: 10.1109/IE.2010.18.

[23] K. Farrahi and D. Gatica-Perez. "Discovering routines from large-scale human locations using probabilistic topic models." In: *ACM Transactions on Intelligent Systems and Technology* 2.1 (2011), pp. 1–27. ISSN: 21576904. DOI: 10.1145/1889681.1889684.

[24] V. Chandola, A. Banerjee, and V. Kumar. "Anomaly detection: A survey." In: *ACM Computing Surveys* 41.3 (2009), pp. 1–58. ISSN: 1557-7341. DOI: 10.1145/1541880.1541882.

[25] N. Suzuki, K. Hirasawa, K. Tanaka, Y. Kobayashi, Y. Sato, and Y. Fujino. "Learning motion patterns and anomaly detection by human trajectory analysis." In: *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*. 2007, pp. 498–503. ISBN: 1424409918. DOI: 10.1109/ICSMC.2007.4413596.

[26] A. Rahim, M. Forkan, I. Khalil, Z. Tari, S. Foufou, and A. Bouras. "A context-aware approach for long-term behavioural change detection and abnormality prediction in ambient assisted living." In: *Pattern Recognition* 48.3 (2015), pp. 628–641. DOI: 10.1016/j.patcog.2014.07.007.

[27] S Tomforde, T Dehling, R Haux, D Huseljic, D Kottke, J Scheerbaum, B Sick, A Sunyaev, and K. H. Wolf. "Towards Proactive Health-enabling Living Environments: Simulation-based Study and Research Challenges." In: *31th International Conference on Architecture of Computing Systems*. 2018, pp. 1–8. ISBN: 9783800745593.

[28] B. F. Skinner. *Science and Human Behavior*. Simon and Schuster, 1953, p. 449.

[29] M. D. Lezak, D. B. Howieson, D. W. Loring, H. J. Hannay, and J. S. Fischer. *Neuropsychological Assessment*. Fourth. Oxford University Press, 2004, pp. 18–38. ISBN: 9780195395525.

[30] J. W. Berry, Y. H. Poortinga, M. H. Segall, and P. R. Dasen. *Cross-cultural Psychology*. Second. Vol. 22. 1. Cambridge University Press, 2002, pp. 10–15. ISBN: 9780521641524. DOI: 10.1177/0022022191221002.

[31] R. E. Wray and J. E. Laird. "Variability in human behavior modeling for military simulations." In: *Behavior Representation in Modeling and Simulation Conference*. Vol. 1001. 48109. 2003, pp. 2953–2960. DOI: 10.1.1.78.2699. URL: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.78.2699.

[32] N. Magdy, M. A. Sakr, T. Mostafa, and K. El-Bahnasy. "Review on Trajectory similarity measures." In: *2015 IEEE Seventh International Conference on Intelligent Computing and Information Systems*. Vol. 7. 1. 2015, pp. 613–619. DOI: 10.1145/2782759.2782767.

[33] J.-R. Hwang, H.-Y. Kang, and K.-J. Li. "Searching for Similar Trajectories on Road Networks using Spatio-Temporal Similarity." In: *Lecture Notes in Computer Science*. 2006, pp. 282–295. DOI: 10.1007/11827252_22.

[34] D. Zhang. "Periodic Pattern Mining from Spatio-temporal Trajectory Data." Doctoral dissertation. James Cook University, 2018. URL: https://doi.org/10.25903/5c89cdf41ca5b.

[35] L. Atallah and G. Z. Yang. "The use of pervasive sensing for behaviour profiling - a survey." In: *Pervasive and Mobile Computing* 5.5 (2009), pp. 447–464. ISSN: 15741192. DOI: 10.1016/j.pmcj.2009.06.009. URL: http://dx.doi.org/10.1016/j.pmcj.2009.06.009.

[36] J. G. Lee, J. Han, and X. Li. "Trajectory Outlier Detection: A Partition-and-Detect Framework." In: *2008 IEEE 24th International Conference on Data Engineering*. 2008, pp. 140–149. ISBN: 9781424418374. DOI: 10.1109/ICDE.2008.4497422.

[37] Y. Wang, K. Qin, Y. Chen, and P. Zhao. "Detecting Anomalous Trajectories and Behavior Patterns Using Hierarchical Clustering from Taxi GPS Data." In: *ISPRS International Journal of Geo-Information* 7.1 (2018), p. 25. ISSN: 2220-9964. DOI: 10.3390/ijgi7010025. URL: http://www.mdpi.com/2220-9964/7/1/25.

[38] E. M. Knorr, R. T. Ng, and V. Tucakov. "Distance-based outliers : algorithms and applications." In: *The VLDB Journal* 8.3-4 (2000), pp. 237–253. DOI: https://doi.org/10.1007/s007780050006.

[39] A. Bayat, M. Pomplun, and D. A. Tran. "A Study on Human Activity Recognition Using Accelerometer Data from Smartphones." In: *Procedia - Procedia Computer Science* 34 (2014), pp. 450–457. ISSN: 1877-0509. DOI: 10.1016/j.procs.2014.07.009. URL: http://dx.doi.org/10.1016/j.procs.2014.07.009.

[40] O. Gani, A. K. Saha, G. Mushih, T. Ahsan, and R. O. Smith. "A Novel Framework to Recognize Complex Human Activity." In: *2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC)* (2017), pp. 948–956.

[41] C. D. R Grotz, Devin R Cash, Paul S. Katz. "Progress in the development of the index of ADL." In: *Gerontologist* 10.1 (1970), pp. 20–30. DOI: 10.1093/geront/10.1_part_1.20.

[42] K. R. Westerterp. "Assessment of physical activity : a critical appraisal." In: *European Journal of Applied Physiology* 105.6 (2009), pp. 823–828. DOI: 10.1007/s00421-009-1000-2.

[43] J. K. Aggarwal and M. S. Ryoo. "Human Activity Analysis : A Review." In: *Acm* 5 (2007). DOI: 10.1145/1922649.1922653.

[44] T. L. Van Kasteren, G. Englebienne, and B. J. Kröse. "An activity monitoring system for elderly care using generative and discriminative models." In: *Personal and Ubiquitous Computing* 14.6 (2010), pp. 489–498. ISSN: 16174909. DOI: 10.1007/s00779-009-0277-9.

[45] Ó. D. Lara and M. A. Labrador. "A survey on human activity recognition using wearable sensors." In: *IEEE Communications Surveys and Tutorials* 15.3 (2013), pp. 1192–1209. ISSN: 1553877X. DOI: 10.1109/SURV.2012.110112.00192.

[46] H. F. Nweke, Y. W. Teh, M. A. Al-garadi, and U. R. Alo. "Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: State of the art and research challenges." In: *Expert Systems with Applications* 105 (2018), pp. 233–261. ISSN: 09574174. DOI: 10.1016/j.eswa.2018.03.056. URL: https://doi.org/10.1016/j.eswa.2018.03.056.

[47] V. M. A. Souza. "Asphalt pavement classification using smartphone accelerometer and Complexity Invariant Distance." In: *Engineering Applications of Artificial Intelligence* 74 (2018), pp. 198–211. ISSN: 0952-1976. DOI: 10.1016/j.engappai.2018.06.003. URL: https://doi.org/10.1016/j.engappai.2018.06.003.

[48] D. ROYSTER. *Global positioning system (GPS)*. 2018.

[49] N. Hernández, M. Ocaña, J. M. Alonso, and E. Kim. "WiFi-based indoor localization and tracking of a moving device." In: *2014 Ubiquitous Positioning Indoor Navigation and Location Based Service, UPINLBS 2014 - Conference Proceedings* (2015), pp. 281–289. DOI: 10.1109/UPINLBS.2014.7033738.

[50] M. E. Rida, F. Liu, Y. Jadi, A. Ali, A. Algawhari, and A. Askourih. "Indoor location position based on Bluetooth Signal Strength." In: *2015 2nd International Conference on Information Science and Control Engineering*. Shanghai: IEEE, 2015, pp. 769–773. ISBN: 9781467368506. DOI: 10.1109/ICISCE.2015.177.

[51] H. Koyuncu and S. H. Yang. "A survey of indoor positioning and object locating systems." In: *International Journal of Computer Science and Network Security* 10.5 (2010), pp. 121–128. URL: http://paper.ijcsns.org/07{_}book/201005/20100518.pdf.

[52] A. El-Rabbany. *Introduction to GPS - The Global Positioning System*. 2002, pp. 1–9.

[53] *GPS Accuracy*. URL: https://www.gps.gov/systems/gps/performance/accuracy/ (visited on 06/09/2019).

[54] R. Hantoush. "Evaluating WI-FI indoor positioning approaches In a real world environment." Master Thesis. Universidade Nova de Lisboa, 2016, p. 8.

[55] H. Liu, H. Darabi, P. Banerjee, and J. Liu. "Survey of wireless indoor positioning techniques and systems." In: *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews* 37.6 (2007), pp. 1067–1080. ISSN: 10946977. DOI: 10.1109/TSMCC.2007.905750.

[56] B. Viel and M. Asplund. "Why is fingerprint-based indoor localization still so hard?" In: *2014 IEEE International Conference on Pervasive Computing and Communication Workshops, PERCOM WORKSHOPS 2014* (2014), pp. 443–448. DOI: 10.1109/PerComW.2014.6815247.

[57] A. R. Pratama, Widyawan, and R. Hidayat. "Smartphone-based Pedestrian Dead Reckoning as an indoor positioning system." In: *Proceedings of the 2012 International Conference on System Engineering and Technology, ICSET 2012* December 2014 (2012). DOI: 10.1109/ICSEngT.2012.6339316.

[58] H. Xie, T. Gu, X. Tao, H. Ye, and J. Lv. "MaLoc: A Practical Magnetic Fingerprinting Approach to Indoor Localization using Smartphones." In: *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (2014), pp. 243–253. ISSN: 1536-5964. DOI: 10.1145/2632048.2632057. URL: http://dl.acm.org/citation.cfm?doid=2632048.2632057.

[59] R. Leonardo. "Contextual Information based on Pervasive Sound Analysis." Master Thesis. Universidade Nova de Lisboa, 2017, p. 69.

[60] M. B. del Rosario, S. J. Redmond, and N. H. Lovell. "Tracking the evolution of smartphone sensing for monitoring human movement." In: *Sensors* 15.8 (2015), pp. 18901–18933. ISSN: 14248220. DOI: 10.3390/s150818901.

[61] R. Santos. "Human Crowdsourcing Data for Indoor Location Applied to Ambient Assisted Living Scenarios." Master Thesis. Universidade NOVA de Lisboa, 2018.

[62] T. M. Mitchell. *Machine Learning*. Ed. by WCB/McGraw-Hill. McGraw-Hill Science/Engineering/Math, 1997. ISBN: 0070428077 9780070428072.

[63] H. Cardoso and J. Mendes-Moreira. "Improving human activity classification through online semi-supervised learning." In: *CEUR Workshop Proceedings* 2069 (2017), pp. 1–12. ISSN: 16130073.

[64] O. Banos, J. M. Galvez, M. Damas, H. Pomares, and I. Rojas. "Window size impact in human activity recognition." In: *Sensors (Switzerland)* 14.4 (2014), pp. 6474–6499. ISSN: 14248220. DOI: 10.3390/s140406474.

[65] G. Wang, Q. Li, L. Wang, W. Wang, M. Wu, and T. Liu. "Impact of sliding window length in indoor human motion modes and pose pattern recognition based on smartphone sensors." In: *Sensors (Switzerland)* 18.6 (2018). ISSN: 14248220. DOI: 10.3390/s18061965.

[66] N. Kern, B. Schiele, and A. Schmidt. "Multi-sensor Activity Context Detection for Wearable Computing." In: *Lecture Notes in Computer Science*. 2003, pp. 220–232. ISBN: 9783540398639. DOI: 10.1007/978-3-540-39863-9_17.

[67] A. Mannini, M. Rosenberger, W. L. Haskell, A. M. Sabatini, and S. S. Intille. "Activity recognition in youth using single accelerometer placed at wrist or ankle." In: *Medicine and Science in Sports and Exercise* 49.4 (2017), pp. 801–812. ISSN: 15300315. DOI: 10.1249/MSS.0000000000001144.

[68] N. V. Chawla, N. Japkowicz, and A. Kotcz. "Editorial." In: *ACM SIGKDD Explorations Newsletter* 6.1 (2007), p. 1. ISSN: 19310145. DOI: 10.1145/1007730.1007733.

[69] S. Vluymans. "Learning from imbalanced data." In: *Studies in Computational Intelligence* 807.9 (2019), pp. 81–110. ISSN: 1860949X. DOI: 10.1007/978-3-030-04663-7_4.

[70] P. Foster. "Machine learning from imbalanced data sets 101." In: *Proceedings of the AAAI'2000 workshop on imbalanced data sets* (2000), pp. 1–3.

[71] N. Chumerin and M. M. Van Hulle. "Comparison of two feature extraction methods based on maximization of mutual information." In: *Proceedings of the 2006 16th IEEE Signal Processing Society Workshop on Machine Learning for Signal Processing, MLSP 2006*. May. 2007, pp. 343–348. ISBN: 1424406560. DOI: 10.1109/MLSP.2006.275572.

[72] M. A. Hall and L. A. Smith. "Feature Selection for Machine Learning : Comparing a Correlation-based Filter Approach to the Wrapper." In: *Proceedings of the Twelfth International FLAIRS Conference*. 1999, p. 5. ISBN: 1-57735-080-4.

[73] S. Khalid, T. Khalil, and S. Nasreen. "A survey of feature selection and feature extraction techniques in machine learning." In: *Proceedings of 2014 Science and Information Conference, SAI 2014*. 2014, pp. 372–378. ISBN: 9780989319317. DOI: 10.1109/SAI.2014.6918213.

[74] S. O.N. G. Yan-yan and L. U. Ying. "Decision tree methods: applications for classification and prediction." In: *Shanghai Archives of Psychiatry* 27.2 (2015), pp. 130–135. ISSN: 1002-0829. URL: http://www.ncbi.nlm.nih.gov/pmc/articles/PMC4466856/.

[75] P. Cunningham and S. J. Delany. "k-Nearest Neighbour Classifiers." In: *Multiple Classifier Systems* 34 (2007), pp. 1–17.

[76] A. Cutler, D. R. Cutler, and J. R. Stevens. "Random forests." In: *Ensemble Machine Learning: Methods and Applications*. 2012, pp. 157–175. ISBN: 9781441993267. DOI: 10.1007/978-1-4419-9326-7_5.

[77] T. Hastie, S. Rosset, J. Zhu, and H. Zou. "Multi-class AdaBoost." In: *Statistics and Its Interface* 2.3 (2013), pp. 349–360. ISSN: 19387989. DOI: 10.4310/sii.2009.v2.n3.a8.

[78] W. Chen, X. Wei, and K. X. Zhu. "Engaging Voluntary Contributions in Online Communities: A Hidden Markov Model." In: *MIS Quarterly* 42.1 (2018), pp. 83–100. ISSN: 02767783. DOI: 10.25300/MISQ/2018/14196. URL: https://misq.org/engaging-voluntary-contributions-in-online-communities-a-hidden-markov-mode.html.

[79] J Bulla. "Application of hidden Markov models and hidden semi-Markov models to financial time series." PhD. Georg-August-University of Gottingen, 2006, p. 157. URL: http://mpra.ub.uni-muenchen.de/7675.

[80] Z. Ghaharamani. "An Introduction to Hidden Markov Models and Bayesian Networks." In: *International Journal of Pattern Recognition and Artificial Intelligence* 15.1 (2001), pp. 9–42. DOI: https://doi.org/10.1142/S0218001401000836.

[81] L. R. Rabiner. "A Tuturial on Hidden Markov Models and Selected Applications in Speech Recognition." In: *Proceedings of the IEEE* 77.2 (1989), pp. 257–286. DOI: 10.1109/5.18626.

[82] P. Rai and S. Singh. "A Survey of Clustering Techniques." In: *International Journal of Computer Applications* 7.12 (2010), pp. 1–5. ISSN: 09758887. DOI: 10.5120/1326-1808. URL: http://www.ijcaonline.org/volume7/number12/pxc3871808.pdf.

[83] L. Rokach and O. Maimon. "Clustering Methods: Chapter 15." In: *Data mining and knowledge discovery handbook*. 2005, pp. 321–352.

[84] V. Kumar, P.-T. Tan, and M. Steinbach. "Cluster analysis: basic concepts and algorithms." In: *Introduction to data mining*. 2006, pp. 488–568. ISBN: 0321321367.

[85] M. Rosenblatt. "A Central Limit Theorem and a Strong Mixing Condition." In: *Proceedings of the National Academy of Sciences* 42.1 (1956), pp. 43–47. ISSN: 0027-8424. DOI: 10.1073/pnas.42.1.43.

[86] E Parzen. "On the Estimation of Probability Density Functions and Mode." In: *Ann. Math. Statist* 33 (1962), pp. 1065–1076. DOI: doi:10.1214/aoms/1177704472.

[87] S. Węglarczyk. "Kernel density estimation and its application." In: *ITM Web of Conferences*. Vol. 23. 2018. DOI: 10.1051/itmconf/20182300037.

[88] B. A. Turlach. "Bandwidth selection in kernel density estimation: A review." In: *Handbook of Systemic Autoimmune Diseases* February 1999 (1993), pp. 23–493. URL: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.44.6770.

[89] B. W. Silverman. "Kernel density estimation using the fast Fourier transform." In: *Applied Statistics* 31.1 (1982), pp. 93–99. DOI: 10.2307/2347084.

[90] S. B. Kotsiantis, I. D. Zaharakis, and P. E. Pintelas. "Supervised Machine Learning : A Review of Classification Techniques General Issues of Supervised Learning Algorithms." In: *Informatica* 31.3 (2007), pp. 249–268. ISSN: 03505596.

[91] R Kohavi. "A study of cross-validation and bootstrap for accuracy estimation and model selection." In: *Proceedings of the 14th international joint conference on Artificial intelligence - Volume 2* 2.0 (1995), pp. 1137–1143.

[92] E. Alpaydin. *Introduction to Machine Learning Second Edition*. 2010, pp. 350–380. ISBN: 9780262012430. DOI: 10.1007/978-1-62703-748-8_7. arXiv: 0904.3664.

[93] A. S. Furtado, L. L. Pilla, and V. Bogorny. "A branch and bound strategy for Fast Trajectory Similarity Measuring." In: *Data and Knowledge Engineering* 115 (2018), pp. 16–31. ISSN: 0169023X. DOI: 10.1016/j.datak.2018.01.003. URL: https://doi.org/10.1016/j.datak.2018.01.003.

[94] M. Meinard. "Dynamic Time Warping." In: *Information retrieval for music and motion*. Berlin: Springer-Verlag, 2007. Chap. chapter 4, pp. 69–84. ISBN: 978-3-540-74048-3. DOI: https://doi.org/10.1007/978-3-540-74048-3_4.

[95] A. Xavier and F. Miquel. "Memory efficient subsequence DTW for query-by-example spoken term detection." In: *2013 IEEE International Conference on Multimedia and Expo (ICME)*. 2013. DOI: 10.1109/ICME.2013.6607546.

[96] J. Gomez-Gil, R. Ruiz-Gonzalez, S. Alonso-Garcia, and F. J. Gomez-Gil. "A Kalman filter implementation for precision improvement in Low-Cost GPS positioning of tractors." In: *Sensors (Switzerland)* 13.11 (2013), pp. 15307–15323. ISSN: 14248220. DOI: 10.3390/s131115307.

[97] C. Figueira. "Body Location Independent Activity Monitoring." Master Thesis. New University of Lisbon, 2015.

[98] A. Jović, K. Brkić, and N. Bogunović. "A review of feature selection methods with applications." In: *2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2015 - Proceedings* (2015), pp. 1200–1205. DOI: 10.1109/MIPRO.2015.7160458.

[99] M. Kheirkhahan, A. Chakraborty, A. A. Wanigatunga, D. B. Corbett, T. M. Manini, and S. Ranka. "Wrist accelerometer shape feature derivation methods for assessing activities of daily living." In: *BMC Medical Informatics and Decision Making* 18.Suppl 4 (2018), pp. 1–13. ISSN: 14726947. DOI: 10.1186/s12911-018-0671-1.

[100] T. Chuk, K. Crookes, W. G. Hayward, A. B. Chan, and J. H. Hsiao. "Hidden Markov model analysis reveals the advantage of analytic eye movement patterns in face recognition across cultures." In: *Cognition* 169 (2017), pp. 102–117. ISSN: 18737838. DOI: 10.1016/j.cognition.2017.08.003. URL: http://dx.doi.org/10.1016/j.cognition.2017.08.003.

[101] D. Lederman and J. Tabrikian. "Classification of multichannel EEG patterns using parallel hidden markov models." In: *Medical and Biological Engineering and Computing* 50.4 (2012), pp. 319–328. ISSN: 01400118. DOI: 10.1007/s11517-012-0871-2.

[102] K. R. Sungkono and R. Sarno. "Patterns of Fraud Detection Using Coupled Hidden Markov Model." In: *3rd International Conference on Science in Information Technology (ICSITech)*. 2017, pp. 235–240. ISBN: 9781509058648. DOI: 10.1109/ICSITech.2017.8257117.

[103] S. Jeebun, R. Ballgobin, and T. Al-ani. "Optimal Number of States in Hidden Markov Models and its Application to the Detection of Human Movement." In: *University of Mauritius Research Journal* 21 (2015), pp. 438–469. ISSN: 1694-0342.

[104] G. Harari, N. D. Lane, R. Wang, B. S. Crosier, A. T. Campbell, and S. D. Gosling. "Using Smartphones to Collect Behavioral Data in Psychological Science: Opportunities, Practical Considerations and Challenges." In: *Perspectives on Psychological Science* 11.6 (2016), pp. 838–854. DOI: 10.1177/1745691616650285.

[105] J. Boase. "Implications of softwarebased mobile media for social research." In: *Mobile Media and Communication* 1.1 (2013), pp. 57–62. ISSN: 20501587. DOI: 10.1177/2050157912459500.

[106] G. Ogris, T. Stiefmeier, P. Lukowicz, and G. Tröster. "Using a complex multimodal on-body sensor system for activity spotting." In: *Proceedings - International Symposium on Wearable Computers, ISWC* (2008), pp. 55–62. ISSN: 15504816. DOI: 10.1109/ISWC.2008.4911585.

Appendix A

TSFEL Features Description

Table A.1: Temporal domain features implemented in TSFEL.

Temporal Domain Features	
Autocorrelation	Computes the correlation of the signal with itself shifted over successive time intervals.
Zero Crossing rate	Number of times the signal changes from positive to negative and vice-versa.
Mean absolute difference	Computes the mean of the absolute difference between consecutive signal values.
Median absolute difference	Computes the median of the absolute difference between consecutive signal values.
Sum of absolute differences	Calculates the sum of the absolute differences.
Mean difference	Calculates the mean difference of a signal between consecutive signal values.
Median difference	Calculates the median difference of a signal between consecutive signal values.
Distance	Computes the total distance travelled by the signal by approximating the distance between two points to the hipotenusa.
Centroid	Calculates the arithmetic mean position of all signals point positions.
Maximum peaks	Returns the total number of positive peaks of a dataset.
Minimum peaks	Returns the total number of minimum peaks of a dataset.

Linear regression	Computes the linear regression of a dataset.
Total Energy	Computes the total power of a signal by the fraction between sum of the powered signal and the signals duration.

Table A.2: Spectral domain features implemented in TSFEL.

Spectral Domain Features	
Curve distance	Returns the distance of the signal's cumulative sum FFT magnitude elements to the respective linear regression.
Fundamental Frequency	Computes the fundamental frequency, which is the frequency at 0 Hz.
Maximum Frequency	Computes the maximum frequency of a signal.
Median Frequency	Computes the median frequency of a signal.
Maximum power of the spectrum	Computes the maximum power spectrum density of a signal.
MFCC	Computes the MFCC.
Spectral centroid	Calculates the weighted mean of the frequency values positions of a signal.
Spectral decrease	Computes the spectral decrease of a dataset through linear regression.
Spectral kurtosis	Computes the flatness of a spectral distribution around the mean value.
Spectral maximum peaks	Calculates the number positive spectral peaks of a dataset.
Spectral roll-off	Computes the frequency where 95% of the energy is contained bellow this value.
Spectral roll-on	Computes the frequency where 5% of the energy is contained bellow this value.
Spectral skewness	Measures the asymmetry of a spectral distribution around its mean value.
Spectral slope	Computes the linear regression of the spectral amplitude.
Spectral spread	Computes the spread of the spectrum around its mean value.
Spectral variation	Computes the amount of variation of the spectrum along time through the normalized cross correlation between two consecutive spectral amplitude values.
Total Energy	Computes the total energy by the fraction between sum of the powered spectrum and the signals length.

Table A.3: Statistical domain features implemented in TSFEL.

Statistical Domain Features	
Histogram	Computes the histogram of a signal.
Interquartile range	Computes the difference between the upper and lower quartile.
Kurtosis	Measures the flatness of the signal distribution around the mean value.
Maximum	Calculates the maximum value of a signal.
Mean	Calculates the mean value of a signal.
Mean absolute deviation	Computes the mean absolute deviation of a signal.
Median	Calculates the median value of a signal.
Median absolute deviation	Computes the mean absolute deviation of a signal.
Minimum	Calculates the minimum value of a signal.
Root mean square	Computes the square root of the arithmetic mean of the squares of the original values.
Skewness	Computes the asymmetry of the signal distribution.
Variance	Computes the variance of the dataset.
Standard Deviation	Computes the standard deviation of the dataset.

Appendix B

Informed Consent to Participants

The following page comprises a portuguese version of the informed consent that was filled by the participants that enroll in this study. The informed consent contains also information about the collected data. The form was printed in duplicates so that one copy is kept by the participant.

Fraunhofer
PORTUGAL

www.fraunhofer.pt
Rua Alfredo Allen 455/461
4200-135 Porto, PORTUGAL

CONSENTIMENTO PARA PARTICIPAÇÃO EM INVESTIGAÇÃO

A *Associação Fraunhofer Portugal Research* faz trabalho de investigação destinado a encontrar soluções tecnológicas que promovam o bem-estar da população.

A Faculdade de Ciências e Tecnologias da Universidade Nova de Lisboa em parceria com a *Associação Fraunhofer Portugal Research* pretendem realizar um estudo de investigação, no âmbito da dissertação de Mestrado em Engenharia Biomédica, sobre a aprendizagem dos padrões do comportamento humano através da análise da trajetória e do reconhecimento automático de atividades.

Para este estudo, propomos a S/ participação na realização de recolhas de dados da sua rotina diária. Para isso iremos proceder à recolha de informação relativa aos seguintes sensores presentes nos telemóveis através da instalação de uma aplicação Android: Acelerómetro, Giroscópio, Magnetómetro, Barómetro, Microfone, Wi-Fi e Localização. Estes dados são depois anonimizados e usados para definir e criar uma proposta de solução para a aplicação acima referida.

Gostaríamos de contar com a sua participação. A participação não envolve qualquer prejuízo ou dano material e não haverá lugar a qualquer pagamento. Os dados recolhidos são confidenciais e serão anonimizados para tratamento.

Serão tomadas todas as medidas necessárias à salvaguarda e proteção dos dados recolhidos por forma a evitar que venham a ser acedidos por terceiros não autorizados. A sua participação é voluntária, podendo em qualquer altura cessá-la sem qualquer tipo de consequência. Também poderá pedir a retificação ou destruição da informação recolhida a qualquer momento.

Agradecemos muito o seu contributo, fundamental para a nossa investigação!

O participante:

Declaro ter lido e compreendido este documento, bem como as informações verbais fornecidas e aceito participar nesta investigação. Permito a utilização dos dados que forneço de forma voluntária, confiando que apenas serão utilizados para investigação e com as garantias de confidencialidade e anonimato que me são dadas pelo investigador.

Nome: _____

Assinatura: _____ Data ___ / ___ / _____

Investigador responsável:

Nome: Letícia Fernandes

Assinatura:_____

Telefone: 220 430 360

E-mail: leticia.fernandes@fraunhofer.pt

ESTE DOCUMENTO É FEITO EM DUPLICADO: UM PARA O PARTICIPANTE E OUTRO PARA O INVESTIGADOR.