Predictive Models of Quality of Life from Biometric Sensors

Anaís Silva Dias

MASTER’S DISSERTATION

Mestrado Integrado em Engenharia Informática e Computação

Supervisor: Pedro Gonçalo Ferreira Alves Nogueira

June 27, 2017
Predictive Models of Quality of Life from Biometric Sensors

Anaís Silva Dias

Mestrado Integrado em Engenharia Informática e Computação

June 27, 2017
Abstract

With the improvement of health care and living conditions, there has been a decrease in mortality, accompanied by an increase in life expectancy of the overall population. This increase is also associated with a rise in chronic illnesses, which can induce a great deal of pain and discomfort. Therefore, it is important to not only focus on the length of a patient’s life, but also on its quality. Even though Quality of Life (QoL) is now a crucial measurement in the treatment of patients, the methods to assess it are not as evolved as necessary. There is a lack of continuous evaluation, which is the most effective way of assessing QoL.

Stemming from the need to predict and monitor QoL with greater precision, we aim to develop predictive models based on physiological data obtained through biometric sensors, by simulating a wide variety of this data and attributing QoL scores to it, depending on their features, and then training models with the resulting data. This will be done within the scope of the QVida+ research project, which aims to create an information system that will use physiological data from patients to evaluate QoL, and that will adapt continuously to the patient and serve as a decision support system for health care professionals.

In order to achieve the best results, several biometric measurements will be analyzed. We will have in account both the precision of their output, but also their practicality for the patient, due to the need for constant wear. A large set of physiological data will be generated, with a great variation, in order to allow for extensive testing and tuning of the predictive models. This data will be labeled in terms of QoL, according to its features. Then, we will compare the possible methods of classification of the data, in order to achieve predictive models that will be precise and generalizable to a larger population, and that will allow a continuous measurement without significant effort required from the patient. Some of the techniques that will be tested in this step are random forests and deep learning.

The possibility of assessing QoL continuously can have a significant impact in health care, serving as a support to clinical decision, as it provides a highly important and reliable measurement, benefiting patients and professionals alike.
Resumo

Como resultado da melhoria dos cuidados de saúde e das condições de vida, tem havido um decréscimo na mortalidade, acompanhado por um aumento da expectativa de vida da população. Este aumento vem seguido de uma subida na incidência de doenças crónicas, que podem causar muitas dores e desconforto aos seus portadores. É, por isso, importante haver um foco não só na longevidade do paciente, mas também na sua Qualidade de Vida (QdV). Apesar da QdV ser atualmente uma medida crucial no tratamento de doentes, os métodos para a avaliar não são suficientemente desenvolvidos, havendo uma falta de métodos de avaliação contínua, que é a forma mais eficiente de avaliar a QdV.

Este projeto origina da necessidade de monitorização e previsão de QdV com maior precisão, e tem como objetivo o desenvolvimento de modelos de previsão baseados em dados fisiológicos obtidos através de sensores biométricos. Este desenvolvimento será feito ao simular uma grande variedade de dados e atribuir pontuações de QdV aos mesmos, dependendo das suas características. Os modelos de previsão serão de seguida treinados com os dados e as pontuações resultantes. O desenvolvimento será concretizado no âmbito do projeto europeu QVida+, que tenciona criar um sistema de informação que usa dados fisiológicos de pacientes para avaliar a QdV, e que se adapta continuamente ao paciente e que servirá como um sistema de apoio à decisão para os profissionais de saúde.

De forma a obter os melhores resultados, serão analisadas várias medidas biométricas, tendo em conta tanto a precisão das mesmas, como a sua praticidade de medição para o paciente, devendo à necessidade de utilização constante da ferramenta de medição. De seguida, será gerado um grande conjunto de dados fisiológicos, com uma grande variação, de forma a permitir um treino e afinação extenso dos modelos de previsão. Os dados serão categorizados em termos de QdV de acordo com as suas características. Finalmente, será feita uma comparação de possíveis métodos de classificação dos dados, com o objetivo de obter modelos de previsão precisos e generalizáveis numa população de maior dimensão, que permitam uma medição contínua sem exigirem um esforço significativo do paciente. Nesta fase, serão utilizadas técnicas como random forests e deep learning.

A possibilidade de avaliar QdV continuamente terá um impacto significativo nos cuidados de saúde, servindo como apoio a decisões clínicas, ao providenciar uma medida de saúde altamente importante e fiável, o que beneficiará tanto pacientes como profissionais.
Acknowledgements

This dissertation would not have been possible without the help of several people.

First, I would like to thank my mother, for making sure, through all her sacrifices, that I would get the best education possible. For supporting me every step of the way, and for being the best role model of resilience I could ask for.

I would like to thank my supervisor Pedro Nogueira for the support and guidance, and most of all, for the helpful tips which helped me overcome the several obstacles in the rocky road that was this dissertation process.

Finally, I would like to thank all my friends for their uplifting effect on my mood. In particular, I would like to thank my boyfriend, João Ramos, for his unconditional support and encouragement and for believing in me more than I do myself. I would also like to thank Vânia Leite, for being my Starbucks work buddy, my best friend, and a patient listener to my stress-fueled rants.

Anaís Dias
“Daring ideas are like chessmen moved forward. They may be beaten, but they may start a winning game”

Johann Wolfgang von Goethe
# Contents

1 Introduction ................................. 1
   1.1 Context ........................................ 1
   1.2 Motivation and Goals ...................... 2
   1.3 Dissertation Structure ................... 3

2 Concepts and Related Work ................. 5
   2.1 Introduction .................................... 5
   2.2 Traditional Methods ......................... 6
       2.2.1 Computerized assessment ................. 8
   2.3 Sensor based assessment ..................... 8
       2.3.1 Physiological data ....................... 9
       2.3.2 Wearable Sensors ......................... 9
   2.4 Prediction from physiological data ........ 10
       2.4.1 Relating physiological data with health and lifestyle 10
       2.4.2 Algorithms .................................. 11
       2.4.3 Methods used in studies .................. 16
   2.5 Related work and similar approaches ...... 16
   2.6 Conclusions .................................... 17

3 Implementation of proposed solution ...... 19
   3.1 Tools ........................................... 20
   3.2 Architecture of proposed solution ........ 20
   3.3 Signal transformation tool .................. 21
   3.4 Data simulation ............................... 23
   3.5 Data labeling ................................... 24
       3.5.1 Randomly labeled data .................... 25
       3.5.2 Data labeled from features ............... 25
   3.6 Modeling and clustering .................... 26
       3.6.1 Tools and packages used ................. 26
       3.6.2 Methodology ................................ 27

4 Results and Discussion .................... 31
   4.1 Data simulation ............................... 31
   4.2 Data labeling ................................... 32
   4.3 Clustering of simulated data ............... 33
   4.4 Classification of randomly labeled data .... 35
   4.5 Classification of data with labels extracted from features .............. 37
       4.5.1 Without feature selection ................. 37
CONTENTS

4.5.2 With feature selection .................................................. 37
4.6 Discussion ......................................................................... 39

5 Conclusions and Future Work ........................................... 43
5.1 Results .............................................................................. 43
5.2 Study limitations .............................................................. 44
5.3 Satisfaction of goals .......................................................... 44
5.4 Suggestions for future work .............................................. 44

References .............................................................................. 45

A Further statistics of simulated data ..................................... 49
B Selected features ................................................................. 51
C Images of developed tools .................................................. 53
## List of Figures

2.1 Example of a Bayesian network ........................................... 13
2.2 Example of a decision tree ............................................... 14
3.1 Architecture of the solution ............................................. 21
4.1 Dendrogram resulting from hierarchical clustering of simulated data ................................................ 35
C.1 Interface of signal transformation tool, used for transformation of data, visualization and data generation ........................................ 53
C.2 Interface of signal transformation tool, including filters and log of changes ........................................ 53
List of Tables

2.1 Domains and Items of HRQoL ........................................... 6
2.2 Strengths and weaknesses of HRQoL measurement instruments .......... 7
2.3 Physiological metrics and their applications ................................ 9

4.1 Statistics of main overall features in simulated data ....................... 32
4.2 Distribution of QoL labels in simulated dataset ............................. 33
4.3 Statistics on QoL labels of simulated data .................................. 33
4.4 Clustering results with 3 clusters for simulated data ......................... 33
4.5 Clustering results with 4 clusters for simulated data ......................... 33
4.6 Clustering results with 5 clusters for simulated data ......................... 34
4.7 Distribution of instances in clusters for hierarchical clustering .......... 34
4.8 Statistics of most noteworthy features for cluster 1 resulting from hierarchical clustering .................................................. 34
4.9 Statistics of most noteworthy features for cluster 2 resulting from hierarchical clustering .................................................. 34
4.10 Statistics of most noteworthy features for cluster 3 resulting from hierarchical clustering .................................................. 35
4.11 Ideal values for classification performance metrics ......................... 36
4.12 Classification results for randomly labeled data with entropy 0 ............. 36
4.13 Classification results for randomly labeled data with entropy 0.5, averaged from 5 runs .............................................................. 36
4.14 Classification results for randomly labeled data with entropy 1 averaged from 5 runs .............................................................. 37
4.15 Classification results for data labeled from features, without feature selection .............................................................. 37
4.16 Classification results for data labeled from features, with feature selection, using 23 features .............................................................. 38

A.1 Statistics of overall features of simulated data ............................. 50
Abbreviations

QoL   Quality of Life
HRQoL Health-Related Quality of Life
WHO   World Health Organization
bpm   Beats per minute
ECG   Electrocardiography
HR    Heart rate
HRV   Heart Rate Variability
ANS   Autonomic Nervous System
PAM   Partitioning Around Medoids
SVM   Support Vector Machines
UPGMA Unweighted Pair Group Method with Arithmetic Mean
APN   Average proportion of non-overlap
AD    Average distance
ADM   Average distance between means
FOM   Figure of merit
AUC   Area under the curve
ROC   Receiver operating characteristic
Chapter 1

Introduction

The assessment of quality of life has increasingly become more relevant for health care and is now seen as a medical goal. This has prompted the identification of a need to monitor QoL more constantly and accurately, which is the motivation behind this project. We aim to create predictive models using simulated physiological data, specifically heart rate and RR interval data, that will precisely estimate QoL and demonstrate the feasibility of predicting QoL with machine learning models using real patient data. The present chapter will give an overview of the context of this project, describe the motivation behind it and which goals it should achieve and will give a summary of its structure.

1.1 Context

This dissertation is inserted in the area of health informatics, which aims to improve health care through the use of information systems. This has been shown to improve efficiency and quality of care. Despite this, the adoption of informatics in this field has grown slowly, and it requires the involvement of all relevant participants, for a successful implementation [FGRR15].

This work will also be developed within the scope of the QVida+ research project [RFG+16]. This project aims to create an integrated information system that will continuously monitor patients and provide information about their health and QoL. The monitorization will be done through biometric devices, both personal and environmental. The resulting data will be used to predict patients’ QoL, and to provide health professionals with important information to help their decision process, through a friendly and intuitive interface. This information system will be designed with all diseases in mind, but its validation will be done with oncological patients.

The QVida+ research project is being developed in a consortium formed by the company Optimizer, the Artificial Intelligence and Computer Science Laboratory (LIACC) from University
Introduction

of Porto, Centro Algoritmi, from University of Minho, and Instituto de Ciências de Vida e da Saúde (ICVS), also from University of Minho.

The part of the above described project that will be tackled in the dissertation is the predictive modeling from simulated data and it will be done in integration with LIACC, which is responsible for the Artificial Intelligence component of QVida+. LIACC \(^1\) is a laboratory that supports the collaboration of researchers from different Faculties that work in the fields of Computer Science and Artificial Intelligence.

1.2 Motivation and Goals

The extension of life expectancy brought about by the evolution in health care, coupled with a rise in chronic illnesses, has accentuated the need to focus on the quality of patients’ lives. This explains the growing importance of QoL as a measure for medical care. Research [GR12] shows that a continuous monitoring of QoL has a great effect in the prevention and early detection of symptoms, particularly in chronically ill patients. A number of research studies that seek to facilitate the integration of this measure in decision systems have been conducted [GR12, FGRR15, RSS+09], but few have been put into practice, and the current assessment is still based in infrequent application of extensive questionnaires.

The gap between the available methods and the optimal way of assessment creates a need for a solution that allows continuous evaluation of QoL. This solution is the goal of the QVida+ project, which aims to achieve this through the use of biometric devices that collect physiological data, using this data in a system that will monitor and predict a patients’ QoL. The creation of predictive models that can be used in the system is the aim of this dissertation. With this aim in mind, the following goals were set:

- Simulation and labeling of physiological data, specifically heart rate and RR interval data;
- Application of machine learning classification methods, namely random forest, deep learning, linear SVM and feed-forward neural networks, as well as clustering methods, specifically k-means, PAM and hierarchical clustering, to the data;
- Evaluation and analysis of the precision of generated predictive models

\(^1\)http://www.liacc.up.pt/
1.3 Dissertation Structure

Besides this introduction, this dissertation contains 4 more chapters. In chapter 2, an overview of the state of the art in the assessment of quality of life, as well as in the use of prediction models for this purpose, is provided, followed by a review of the related work. In chapter 3, the used methodology is detailed, providing the proposed solution and the steps taken to achieve it. In chapter 4, the results of the performed experiments are presented and discussed. In chapter 5, the main conclusions of the dissertation can be found, including an overview of this study’s limitations, as well as suggestions for possible future improvements.
Introduction
Chapter 2

Concepts and Related Work

The present chapter will describe the state of the art in the assessment of quality of life, by reviewing the literature in this area and identifying what progress has already been achieved and in what ways innovation is most needed. First, there will be a short introduction to important concepts. This is followed by a description of the most common methods of quality of life estimation, and then specifically of assessment from physiological data. Finally, there will be a summation of the possible tools that can be used to tackle this problem.

2.1 Introduction

In order to better understand the following sections, it is important to describe the concepts that are involved.

The central concept of the problem at hand is quality of life. Quality of life (QoL) can be defined as the result of a combination of two factors that influence well-being: the capacity to engage in everyday activities along with a person’s satisfaction with this capacity, both of which are a reflection of the mental and physical well being of the subject; and the management of symptoms originating from illnesses and of the illnesses themselves. It encompasses both internal aspects, such as health, independence, life satisfaction, and external ones, like the support surrounding an individual [VFM14].

According to Naughton et al. [NSAC96], QoL is defined as the individual’s subjective perception of their ability to complete activities they consider significant, which is influenced by their current health status. In the context of this problem, a more specific concept derived from the previous arises: Health-Related Quality of Life (HRQoL). This term is, as well, composed of internal and external factors, but it is a narrower term, given that it does not pertain to aspects that are outside the realm of health, even though they might still affect QoL. A factor like low income
Concepts and Related Work

Table 2.1: Domains and Items of HRQoL

<table>
<thead>
<tr>
<th>Physical Health</th>
<th>Psychological</th>
<th>Social Relationships</th>
<th>Relationship with the Environment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Activities</td>
<td>Self-Esteem</td>
<td>Sexual Activity</td>
<td>Economy</td>
</tr>
<tr>
<td>Pain</td>
<td>Spirituality</td>
<td>Social Support</td>
<td>Information</td>
</tr>
<tr>
<td>Dyspnea</td>
<td>Body Image</td>
<td>Family</td>
<td>Means of Transportation</td>
</tr>
<tr>
<td>Mobility</td>
<td>Thoughts</td>
<td>Personal Relationships</td>
<td>Security</td>
</tr>
<tr>
<td>Medication</td>
<td>Negative Feelings</td>
<td></td>
<td>Services</td>
</tr>
<tr>
<td>Insomnia</td>
<td>Positive Feelings</td>
<td></td>
<td>Free Time</td>
</tr>
</tbody>
</table>

has a significant effect on general QoL, but it is outside the concern of health professionals, and is therefore excluded from the scope of HRQoL [GFP93].

Besides the obvious importance for the life of patients, these concepts have a growing significance for the medical practice, especially in the measurement of the impact of diseases, namely the ones with a chronic nature. The standard for medical assessment is to measure physical indicators to evaluate the capacity of patients, but these measurements are often poorly correlated with functional capacity in daily life. Also, the same clinical status has been known to affect different patients in different ways, sometimes with dramatic contrasts in their QoL [GFP93].

Nowadays, HRQoL is considered a medical goal, and it is used in clinical essays, research studies as a way of planning and comparing health management strategies by using this measure as a guide. The assessment of QoL and the practices towards its improvement are also used outside the medical scope, by managers, political analysts, economists, among others. HRQoL is also of particular interest to health administrators, since this measure is often used as an indicator of the quality of care in medical facilities, and also because its variation in patients with the same condition affects the expenditure of hospitals [GR12, GFP93].

2.2 Traditional Methods

Despite the disagreement on a precise definition of HRQoL, this is a measure that can be quantified objectively and in a methodical and standardized way. This quantification takes into account several domains that altogether make up an individual's wellbeing. Table 2.1 presents the domains and items that are part of HRQoL assessment, according to the World Health Organization (WHO) [GR12].

There are two approaches to QoL assessment that are most commonly used: generic instruments that offer an overall summary of HRQoL; and specific instruments that focus on certain diseases or patient groups. These approaches are not mutually exclusive [GFP93].

Within the generic instruments, there are health profiles and utility measures. Specific instruments can be either disease specific, population specific, function specific (i.e. targeting only sleep function) or problem specific, and they tend to try to assess only aspects relevant to the target’s
Table 2.2: Strengths and weaknesses of HRQoL measurement instruments

<table>
<thead>
<tr>
<th>Approach</th>
<th>Strengths</th>
<th>Weaknesses</th>
</tr>
</thead>
<tbody>
<tr>
<td>Generic</td>
<td>Single measurement</td>
<td>May not focus adequately on area of interest</td>
</tr>
<tr>
<td>Health profile</td>
<td>Detects differential effects on different aspects of health status</td>
<td>May not be responsive</td>
</tr>
<tr>
<td></td>
<td>Comparison across conditions possible</td>
<td></td>
</tr>
<tr>
<td>Utility measures</td>
<td>Single number representing net impact on quantity and quality of life</td>
<td>Difficulty determining utility values</td>
</tr>
<tr>
<td></td>
<td>Cost-utility analysis possible</td>
<td>Does not allow examination of effect on different aspects of quality of life</td>
</tr>
<tr>
<td></td>
<td>Incorporates death</td>
<td>May not be responsive</td>
</tr>
<tr>
<td>Specific</td>
<td>Clinically sensible</td>
<td>Does not allow cross-condition comparisons</td>
</tr>
<tr>
<td></td>
<td>May be more responsive</td>
<td>May be limited in terms of populations and interventions</td>
</tr>
</tbody>
</table>

HRQoL, being more narrow than generic methods, and in turn increasing the response rate. These multiple instruments can be combined since they complement each other and provide, when used simultaneously, a more complete and informative representation of a patient’s HRQoL [GFP93].

Health profiles aim to portray all dimensions of HRQoL, including usually physical and psychosocial features. They can include other measures, as is the case of the Sickness Impact Profile, which also evaluates categories such as intellectual functioning, personal hygiene and leisure activities [GGB+75]. Utility measures are derived from decision theory and strive to represent the preferences of patients for their treatment process and the relation between health and death. They result in a HRQoL summary score, usually varying from 0.0, which represents death, and 1.0, which means full health. Table 2.2 provides a comparison of the available assessment instruments, in terms of their strengths and weaknesses [GFP93].

In order to create the measurements previously described, one of the tools most commonly used is questionnaires. Some of the most widely adopted are the WHOQOL-BREF questionnaire, the RAND-36 questionnaire and the Barthel questionnaire. These result in health profiles and utility measures [VFM14].

There are also several standard classifications for QoL that allow a consistent representation of this measure, such as the WHO’s International Classification of Functioning, disability and health (ICF), and the Unified Medical Language System. [VFM14]

Another relevant factor in the traditional measurement of QoL is the mode of administration of the evaluation instruments. The instruments can be administered by an interviewer, by telephone,
Concepts and Related Work

by the patients themselves or by surrogate responders. The most efficient method is using an interviewer, since it maximizes the response rate and minimizes misunderstandings, but it’s the most resource consuming option. In order to consume less resources, self-assessment is the best option, but it might lead to a very low response rate [GFP93].

2.2.1 Computerized assessment

In order to incorporate the assessment of QoL in the decision support systems of health professionals, there has been an attempt to computerize the evaluation of this measure. According to [GR12], the timely knowledge of a patient’s QoL contributes to better decision making, and it is facilitated by the inclusion of systematic patient QoL data in the system.

In a study to create a decision support system for QoL assessment in head and neck oncology patients [GR12], a software was developed to allow patients to perform a self-assessment with standardized measurements, with a friendly and easy to use interface, and the gathered information was graphically displayed to the physician. This study then compared the computerized self-assessment to the traditional one, and found that the computerization did not influence the patients’ answers and allows to identify problems that might not be visible in doctor appointments. It also makes the assessment process considerably faster and less resource-intensive.

Rogausch et al. [RSS+09] also demonstrated, through an implementation study, that computerized QoL assessment is technically feasible and a good source of information and communication for health professionals, but that several impediments must be overcome to allow a broader adoption, such as the lack of time in health care facilities.

2.3 Sensor based assessment

The QoL predictive models we are looking to create are based off physiological data, gathered from wearable sensors. It is, therefore, of importance to review the state of the art in sensor-based assessment. We will be looking at the several types of physiological data that can be gathered and what significance they have, and then also specifically at wearable sensors.

Sensor-based assessment has been used for various purposes, among which the most significant cases will be discussed. Iakovakis et al. [IH16] showed that it is possible to use a smartwatch to predict standing hypotension with high precision, by using the device’s capability to measure heart rate variability. Sun et al. [SSHE10] used physiological time series data of patients and managed to predict their health trajectories with high precision, by using the information of similar patients.

Besides the applications in health informatics, sensor-based assessment can also be used for emotion and behavior recognition. Bogomolov et al. [BLF+14] managed to recognize daily stress from mobile phone sensor-detected activity information, weather conditions and personality traits.
### 2.3.1 Physiological data

There are several types of physiological data that can be collected from patients through wearable sensors. Table 2.3 shows the most popular and significant metrics, their meaning and their possible applications. Other significant measures are body temperature and respiration, which measures the volume of air in the lungs and gathers breathing patterns.

### 2.3.2 Wearable Sensors

We will be extracting data from wearable sensors. These were chosen due to their practicality, in order to allow for continuous wear, and in turn, continuous monitoring. There are, however, two types of wearable sensors: consumer-grade and medical-grade.

Most of the research based on physiological data has collected it from medical-grade devices. These are the most accurate devices in terms of measurements but they are not the most practical, which means their use might not be feasible for continuous wear. Also, their high cost is prohibitive for research with large validation.

This means the best option in this case is to use consumer-grade devices, which offer practicality, but have less precision in their measurements. Nogueira et al. [NR16] performed a comparison of medical-grade and consumer-grade wearable devices, and evaluated the trade-offs of the use of

---

Table 2.3: Physiological metrics and their applications

<table>
<thead>
<tr>
<th>Measure</th>
<th>Metrics</th>
<th>Meaning</th>
<th>Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Electrodermal activity</td>
<td>Skin conductance</td>
<td>Measure of skin conductivity</td>
<td>Emotional changes [CKGP06]; cognitive activity [SRQ00]</td>
</tr>
<tr>
<td>Cardiovascular measures</td>
<td>Blood pressure, blood volume pulse, heart rate, heart rate variability, inter-beat interval</td>
<td>Activity of the cardiovascular system</td>
<td>Distinguishing positive and negative emotional states [MA07]</td>
</tr>
<tr>
<td>Electromyography (EMG)</td>
<td>EMG (ie. Facial EMG)</td>
<td>Contraction of muscles</td>
<td>Distinguishing positive and negative emotional states [HL06]</td>
</tr>
</tbody>
</table>

However, it was shown that removing one of the sources of information made the predictions a lot less accurate, meaning stress could not be inferred simply from sensor data. Another study [JPP+15] analyzed the difference in physiological responses in the presence of boredom, pain and surprise, and found significant differences in heart rate, skin conductance, skin temperature, blood volume pulse and pulse transit time. It also showed the possibility of recognition of these emotions by machine learning approaches.
either. It was also found that the best consumer-grade devices were Microsoft Band 2 \(^1\) and BITalino \(^2\), due to their versatility, high accuracy and completeness. Microsoft Band 2 was considered less intrusive, but BITalino offers the most complete set of measurements.

### 2.4 Prediction from physiological data

The present section will focus on the methods used for prediction, drawing from both studies mentioned in the previous section and other relevant studies that performed prediction based on physiological data. First, the relationship between these measures and QoL and health will be analysed, followed by a review of possible algorithms that can be used to aid our solution, and after that, there will be an overview of other research works where the application of machine learning methods allowed to predict from this type of data.

#### 2.4.1 Relating physiological data with health and lifestyle

The relationship of the previously mentioned physiological data with health and the presence of diseases has been studied numerous times. In this section, we will focus on cardiovascular measures, since these are the most common measurements, and the most studied as well.

In a state of the art paper to review the relation of resting heart rate with cardiovascular disease, Fox et al found that risk for disease increases continuously with resting heart rate above 60 bpm (beats per minute) \([FBC^+07]\). In a study to evaluate heart rate as a risk factor in patients with coronary artery disease and left-ventricular dysfunction, it was found that patients with resting heart rate over 70 beats/min had increased risk for cardiovascular death, and of recurrence of cardiac episodes. This risk got more significant with every 5 bpm increase in resting heart rate \([FFS^+08]\). The significance of heart rate as a predictor of morbidity was also observed in a study that measured heart rate recovery after exercise, as well as patients’ treadmill exercise scores. It was found that abnormal heart rate recovery after exercise, which is defined as the failure of the heart rate to decrease by more than 12 bpm during the first minute after peak exercise, was a predictor of death \([NCB^+00]\).

The relation of heart rate with QoL has also been studied by Jaber et al. Measuring patients’ heart rate with a Holter monitor and the 6MWT, it was observed that patients with maximal heart rate over 110 bpm on the 6MWT and average heart rate over 80 bpm on the Holter monitor had worse QoL than patients with measurements below these values \([JCI^+10]\).

Another important indicator of health and overall wellbeing derived from cardiovascular measures is heart rate variability (HRV), which refers to the variations in the inter-beat intervals. This

---

\(^1\)https://www.microsoft.com/microsoft-band
\(^2\)http://www.bitalino.com/
The measure has been studied countless times in connection with the presence of physical and mental disease, as well as stress.

In an article by Rajendra Acharya et al, a review of several studies researching the effectiveness of the use of HRV as a health predictor was conducted. This measure can be used to assess the state of the autonomic nervous system (ANS), which regulates cardiac activity, and this assessment allows a better prediction of a number of health conditions. An example of this is the decrease of HRV after a recent myocardial infarction, or the decrease of HRV in patients with left ventricular hypertrophy, which is associated with increased mortality. HRV was also shown to be lower in patients with diabetes, even before they developed clinical symptoms of neuropathy. Smoking, drugs and acute ingestion of alcohol have also been shown to decrease HRV. On the other hand, the HRV of aerobic athletes presents an increased power in all frequency bands. This measure is also influenced by gender and age, being lower the older a person is, and usually more so in females \[\text{[RPK}^{+06}]\]. Therefore, predictions made using HRV should take this variation into account.

Another factor of influence in HRQoL is stress, which can also be assessed through HRV. In a study to evaluate the effect of mental stress on HRV and blood pressure during computer work, Hjortskov et al found that during stressful situations, there was a reduction in the high-frequency component of HRV and an increase in the low- to-high-frequency ratio, as well as an increase in blood pressure, in contrast to the control situation \[\text{[HRB}^{+04}]\]. Dishman et al observed that, for physically fit men and women, after perceived emotional stress during the previous week, the high-frequency component of HRV is lower, independently of age, gender and fitness \[\text{[DNG}^{+00}]\]. The cumulative high-frequency component of HRV has also been shown to be lower during a stress test in subjects who had previously attempted suicide \[\text{[WCF}^{+16}]\].

Given the extensive research mentioned, we can conclude that HRV is a very informative measure that should be collected when trying to predict QoL from physiological data. Spectral analysis of HRV is also valuable in this context. Blood pressure, resting heart rate and continuous heart rate monitoring are also significant for these predictions.

2.4.2 Algorithms

This section will present a review of promising methods to be used in the solution, and will be based on this book \[\text{[WFHP16]}\].

2.4.2.1 Linear Regression

Linear regression is a staple method in statistics that works naturally with numeric classes and attributes. It predicts a numeric value with a linear combination of the attributes, by assigning weights to them, in the following way:

\[ x = w_0 + w_1 a_1 + w_2 a_2 + \ldots + w_k a_k \]  (2.1)
Concepts and Related Work

where \( x \) is the class; the \( a \) values are attribute values, and the \( w \) values are the determined weights.

These weights are calculated from the training data, following the method of least-squares, which chooses the coefficients \( w \) in order to minimize the sum of the squares of the difference between the predicted values and the corresponding real instance values. This sum of the squares of the differences is shown below:

\[
\sum_{i=1}^{n} \left( x^{(i)} - \sum_{j=0}^{k} w_{j}a_{j}^{(i)} \right)^{2}
\]  

(2.2)

This method is a simple solution for numeric prediction, but it is not appropriate if the data has a nonlinear dependency. Despite this, it is a good starting point for the exploration of other methods.

2.4.2.2 Polynomial Regression

Polynomial regression is a form of linear regression (2.4.2.1) that predicts a numeric value \((x)\) with an \(n\)th degree polynomial:

\[
x = w_{0} + w_{1}a_{1} + w_{2}a_{2}^{2} + w_{3}a_{3}^{3} + \ldots + w_{k}a_{k}^{k}
\]  

(2.3)

where \( x \) is the value to be predicted, \( w \) are the weights determined, and \( a \) are the values of the attributes. The determination of the weights is done through the minimization of the sum of the squares of the differences between the predicted values and the true values of training data instances, similarly to linear regression (2.2).

This method is used when the relationship between the class to predict and the attributes is nonlinear, but it is still a linear statistical estimation.

2.4.2.3 Bayesian Networks

Bayesian networks are structures that represent probability distributions. They are directed acyclic graphs with nodes that represent attributes, connected by directed edges. Inside each node, there is a probability distribution table that predicts the class probability for each instance, as is shown in figure 2.1 [WFHP16]. In this image, the attributes are "Outlook", "Windy", "Temperature" and "Humidity". Each row of a table defines a probability distribution of the values of the node’s attribute. In order to find the probability of the class values for an instance, all that needs to be done is to multiply the probabilities of that instance’s attribute values.

In the creation of a Bayesian network, the nodes are already predetermined, since there is one node for each attribute. The searching for possible sets of edges is the first step, followed by the estimation of the conditional probability tables for those sets, and then calculating the log-likelihood of the network, in order to measure its quality. There are two common metrics to
estimate a network’s quality: the Akaike Information Criterion (AIC); and the MDL score. Both of these use the log-likelihood as part of their calculations.

The search of the sets of edges is different depending on the learning algorithm used. K2 is a fast algorithm that starts with ordered nodes, processes each node and adds to it the edge that will maximize the network’s quality. Due to the greedy nature of this algorithm, it is advisable to run it several times with different node orderings. Another algorithm is the Tree Augmented Naïve Bayes (TAN), which takes a Naïve Bayes classifier and adds edges to it, with the class attribute as the parent node of each attribute node.

The use of Bayesian networks allows the graphical representation of probability distributions in a concise and understandable way.

2.4.2.4 Random Forests

Random forests are an ensemble learning method which uses a collection of decision trees, each of them learning from different sets of randomized attributes. The output is decided as either the mode of the classes predicted by the decision trees, in case of a classification problem, or by the mean values of prediction of each tree, in the case of a regression problem.

Decision trees are a predictive model that infers the value (regression) or the class (classification) of an instance by its attributes. They take the shape of a tree that represents the rules for their predictions, as seen in figure 2.2 [WFHP16].

The creation of this tree is done through the partitioning of the training data set based on a test of the value of an attribute. This partitioning is done on each subset recursively until all the instances present in a subset are of the same class, or until further partitioning no longer adds predictive value to the tree. Notable algorithms for learning decision trees are ID3, C4.5
Concepts and Related Work

Figure 2.2: Example of a decision tree

(An evolution on ID3) and CART (Classification and Regression Tree), and each algorithm uses certain metrics to determine which attribute will best partition a set. Some of the metrics that are commonly used are information gain (change in information entropy) and variance reduction.

Random forests present an advantage over standalone decision trees since they tend to rectify the overfitting that usually occurs in decision trees.

2.4.2.5 Support Vector Machines

Support Vector Machines (SVM) are predictive models that infer the value of instances by constructing a hyperplane or set of hyperplanes in a high-dimensional space that divide instances according to their attributes. This method can be used for both classification and regression problems. In the creation of these models, there is an attempt to create the largest margin between the hyperplanes and the nearest instances, in order to prevent overfitting and improve generalization. If a hyperplane that divides the classes cannot be found in an $n$-dimensional space, ($n$ being the number of attributes), then a new attribute is added to allow the division.

When applied to regression problems, SVMs attempt to find a function that approximates the training data points, like linear regression, while minimizing the prediction error. Yet, in contrast, the minimization of error is accompanied by a maximization of flatness of the function, in order to avoid overfitting. The minimization of error is also done differently, since it focuses on the absolute error, instead of the squared error.

This method creates models with a high accuracy, good generalization power and no local minima, but it is considered slow and highly complex.

2.4.2.6 K-Means Clustering

Clustering is useful when the instances of data are to be divided in groups, since it does not predict classes, but it creates clusters, grouped by their characteristics.
Concepts and Related Work

The most basic clustering technique is K-Means. In this technique, k initial points are selected to be initial cluster centers, and instances are grouped with their nearest center. K is previously specified and not chosen by K-Means. Then, the mean value of the instances in each cluster is computed to create a new cluster center, reassigning instances after this change. This continues until there are no changes in the clusters.

This technique requires knowing previously which is the ideal number of clusters - k -, or testing with several possible numbers of clusters.

2.4.2.7 Partitioning Around Medoids Clustering

Partitioning Around Medoids (PAM) is a technique that performs clustering of the data into k clusters around medoids, and it is considered a more robust version of K-Means, being less sensitive to noises and outliers. It uses a medoid instead of the mean, which is an actual instance of the cluster that is meant to represent it. It is the most centrally located object of the cluster, with the minimum sum of distances to the other instances \([MJH+11]\).

2.4.2.8 Hierarchical Clustering

Hierarchical clustering consists in the agglomerative or top-down placement of clusters in a hierarchy of clusters, which can be represented in a dendrogram. The agglomerative approach starts with each instance in its own cluster, and proceeds to join instances that are the closest to each other, following a certain specified measure of distance. This is done until only one cluster is left. Reversely, the top-down approach starts with only one cluster, and divides it recursively, until every instance is on its own.

This method has the advantage over other clustering approaches of not requiring the previous definition of a number of clusters. It is also simple to implement and generates small clusters which may be helpful for knowledge discovery. However, it is sensitive to noise and outliers.

2.4.2.9 Deep Learning

Deep learning is a set of learning techniques that uses multiple steps to create very complex features. This is done using a sequence of numerous layers (more than 3) that perform non-linear processing, each of which will learn the attribute representations from the input they receive from the previous layer. This process is helpful to distinguish which features are most significant for learning.

One of the main reasons to use deep learning is its empirical effectiveness when compared to other methods. It is a method that has high accuracy and gets more precise when it is trained with more data. It also allows to find structure in unlabeled and unstructured data. However, it is very
computationally costly to train, and, since it works best with large amounts of data, it requires a large training set, otherwise it might not offer a significant precision advantage over other methods.

### 2.4.3 Methods used in studies

Wang et al. [WZX+16] used a deep learning model based on a convolutional neural network to learn features from physiological data, and used the results on an anomaly detection method, in order to identify patients with latent risks of future illness. The results were promising and the anomaly detection had high performance.

In another study [YLB10], aimed at recognizing driver fatigue, physiological signals from ECG (electrocardiography) and EEG (electroencephalogram) were used in a dynamic Bayesian network, and good results were obtained. It was also found that the more features were used, the more accurate the results were, and that the gathered physiological data was significant for fatigue recognition.

Clifton et al. [CCP+14] managed to identify patient deterioration and abnormal states with high accuracy, using a one-class SVM trained on physiological data from wearable sensors, specifically heart rate, blood pressure, oxygen saturation and respiratory rate. This study also offered a proposal for parameter-estimation for SVM that is suited for physiological data.

### 2.5 Related work and similar approaches

Since this project is within the scope of the QVida+ project, it is crucial to mention the paper "QVida+: Quality of life continuous estimation for clinical decision support" [RFG+16], which outlines the proposed solution for the system and provides an extended explanation of the motivation behind this project.

There is a lot of research that relates to the problem of QoL estimation and prediction, in order to improve patients’ lives. As was mentioned in the previous section, Wang et. al [WZX+16] used physiological data to predict risk of future illness. Yet, there is not a focus on QoL or continuous monitoring through unintrusive wearable sensors. The study conducted by Penny et al. [PS12] was able to identify which symptoms affect HRQoL more in patients with IBS (Irritable Bowel Syndrome), through the use of data mining and through the attempt to classify HRQoL using models trained with the data from these patients. However, the classification was not highly accurate, and the data used was not physiological and did not allow a continuous monitoring.

There are also some studies proposing similar solutions, but without validation, such as the study by Liaqat [LD16], that proposes a solution for prediction of episodes of medical conditions through data from wearable sensors.
2.6 Conclusions

It is possible to conclude that QoL is a significant measurement in health care nowadays, being a focus of professionals and patients alike. Yet, the currently adopted methods of evaluation and monitoring are outdated and cannot be used in a continuous way. The creation of a continuous monitoring solution that is non-intrusive, and, therefore, more easily adoptable, is a logical next step. The prediction of QoL would also help health professionals, by providing a new decision support metric.

Through the analysis of several studies, we can gather that the prediction from physiological signals is possible, and that several methods can be used to that end. We can also conclude that there are currently no solutions for the prediction of QoL from data obtained from wearable sensors, even though there is an interest in creating systems that focus on this metric.
Concepts and Related Work
Chapter 3

Implementation of proposed solution

The proposed solution is a part of the system presented in the QVida+ project [RFG+16]. The methodology that was followed to create this solution consists of the following steps:

- **Creation of a signal transformation tool**: A signal transformation tool was created, in order to insert anomalies and trends in data, and visualize and export the results. The goal is to have training data that will create models resistant to irregular and faulty input. This involved several steps: the creation of the interface, the development of the importing and exporting of data, the creation of the controls for data transformation, and the testing of the tool.

- **Data simulation**: Through small samples of data, the signal transformation tool simulates weeks of data for different patients, with a lot of variation among them. This was done to overcome the obstacle of having no available collected data, as was previously planned.

- **Data labeling**: The instances of the collected data will be labeled according to their heart rate and RR interval characteristics, having in account the research mentioned in chapter 2. This type of labeling is necessary due to the data being simulated and not real. In order to further test our models, the data will also be randomly labeled and then put through the models, to see the influence of the labeling and the resistance of the models.

- **Application of algorithms - Modeling**: The predictive models for QoL were learned based on the simulated data and using several methods.

- **Comparison and evaluation of the resulting models**: The created predictive models were evaluated and compared in terms of accuracy. These two steps were iterative, for a continuous improvement of the predictive models according to the comparisons made.
Implementation of proposed solution

3.1 Tools

In order to apply the methods that were previously mentioned and create an appropriate solution, several tools are available. In the context of this project, due to the architecture of the solution, Java is used, in conjunction with R\(^1\). R is a language and environment for statistical computing and graphics, which provides a great variety of statistical and graphical techniques. It is highly extensible, since it can have packages added onto it. The open source nature of the development of these packages makes it so there are a lot of packages of great quality available. In order to use R and Java together, JRI\(^2\) will be used. JRI is a Java/R interface that allows running R code within a Java application.

3.2 Architecture of proposed solution

The solution aims to cover all the steps previously mentioned and is divided in two programs: the signal transformation and data creation tool; and the labeling and modeling program.

The signal transformation and data creation tool is a Java application with a JavaFX UI, that allows the import of data contained in csv files, the visualization of such data, and the manipulation of it, as well as exporting it after the manipulation, and simulating mass amounts of new data from the input.

The labeling and modeling program is a Java and R application with a JavaFX UI, that is responsible for the labeling of the simulated data and for the clustering and modeling using the labeled data.

Figure 3.1 illustrates the structure of these programs. They will be discussed in further detail in the next sections.

---

\(^1\)https://www.r-project.org/

\(^2\)http://www.rforge.net/JRI/
3.3 Signal transformation tool

The signal transformation tool was built in order to manipulate data and visualize it, and most importantly, to simulate mass amounts of data. It allows the import of csv files in a specific format, and the export of data that has been transformed to csv files, as well as the mass creation and export of simulated data derived from the input. The transformation of data is done through filters, and the available filters are the following:
Implementation of proposed solution

- **Arithmetic filter**: it allows the sum, subtraction, multiplication and division of the signal, with a specific value, and the setting of the slope of the decay before and after the transformation;

- **Sine/cosine filter**: it allows the transformation of the signal into a sine or cosine wave, with the specified frequency and amplitude, as well as the setting of the slope of the decay before and after the transformation;

- **Periodic sine/cosine filter**: it allows the transformation of the signal into a sine or cosine wave that repeats itself after a certain period. It also allows the setting of the slope of the decay before and after the transformation;

- **Sigmoid filter**: it allows the transformation of the signal into a sigmoid curve, and the direction and scaling of it. It also allows the setting of the slope of the decay before and after the transformation;

- **Triangle filter**: it allows the transformation of the signal into a triangle shape, as well as the setting of the highest and lowest values, and the upward and downward slope of the desired signal. It also allows the setting of the slope of the decay after the transformation.

All of the above filters enable the user to select the beginning and end of the transformation, as well as which channel to apply it to. The applied filters are listed in a filter log present in the UI.

This tool has a few other features, listed below:

- Undo applied filters;

- Setting of minimum and maximum values for each signal channel;

- Visualization of original data and transformed data, with possibility of selecting signal channel;

- Save data post-transformation in csv files;

- Mass simulation of data from input date;

- Feature extraction from several csv files.
3.4 Data simulation

The simulation of mass amounts of data was done following these steps:

- Import of sample csv data with two columns, one with RR data and another with HR data. The used data had 241 rows, and it was extracted from a MSBand during rest, and therefore, had little variability;

- For each column of this sample data (RR and HR):
  - Definition of limit levels for the values that can be reached in the column, for example: the maximum limit for HR would be around 190;
  - For 1000 turns, repeat the following process:
    * Run a combination of the mentioned filters, depending on the number of the turn. The higher the turn number, the more transformed the data will be, applying more filters with higher values;
    * Generate ten weeks of data five times (being that one set of transformed data represents a week), following the process below:
      · For the first generation, make ten copies of the data resulting for the transformation of the present turn;
      · For the next four generations, make four different variants of sets of 10 weeks. In each of these 10 week sets, there should be increasing variation as weeks progress;
  - Remove sets of 10 weeks with unreasonable values, for example, with too much variation too quickly;

- After the generation of the sets of 10 weeks, order them by ascending variability;

- Save all sets to separate csv files, and each file will represent a patient’s set of 10 weeks.

After this data generation, feature extraction takes place, in order to facilitate the modeling and clustering process. The feature extraction is done to each week of a patient’s set, and also to the full set of 10 weeks. The features that were extracted were mostly statistical measures, as well as other measures that will be explained in this section. The common statistical metrics taken from the data were the mean, the median, the mode, the maximum value, the minimum value, the standard deviation and the interquartile range. Some other measurements were also taken:

- Average difference between each measurement: The mean of difference in values between each measurement is calculated for each week of a set and for the full set;

- Standard deviation of the different between each measurement: The deviation from the mean of the difference in values between each measurement;
Implementation of proposed solution

- **Mean absolute deviation from rest values:** This measure is calculated as the mean of the absolute difference of each of the measurements from rest values. It is valid only for the HR column, since there is no rest value for RR: The value that was used to represent rest was 70 bpm.

- **Skewness:** This measure represents the asymmetry of the probability distribution of a variable about its mean. It can be positive, negative or undefined. In a negative skew, the left tail of the distribution is longer, and the distribution is concentrated in the right side. In a positive skew, the opposite happens. If the distribution is symmetric, the skewness is zero.

- **Longest duration of continuous high values:** This measure only has meaning for the HR column. The duration of continuous values above 125 in a week is measured, and the longest one is saved in this feature. This measurement was taken in case it had some information value when it came to patient exercise habits.

- **Number of spikes:** If the increase between each measurement is higher than 30 for HR and 0.5 for RR, it is considered a spike. This feature could be useful to identify anomalies in patients’ HR.

- **Number of dips:** If the decrease between each measurement is higher than 30 for HR and 0.5 for RR, it is considered a dip. This feature could also be useful to identify anomalous HR in patients.

### 3.5 Data labeling

Since the data available for this project was simulated, it required artificial labeling, to associate QoL scores to each simulated set of weeks. The scores used varied from 1 to 10, with 1 representing very poor QoL, and 10 representing excellent QoL. Two types of labeling were performed on the simulated data: random labeling and labeling based on the features of the data. The random labeling was performed in order to provide us with a control measurement of the accuracy of the modeling, when compared with modeling done on data that was labeled using information contained in the features. It also allowed to test the flexibility of the models in extreme conditions.

The labeling was done in a separate program from the signal transformation tool, since it is directly followed by the modeling and clustering, and in the case of random labeling, this is not a completely separate step.
3.5.1 Randomly labeled data

The labeling in this case was done providing the option to change the entropy, which could vary from 0 to 1. The closer to 1, the more random the labeling is, and in the case of entropy set to 1, the label is assigned using the ThreadLocalRandom function in Java. In the case of entropy set to 0, the labels are assigned to patients sequentially from 1 to 10. The in-between values are set with a mix of sequential and random values, with the randomness increasing with the increase in the entropy value.

3.5.2 Data labeled from features

Research presented in chapter 2 showed that there are several indicators of poor health that can be extracted from HR and RR. Most importantly, RR intervals can be used to infer HRV, since these represent the interval between R’s, which correspond to a peak in an ECG wave. The variation in these intervals, which can be measured by the standard deviation of the RR intervals, is the HRV of the patient.

Following the research previously mentioned, the indicators below were used to infer a label for each patient:

- Heart rate variability (HRV): using information extracted from RR intervals, the HRV was extracted. A low HRV was penalized, since it often indicates poor health [RPK+06];

- Mean heart rate (HR): a consistently high mean HR over the weeks in the patient set was penalized, even more so if it was above 110 bpm, which can indicate cardiac issues [FBC+07, FFS+08, JCJ+10];

- Spikes and dips in HR: too many spikes and dips were penalized since they can indicate an abnormal heart rate control, which negatively affects health [APK+12];

- HR activity that could indicate exercise: longer spurts of continuous high HR activity were seen as indicators of exercise, and therefore, as a factor for a higher QoL, since regular exercise might indicate a higher QoL, due to the possibility of mobility and also have a positive effect on it [GFP93, CCAF+16];

- Abnormal HR values: HR values over 185 and under 30 were penalized, since they can indicate abnormal heart activity, which negatively affects QoL [APK+12, JCJ+10].

It is important to note that this labeling technique is experimental, and it was executed out of necessity due to the lack of real data. The simulated data might not represent usual patient data.
Implementation of proposed solution

and the labeling done on it might not translate to an accurate connection between features and real-world QoL. Yet, it is the closest we could come to a valid labeling method and a varied data set that has elements in common with a real data set, in order to prove experimentally that modeling can provide results with this problem, and that it is flexible to extreme changes in the data set.

3.6 Modeling and clustering

After the simulation of data, the extraction of features and their labeling, the next step is modeling and clustering. This step results in models that should predict QoL given HR and RR data. Both the labeling and the modeling and clustering are done in a separate program, due to the difference in functionality in these steps, to allow the use of these tools separately, and also because of the use of R in the latter program.

This step will be repeated, since it will be performed after five runs of random labeling for several different values of entropy, and it will also be performed after the simulated data is labeled using information from its features.

3.6.1 Tools and packages used

The inclusion of R within a Java application required the use of JRI, a Java/R interface that has been described in chapter 2. In the R sections of the program, the following packages were used:

- caret [KWW+17]: a package with a set of functions to facilitate the process of creating predictive models, including preprocessing, feature selection and model tuning;
- clValid [BPDD08]: a package with functions to validate results of different clustering methods applied to data;
- deepnet [Ron14]: a package that is a toolkit for deep learning, with functions that implement several algorithms for this purpose;
- e1071 [MDH+17]: a package with a mixture of functions related to statistics, that is required by the caret package;
- kernlab [KSH16]: a package with functions for kernel-based machine learning methods, which in this project is used specifically because of SVMs;
- nnet [RV16, VR02]: a package for implementing feed-forward neural networks;
- ipred [PHR+17]: a package required by the caret package, that is used to create improved predictive models;
3.6.2 Methodology

In this section, the modeling and clustering methodology will be explained in more detail, with further explanation on which algorithms and techniques are used in this step.

3.6.2.1 Clustering

Clustering is the process of grouping objects in clusters, using the similarities in their features. It was used in this case to detect if there are natural groups that form within our dataset and to provide another way to perform statistical analysis on the simulated data. There are several algorithms that can be used, and in some of them, the number of clusters is not defined by the algorithm, and it is inputted by the user. Therefore, it was important to not only use several algorithms for comparison, but also, to run each algorithm with a different number of clusters, in order to compare their performance in each case. The clustering in this program was achieved through the clValid package, which not only performs the clustering, but also compares several algorithms with several numbers of clusters in one call.

The algorithms that were used in this step, which were explained in more detail in chapter 2, were the following:

- K-means: This is one of the methods that does not decide the number of clusters and needs it as input. There will be a comparison of number of clusters from 3 to 5. This algorithm has a few drawbacks, one of them being the requirement of input of number of clusters, as well as a preference for similar sized clusters, which can lead to incorrect cluster borders;

- PAM (Partitioning Around Medoids): This is the most common implementation of one of the optimizations of k-means, k-medoids. In contrast to k-means, it chooses a data point as the center of a cluster, and uses a different calculation for distance between points. This makes it more resistant to noise and outliers;

- Hierarchical clustering: This is a method that aims to create a hierarchy of clusters, instead of a single partition. The data set objects are connected to form a cluster based on the distance between them. The specific algorithm that will be used is UPGMA (Unweighted Pair Group Method with Arithmetic Mean), which is agglomerative. A major drawback of hierarchical clustering is its slow performance with large data sets. Yet it is more flexible than the previous methods, since it can use any distance measurement. It can also provide more information, since it creates clusters of a variety of sizes.

To evaluate the performance of these methods, the following metrics provided by the clValid package are used:
Implementation of proposed solution

- Internal measures: these measures reflect the compactness, connectedness and separation between clusters. The chosen comparison metrics were connectivity, silhouette width - the average of each object’s cluster assignment’s degree of confidence - and the Dunn index - the ratio of the smallest distance between observations not in the same cluster to the largest intra-cluster distance;

- Stability measures: these measures are a comparison between results from clustering with the full data set and results from clustering removing each column, one at a time. These were the specific metrics used:
  - Average proportion of non-overlap (APN): this measures the average proportion of observations not placed in the same cluster by clustering based on the full data and clustering based on the data with a single column removed. If this metric is close to zero, it signifies highly consistent clustering results;
  - Average distance (AD): this measures the average distance between observations placed in the same cluster by clustering based on the full data and clustering based on the data with a single column removed. Smaller values are preferred in this metric;
  - Average distance between means (ADM): this measures the average distance between cluster centers for observations placed in the same cluster by clustering based on the full data and clustering based on the data with a single column removed. Smaller values are also preferred in this metric;
  - Figure of merit (FOM): this measures the average intra-cluster variance of the observations in the deleted column, where the clustering is based on the remaining samples. It estimates the mean error based on the cluster averages, and smaller values are also preferred.
3.6.2.2 Modeling

In the context of this work, modeling consists of the creation of predictive models, that should be capable of estimating patients’ QoL when provided with their HR and RR data features. In this step, there was also a variety of methods used, which were all provided by the caret package. This package provides preprocessing functions, as well as feature selection, and it allows automatic tuning of models with optimal parameters.

Before the creation of some of the models, the data was normalized with min-max normalization, which means that it was scaled so that all feature values range from 0 to 1. This is crucial when distance measures are part of the algorithm that creates the predictive model.

Another important step is feature selection, especially in this case, since there are over 300 features, which makes for a high dimensional data set, that can be hard to deal with in some algorithms, and can also make the creation of models very slow. The caret package offers a few options to select the best features, and the one chosen was recursive feature elimination, which is a wrapper method, meaning that it evaluates features’ importance by adding or removing them and assessing the resulting performance of models that use the chosen feature set. The result is an optimal set of features that will create the best performing model.

The caret package provides several ways to facilitate model tuning and validation. We used 10-fold cross-validation, as well as the built-in functionality of caret’s train function, which evaluates model parameters with resampling, and returns the better tuned model.

The algorithms that were used to create predictive models, which were all run through caret’s train function, were the following:

- Random forest: this is an ensemble learning method which uses a collection of decision trees, each of them learning from different sets of randomized attributes, giving as a final output with a function of the predictions of the decision trees, that can be the mean or other calculation. This algorithm presents an advantage over simple decision trees, because it avoid over-fitting. However, it can create very large models that are slow at evaluating new data sets;

- Deep learning: this method consists in using neural networks with more than one hidden layer to predict from data. The advantages it presents over "shallow" neural networks is the reduced need for feature engineering and the easy adaptability to all kinds of problems;

- Support vector machines (SVM): this method is based on finding hyperplanes that divide data instances according to their classes, with these hyperplanes possibly belonging to higher dimensions than 2 or 3, depending on the divisibility of the instances. It can perform very well, especially in smaller, cleaner datasets. However, it is less efficient in noisy datasets, and has a slow performance when used with larger data;
Implementation of proposed solution

- Feed-forward neural network: this method consists in an artificial neural network that is connected without cycles, where the information moves only in one direction, and that has only a single hidden layer. This is a simple method that will be used for comparison.

All of the previously mentioned algorithms were discussed in greater detail in chapter 2.

To assess the performance of these methods, the following metrics provided by the caret package are used:

- Accuracy: measures the ratio of correct predictions to the total number of cases evaluated;

- Area under the curve (AUC): refers to the area under the ROC - or receiver operating characteristic - curve, which is created by plotting the true positive rate against the false positive rate. Unlike accuracy, it is insensitive to imbalanced classes;

- Logarithmic loss: quantifies accuracy by penalizing false classifications. It has no upper bound and it should be minimized;

- Mean sensitivity: for each class in the data, defining that class as the positive class, sensitivity means the proportion of positives that are correctly identified as positives;

- Mean specificity: for each class in the data, defining that class as the positive class, specificity means the proportion of negatives that are correctly identified as negatives;

- Mean balanced accuracy: it is the average accuracy obtained in each class. It is different from the conventional accuracy measure, since if that accuracy is high only because the classifier takes advantage of an imbalanced test set, the balanced accuracy will be much lower.
Chapter 4

Results and Discussion

This chapter will be dedicated to presenting the results obtained from the several steps followed in this Dissertation, ranging from the simulation of data, to its labeling, to the clustering and modeling. It will also contain a discussion of these results.

4.1 Data simulation

The data simulation, executed according to the process described in chapter 3, originated 2237 files, each representing a patient’s set of 10 weeks of HR and RR data. After this, the features listed in section 3.4 were extracted for each week, and for the total set of 10 weeks. This resulted in 309 features for each instance. The ones that are relative to a specific week in the patient’s set are named following this structure: <featurename><channel>W<numberofweek> (e.g. MeanHRW1 for the mean of the HR channel in the first week). The features relative to the total 10-week set are named with this structure: Total<featurename><channel> (e.g. TotalMeanHR for the mean of the HR channel for the full 10-week set).

Given that all of the 10-week sets originated from a small HR and RR sample that was taken during rest, with little variation, and that a significant part of the sets suffered little transformation with a few radical transformations in the latter stages of the simulation, in order to mimic real-life rest conditions, the resulting dataset contains a lot of instances that have what can be considered normal values with little variation. This can affect the labeling, as it was performed mainly by looking for anomalies and specific indicators in the patient’s features. There can be a larger number of instances with average-to-good QoL labels.

The resulting dataset is too large to present here, but some statistics of the main features corresponding to full 10-week sets (and therefore denominated "overall" features) are listed in table 4.1.

The statistics of every overall feature is presented in appendix A.
Results and Discussion

Table 4.1: Statistics of main overall features in simulated data

<table>
<thead>
<tr>
<th>Feature</th>
<th>Minimum</th>
<th>First Quartile</th>
<th>Median</th>
<th>Mean</th>
<th>Third Quartile</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overall Mean HR</td>
<td>55.0</td>
<td>63.11</td>
<td>65.43</td>
<td>69.82</td>
<td>73.40</td>
<td>146.53</td>
</tr>
<tr>
<td>Overall Mean RR</td>
<td>0.2221</td>
<td>0.7368</td>
<td>0.8657</td>
<td>0.8738</td>
<td>1.0058</td>
<td>1.5710</td>
</tr>
<tr>
<td>Overall Median HR</td>
<td>55.0</td>
<td>63.0</td>
<td>63.0</td>
<td>64.15</td>
<td>63.0</td>
<td>159.0</td>
</tr>
<tr>
<td>Overall Median RR</td>
<td>0.20</td>
<td>0.8296</td>
<td>0.8794</td>
<td>0.8745</td>
<td>0.9209</td>
<td>1.60</td>
</tr>
<tr>
<td>Overall Mode HR</td>
<td>55.0</td>
<td>62.0</td>
<td>63.0</td>
<td>65.46</td>
<td>63.0</td>
<td>180.0</td>
</tr>
<tr>
<td>Overall Mode RR</td>
<td>0.20</td>
<td>0.20</td>
<td>0.8960</td>
<td>0.8909</td>
<td>1.60</td>
<td>1.60</td>
</tr>
<tr>
<td>Overall Min HR</td>
<td>55.0</td>
<td>55.0</td>
<td>60.0</td>
<td>58.11</td>
<td>60.0</td>
<td>68.0</td>
</tr>
<tr>
<td>Overall Min RR</td>
<td>0.0996</td>
<td>0.0996</td>
<td>0.0996</td>
<td>0.1227</td>
<td>0.0996</td>
<td>1.1773</td>
</tr>
<tr>
<td>Overall Max HR</td>
<td>55.0</td>
<td>74.0</td>
<td>103.0</td>
<td>110.7</td>
<td>145.0</td>
<td>180.0</td>
</tr>
<tr>
<td>Overall Max RR</td>
<td>0.916</td>
<td>1.775</td>
<td>1.775</td>
<td>1.741</td>
<td>1.775</td>
<td>1.775</td>
</tr>
<tr>
<td>Overall Std Dev HR</td>
<td>0.0</td>
<td>0.0427</td>
<td>1.9047</td>
<td>4.7143</td>
<td>8.7173</td>
<td>21.2552</td>
</tr>
<tr>
<td>Overall Std Dev RR</td>
<td>0.0</td>
<td>0.0464</td>
<td>0.0634</td>
<td>0.0588</td>
<td>0.0796</td>
<td>0.1474</td>
</tr>
<tr>
<td>Overall Spikes HR</td>
<td>0.0</td>
<td>0.0</td>
<td>4.0</td>
<td>5.711</td>
<td>10.0</td>
<td>30.0</td>
</tr>
<tr>
<td>Overall Spikes RR</td>
<td>0.0</td>
<td>20.0</td>
<td>20.0</td>
<td>19.87</td>
<td>20.0</td>
<td>20.0</td>
</tr>
<tr>
<td>Overall Dips HR</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0603</td>
<td>0.0</td>
<td>30.0</td>
</tr>
<tr>
<td>Overall Dips RR</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0268</td>
<td>0.0</td>
<td>10.0</td>
</tr>
<tr>
<td>Overall Interquartile HR</td>
<td>0.0</td>
<td>0.0</td>
<td>2.5</td>
<td>15.3</td>
<td>15.3</td>
<td>100.5</td>
</tr>
<tr>
<td>Overall Interquartile RR</td>
<td>0.0</td>
<td>0.1040</td>
<td>0.4222</td>
<td>0.3416</td>
<td>0.5408</td>
<td>0.9030</td>
</tr>
<tr>
<td>Overall LCH¹ HR</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>93.54</td>
<td>64.75</td>
<td>1740.0</td>
</tr>
<tr>
<td>Overall Skewness HR</td>
<td>-37.97</td>
<td>16.86</td>
<td>25.59</td>
<td>24.73</td>
<td>37.96</td>
<td>87.15</td>
</tr>
<tr>
<td>Overall Skewness RR</td>
<td>-41.9568</td>
<td>-0.7163</td>
<td>2.4209</td>
<td>2.3698</td>
<td>5.2969</td>
<td>89.5901</td>
</tr>
<tr>
<td>Overall ASD² HR</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0017</td>
<td>0.0</td>
<td>0.5622</td>
</tr>
<tr>
<td>Overall MADR³ HR</td>
<td>4.239</td>
<td>7.145</td>
<td>9.532</td>
<td>12.801</td>
<td>15.519</td>
<td>77.045</td>
</tr>
<tr>
<td>Overall SD⁴ Std Dev HR</td>
<td>0.0</td>
<td>1.528</td>
<td>2.244</td>
<td>3.576</td>
<td>4.205</td>
<td>31.653</td>
</tr>
<tr>
<td>Overall SD Std Dev RR</td>
<td>0.0915</td>
<td>0.2163</td>
<td>0.2306</td>
<td>0.2465</td>
<td>0.2641</td>
<td>0.6307</td>
</tr>
</tbody>
</table>

4.2 Data labeling

After labeling the simulated data following the process described in section 3.5.2, the resulting dataset had the distribution of QoL labels presented in table 4.2. A few more statistics on these labels are found in table 4.3. There is a clear prevalence of a middle-range QoL score, between 5 and 6. This is possibly due to the aforementioned simulation conditions.

¹Longest continuous high
²Average seconds difference
³Mean absolute difference from rest
⁴Seconds difference
Results and Discussion

Table 4.2: Distribution of QoL labels in simulated dataset

<table>
<thead>
<tr>
<th>Value</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency</td>
<td>107</td>
<td>116</td>
<td>115</td>
<td>118</td>
<td>133</td>
<td>1365</td>
<td>74</td>
<td>68</td>
<td>69</td>
<td>73</td>
</tr>
<tr>
<td>Proportion</td>
<td>0.048</td>
<td>0.052</td>
<td>0.051</td>
<td>0.053</td>
<td>0.059</td>
<td>0.610</td>
<td>0.033</td>
<td>0.030</td>
<td>0.031</td>
<td>0.033</td>
</tr>
</tbody>
</table>

Table 4.3: Statistics on QoL labels of simulated data

<table>
<thead>
<tr>
<th>QoL labels</th>
<th>Minimum</th>
<th>First Quartile</th>
<th>Median</th>
<th>Mean</th>
<th>Third Quartile</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>5.0</td>
<td>6.0</td>
<td>5.551</td>
<td>6.0</td>
<td>10.0</td>
<td>10.0</td>
</tr>
</tbody>
</table>

### 4.3 Clustering of simulated data

The simulated data was clustered only once because the labels had no influence in the results, so there was no need for multiple runs of random labeling or one run of labeling based on features. The entropy of the labeling also had no effect on this process, so it can be ignored.

The performance metrics for the clustering results are presented in tables 4.4, 4.5 and 4.6, depending on how many clusters were defined.

Table 4.4: Clustering results with 3 clusters for simulated data

<table>
<thead>
<tr>
<th>Metric</th>
<th>Hierarchical clustering</th>
<th>K-Means</th>
<th>PAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connectivity</td>
<td>36.3012</td>
<td>127.5810</td>
<td>315.5734</td>
</tr>
<tr>
<td>Silhouette width</td>
<td>0.6496</td>
<td>0.5507</td>
<td>0.3854</td>
</tr>
<tr>
<td>Dunn index</td>
<td>0.1064</td>
<td>0.0454</td>
<td>0.0185</td>
</tr>
<tr>
<td>APN</td>
<td>0.0048</td>
<td>0.0019</td>
<td>0.0072</td>
</tr>
<tr>
<td>AD</td>
<td>240.9840</td>
<td>191.6066</td>
<td>182.3166</td>
</tr>
<tr>
<td>ADM</td>
<td>4.5843</td>
<td>0.6427</td>
<td>1.6891</td>
</tr>
<tr>
<td>FOM</td>
<td>5.6655</td>
<td>4.9790</td>
<td>4.8869</td>
</tr>
</tbody>
</table>

Table 4.5: Clustering results with 4 clusters for simulated data

<table>
<thead>
<tr>
<th>Metric</th>
<th>Hierarchical clustering</th>
<th>K-Means</th>
<th>PAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connectivity</td>
<td>39.5766</td>
<td>201.7508</td>
<td>294.7</td>
</tr>
<tr>
<td>Silhouette width</td>
<td>0.6437</td>
<td>0.5091</td>
<td>0.3809</td>
</tr>
<tr>
<td>Dunn index</td>
<td>0.1214</td>
<td>0.0613</td>
<td>0.0159</td>
</tr>
<tr>
<td>APN</td>
<td>0.0070</td>
<td>0.0307</td>
<td>0.0083</td>
</tr>
<tr>
<td>AD</td>
<td>239.4118</td>
<td>185.7641</td>
<td>164.2692</td>
</tr>
<tr>
<td>ADM</td>
<td>4.4861</td>
<td>8.1203</td>
<td>1.9604</td>
</tr>
<tr>
<td>FOM</td>
<td>5.5948</td>
<td>4.8330</td>
<td>4.5972</td>
</tr>
</tbody>
</table>
Results and Discussion

Table 4.6: Clustering results with 5 clusters for simulated data

<table>
<thead>
<tr>
<th>Metric</th>
<th>Hierarchical clustering</th>
<th>K-Means</th>
<th>PAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connectivity</td>
<td>43.8270</td>
<td>315.6806</td>
<td>381.8222</td>
</tr>
<tr>
<td>Silhouette width</td>
<td>0.6303</td>
<td>0.3898</td>
<td>0.3365</td>
</tr>
<tr>
<td>Dunn index</td>
<td>0.1286</td>
<td>0.0396</td>
<td>0.0113</td>
</tr>
<tr>
<td>APN</td>
<td>0.0116</td>
<td>0.0015</td>
<td>0.0039</td>
</tr>
<tr>
<td>AD</td>
<td>237.4655</td>
<td>159.3141</td>
<td>156.9211</td>
</tr>
<tr>
<td>ADM</td>
<td>5.7460</td>
<td>0.4061</td>
<td>0.8761</td>
</tr>
<tr>
<td>FOM</td>
<td>5.4207</td>
<td>4.3789</td>
<td>4.4272</td>
</tr>
</tbody>
</table>

Given the higher performance of hierarchical clustering in comparison with the other methods, some statistics of the results of this method are presented in tables 4.7, 4.8, 4.9 and 4.10 for the most noteworthy features, which show some significant differences between clusters. The cluster dendrogram is shown in figure 4.1.

Table 4.7: Distribution of instances in clusters for hierarchical clustering

<table>
<thead>
<tr>
<th>Number of instances</th>
<th>Cluster 1</th>
<th>Cluster 2</th>
<th>Cluster 3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1999</td>
<td>229</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 4.8: Statistics of most noteworthy features for cluster 1 resulting from hierarchical clustering

<table>
<thead>
<tr>
<th>Feature</th>
<th>Minimum</th>
<th>First Quartile</th>
<th>Median</th>
<th>Mean</th>
<th>Third Quartile</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overall Mean HR</td>
<td>55.0</td>
<td>63.08</td>
<td>64.54</td>
<td>67.26</td>
<td>69.92</td>
<td>109.26</td>
</tr>
<tr>
<td>Overall Max HR</td>
<td>55.0</td>
<td>74.0</td>
<td>91.0</td>
<td>105.2</td>
<td>138.0</td>
<td>180.0</td>
</tr>
<tr>
<td>Overall Std Dev RR</td>
<td>0.0</td>
<td>0.0478</td>
<td>0.0640</td>
<td>0.0599</td>
<td>0.0802</td>
<td>0.1474</td>
</tr>
<tr>
<td>Overall Spikes HR</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>4.609</td>
<td>9.0</td>
<td>30.0</td>
</tr>
<tr>
<td>Overall Dips HR</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0050</td>
<td>0.0</td>
<td>1.0</td>
</tr>
<tr>
<td>Overall LCH HR</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>35.04</td>
<td>14.0</td>
<td>390.0</td>
</tr>
</tbody>
</table>

Table 4.9: Statistics of most noteworthy features for cluster 2 resulting from hierarchical clustering

<table>
<thead>
<tr>
<th>Feature</th>
<th>Minimum</th>
<th>First Quartile</th>
<th>Median</th>
<th>Mean</th>
<th>Third Quartile</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overall Mean HR</td>
<td>77.38</td>
<td>84.13</td>
<td>87.94</td>
<td>89.77</td>
<td>93.98</td>
<td>120.15</td>
</tr>
<tr>
<td>Overall Max HR</td>
<td>139.0</td>
<td>149.0</td>
<td>156.0</td>
<td>157.0</td>
<td>163.0</td>
<td>180.0</td>
</tr>
<tr>
<td>Overall Std Dev RR</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0599</td>
<td>0.0599</td>
<td>0.0762</td>
<td>0.1219</td>
</tr>
<tr>
<td>Overall Spikes HR</td>
<td>6.0</td>
<td>12.0</td>
<td>14.0</td>
<td>14.79</td>
<td>20.0</td>
<td>30.0</td>
</tr>
<tr>
<td>Overall Dips HR</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.3275</td>
<td>0.0</td>
<td>20.0</td>
</tr>
<tr>
<td>Overall LCH HR</td>
<td>330.0</td>
<td>420.0</td>
<td>506.0</td>
<td>548.3</td>
<td>651.0</td>
<td>1050.0</td>
</tr>
</tbody>
</table>
Results and Discussion

Figure 4.1: Dendrogram resulting from hierarchical clustering of simulated data

Table 4.10: Statistics of most noteworthy features for cluster 3 resulting from hierarchical clustering

<table>
<thead>
<tr>
<th>Feature</th>
<th>Minimum</th>
<th>First Quartile</th>
<th>Median</th>
<th>Mean</th>
<th>Third Quartile</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overall Mean HR</td>
<td>107.0</td>
<td>116.5</td>
<td>123.8</td>
<td>124.4</td>
<td>130.9</td>
<td>146.5</td>
</tr>
<tr>
<td>Overall Max HR</td>
<td>146.0</td>
<td>154.0</td>
<td>163.0</td>
<td>163.0</td>
<td>171.5</td>
<td>180.0</td>
</tr>
<tr>
<td>Overall Std Dev RR</td>
<td>0.0</td>
<td>5.551e-17</td>
<td>5.551e-17</td>
<td>5.551e-17</td>
<td>5.551e-17</td>
<td>1.110e-16</td>
</tr>
<tr>
<td>Overall Spikes HR</td>
<td>0</td>
<td>20</td>
<td>20</td>
<td>18</td>
<td>20</td>
<td>30</td>
</tr>
<tr>
<td>Overall Dips HR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>5</td>
<td>0</td>
<td>30</td>
</tr>
<tr>
<td>Overall LCH HR</td>
<td>1030</td>
<td>1245</td>
<td>1360</td>
<td>1375</td>
<td>1505</td>
<td>1740</td>
</tr>
</tbody>
</table>

4.4 Classification of randomly labeled data

This step was performed as a benchmark for the next experiments of classification with data labeled from features. Since, in this case, the data was labeled randomly, there were several runs of the labeling and of the creation of models, to get more accurate performance metrics. Given that the random labeling’s entropy was adjustable, five runs were executed for each of two levels of entropy (0.5 and 1), as well as another run with entropy at 0. Entropy at 0 corresponds to labeling
Results and Discussion

done sequentially, with no random labels, while entropy at 1 corresponds to completely random labels.

The performance metrics for data labeled randomly with entropy at 0 are presented in table 4.12. The averaged performance metrics for data labeled randomly with entropy at 0.5 and with entropy at 1 are shown in tables 4.13 and 4.14, respectively. Table 4.11 offers a refresher of the ideal values for the performance metrics, which were explained in greater detail in section 3.6.2.2.

Table 4.11: Ideal values for classification performance metrics

<table>
<thead>
<tr>
<th>Performance metric</th>
<th>Ideal values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>Should be maximized</td>
</tr>
<tr>
<td>AUC</td>
<td>Should be maximized</td>
</tr>
<tr>
<td>Log loss</td>
<td>Should be minimized</td>
</tr>
<tr>
<td>Mean sensitivity</td>
<td>Should be maximized</td>
</tr>
<tr>
<td>Mean specificity</td>
<td>Should be maximized</td>
</tr>
<tr>
<td>Mean balanced accuracy</td>
<td>Should be maximized</td>
</tr>
</tbody>
</table>

Table 4.12: Classification results for randomly labeled data with entropy 0

<table>
<thead>
<tr>
<th>Metric</th>
<th>Random forest</th>
<th>Deep learning</th>
<th>SVM</th>
<th>FFNN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>0.0659</td>
<td>0.0995</td>
<td>0.1389</td>
<td>0.0895</td>
</tr>
<tr>
<td>AUC</td>
<td>0.4965</td>
<td>0.5</td>
<td>0.5767</td>
<td>0.4966</td>
</tr>
<tr>
<td>Log loss</td>
<td>2.3801</td>
<td>2.3026</td>
<td>2.2865</td>
<td>2.3234</td>
</tr>
<tr>
<td>Mean sensitivity</td>
<td>0.0660</td>
<td>0.1</td>
<td>0.1387</td>
<td>0.0909</td>
</tr>
<tr>
<td>Mean specificity</td>
<td>0.8962</td>
<td>0.9</td>
<td>0.9043</td>
<td>0.8990</td>
</tr>
<tr>
<td>Mean balanced accuracy</td>
<td>0.4811</td>
<td>0.5</td>
<td>0.5215</td>
<td>0.4950</td>
</tr>
</tbody>
</table>

Table 4.13: Classification results for randomly labeled data with entropy 0.5, averaged from 5 runs

<table>
<thead>
<tr>
<th>Metric</th>
<th>Random forest</th>
<th>Deep learning</th>
<th>SVM</th>
<th>FFNN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>0.2906</td>
<td>0.2958</td>
<td>0.2953</td>
<td>0.2898</td>
</tr>
<tr>
<td>AUC</td>
<td>0.5143</td>
<td>0.4992</td>
<td>0.5037</td>
<td>0.5007</td>
</tr>
<tr>
<td>Log loss</td>
<td>2.1927</td>
<td>2.2491</td>
<td>2.0020</td>
<td>1.9740</td>
</tr>
<tr>
<td>Mean sensitivity</td>
<td>0.1021</td>
<td>0.1000</td>
<td>0.1005</td>
<td>0.0999</td>
</tr>
<tr>
<td>Mean specificity</td>
<td>0.9006</td>
<td>0.9000</td>
<td>0.9002</td>
<td>0.9000</td>
</tr>
<tr>
<td>Mean balanced accuracy</td>
<td>0.5014</td>
<td>0.5000</td>
<td>0.5003</td>
<td>0.4999</td>
</tr>
</tbody>
</table>
Results and Discussion

Table 4.14: Classification results for randomly labeled data with entropy 1 averaged from 5 runs

<table>
<thead>
<tr>
<th>Metric</th>
<th>Random forest</th>
<th>Deep learning</th>
<th>SVM</th>
<th>FFNN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>0.1078</td>
<td>0.1118</td>
<td>0.1085</td>
<td>0.1087</td>
</tr>
<tr>
<td>AUC</td>
<td>0.5024</td>
<td>0.4999</td>
<td>0.4972</td>
<td>0.5007</td>
</tr>
<tr>
<td>Log loss</td>
<td>2.3989</td>
<td>2.3022</td>
<td>2.3016</td>
<td>2.3049</td>
</tr>
<tr>
<td>Mean sensitivity</td>
<td>0.1038</td>
<td>0.1000</td>
<td>0.1001</td>
<td>0.1008</td>
</tr>
<tr>
<td>Mean specificity</td>
<td>0.9005</td>
<td>0.9000</td>
<td>0.9000</td>
<td>0.9001</td>
</tr>
<tr>
<td>Mean balanced accuracy</td>
<td>0.5021</td>
<td>0.5000</td>
<td>0.5001</td>
<td>0.5004</td>
</tr>
</tbody>
</table>

4.5 Classification of data with labels extracted from features

The most important step is the creation of models with data labeled according to its features. All of the following models were created after the normalization of the data, and the deletion of any NA’s that could have been created during the normalization. Table 4.11 offers a refresher of the ideal values for the performance metrics, which were explained in greater detail in section 3.6.2.2.

4.5.1 Without feature selection

First, the models were created taking in account all 308 features. The performance metrics resulting from this are presented in table 4.15.

Table 4.15: Classification results for data labeled from features, without feature selection

<table>
<thead>
<tr>
<th>Metric</th>
<th>Random forest</th>
<th>Deep learning</th>
<th>SVM</th>
<th>FFNN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>0.6752</td>
<td>0.6089</td>
<td>0.6644</td>
<td>0.6156</td>
</tr>
<tr>
<td>AUC</td>
<td>0.9113</td>
<td>0.5000</td>
<td>—</td>
<td>0.5627</td>
</tr>
<tr>
<td>Log loss</td>
<td>0.8334</td>
<td>2.0551</td>
<td>—</td>
<td>1.4146</td>
</tr>
<tr>
<td>Mean sensitivity</td>
<td>0.2443</td>
<td>0.1000</td>
<td>0.2573</td>
<td>0.1185</td>
</tr>
<tr>
<td>Mean specificity</td>
<td>0.9519</td>
<td>0.9000</td>
<td>0.9610</td>
<td>0.9120</td>
</tr>
<tr>
<td>Mean balanced accuracy</td>
<td>0.5981</td>
<td>0.5000</td>
<td>0.5265</td>
<td>0.5152</td>
</tr>
</tbody>
</table>

4.5.2 With feature selection
Results and Discussion

Since the dataset has a very large number of features, feature selection was executed, using Recursive Feature Elimination. There were 23 selected features, which are listed in appendix B. The train function in the caret package also performs tuning of the model parameters, and the optimal parameters found for this set were the following:

- Random forest:
  - mtry (randomly selected predictors): 7;

- Deep learning:
  - layer1 (Hidden layer 1): 5;
  - layer2 (Hidden layer 2): 0;
  - layer3 (Hidden layer 3): 0;
  - hidden_dropout (Hidden dropouts): 0;
  - visible_dropout (Visible dropouts): 0;

- L2 Regularized Support Vector Machine (dual) with Linear Kernel:
  - cost: 0.5;
  - loss: L1;

- Feed-forward neural network:
  - size (Hidden units): 7;
  - decay (Weight decay): 0.01

The performance metrics for the resulting models are presented in table 4.16.

Table 4.16: Classification results for data labeled from features, with feature selection, using 23 features

<table>
<thead>
<tr>
<th>Metric</th>
<th>Random forest</th>
<th>Deep learning</th>
<th>SVM</th>
<th>FFNN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>0.7073</td>
<td>0.6099</td>
<td>0.6805</td>
<td>0.6949</td>
</tr>
<tr>
<td>AUC</td>
<td>0.9149</td>
<td>0.5000</td>
<td>—</td>
<td>0.9185</td>
</tr>
<tr>
<td>Log loss</td>
<td>0.7860</td>
<td>2.0466</td>
<td>—</td>
<td>0.7468</td>
</tr>
<tr>
<td>Mean sensitivity</td>
<td>0.3343</td>
<td>0.1000</td>
<td>0.2623</td>
<td>0.3123</td>
</tr>
<tr>
<td>Mean specificity</td>
<td>0.9644</td>
<td>0.9000</td>
<td>0.9570</td>
<td>0.9653</td>
</tr>
<tr>
<td>Mean balanced accuracy</td>
<td>0.6504</td>
<td>0.5000</td>
<td>0.6096</td>
<td>0.6388</td>
</tr>
</tbody>
</table>
4.6 Discussion

In terms of simulation of data, it is clear that the results are composed of many instances with little variation and with not many indicators of low QoL. That is due to the intention of generating data that includes healthy individuals with no variation from the sample data, as well as individuals that had an increase in variation from the sample data as weeks went on. This leads to an imbalance in the labels, which could also happen with real-world data. Since an imbalanced data set often has effects on the performance of predictive models, it is important to keep this imbalance in mind when analyzing the results.

From the statistics presented in table 4.1, we can see that even though most data was in a normal range, as evidenced by the mean and median of "Overall Mean HR", "Overall Median HR" and "Overall Mode HR", there are some instances present that deviate from the normal range, as seen in the third quartile and maximum of most features. The instances with anomalies in HR, like spikes and dips, are also not the majority, but are present enough to influence the median of the "Overall Spikes" features. There is also enough variation in the feature "Overall Std Dev RR" to be able to label using it as one of the indicators. It’s important to have it as a decision variable because with real-world data, it is one of the features that could be more decisive, according to research mentioned in chapter 2. Furthermore, there is a lot of variation in the feature "Overall LCH HR", which is important since it was used in the labeling stage to gauge the patient’s exercise frequency.

The labeling step resulted in a dataset with imbalance in terms of class representation, as shown in table 4.2. 61% of the instances were labeled as having QoL 6, due to the data simulation conditions mentioned previously. There are few instances with very low QoL and even less instances with very high QoL. This is not ideal, since this will influence the training of models, and can make them more prone to over-fitting.

Afterwards, the simulated data was clustered in order to extract more information from the features and find any natural groups of instances that might form. First, three methods of clustering were compared, and it was found that the most well performing method was hierarchical clustering across all numbers of clusters tested, as is shown in tables 4.4, 4.5 and 4.6. It produced the clusters with the best internal performance metrics, which means they are the most compact. However, in terms of stability, the best results were obtained with PAM, which provided the most consistent clusters.

Since hierarchical clustering with 3 clusters was the best performing method, the statistics of each cluster resulting from it are shown in tables 4.7, 4.8, 4.9 and 4.10. Most instances are in cluster 1, while cluster 3 has very few instances. The clusters have clear differences between them, both with HR values as well as RR. The first cluster has generally lower "Overall Mean HR" and "Overall Max HR" and is the cluster with least anomalies, with very low values in "Overall Spikes HR" and "Overall Dips HR". It also has the highest HRV of all the clusters, as evidenced by the feature "Overall Std Dev RR". "Overall LCH HR", which represents the sum of the duration of the
longest continuous highs (bpm over 125) present in the total set of 10 weeks, is also lowest in this cluster. The second cluster has mostly in-between values for every feature, except for "Overall Std Dev RR" (HRV), which is quite similar to cluster 1’s. The third cluster has higher HR, as shown by the features "Overall Mean HR", "Overall Max HR" and "Overall LCH HR". It also has a higher occurrence of anomalies (spikes and dips). Lastly, it has a notably low HRV, in the feature "Overall Std Dev RR", when compared with the other clusters. Taking these observations into account, cluster 1 seems to represent the larger majority of the dataset, with no indicators of low QoL or of high QoL. Cluster 3 is the smallest, and seems to group instances with indicators of low QoL, like a very high HR and a very low HRV. Cluster 2 can possibly represent a group of patients that exercise frequently and therefore have a higher mean HR than normal, but have healthy indicators otherwise.

The next step taken was the creation of models for randomly labeled data, to serve as a benchmark for the models created with data labeled from features. The performance results of this experiment are present in tables 4.12, 4.13 and 4.14. The performance is very low across every level of labeling entropy, but especially when data is labeled sequentially (entropy=0.5). When data is labeled in a completely random way, the performance is only as good as the probability of randomly selecting the QoL class (0.1, since there are 10 classes). The halfway level of entropy, which is a mixture of sequential and random labeling, obtains the best results. None of the models created for each entropy level obtained acceptable sensitivity and specificity levels. Using 5 runs to evaluate the performance of models created with random labeling was important in order to obtain truthful results, since there were some changes seen from run to run, due to the random nature of the process.

The model creation based on data labeled from features consisted of two steps, one without feature selection and one with feature selection, because it is a standard step in modeling, especially with a dataset so highly dimensional.

The performance results for modeling without feature selection are shown in table 4.15. All methods had a vastly better performance when compared with the previous results, obtained from randomly labeled data. Random forest was the best performing method, with the highest performance across all metrics. The worst performing method was deep learning, and it had a significantly high logarithmic loss, which means it resulted in several false classifications. SVM had the second-best performance, but its balanced accuracy was lower, which means the model might be over-fitting due to the imbalance in the dataset. Feed-forward neural network had an acceptable performance, but still has significantly low sensitivity, specificity and balanced accuracy.

The performance results for modeling with feature selection are presented in table 4.16. There’s an improvement in performance in comparison with the models created without feature selection, which means that a lot of features might have noise that decreases the performance. Once again, the best performance comes from the model created with the random forest method, achieving 71% accuracy, which is considerable given the experiment conditions. It also has the highest balanced accuracy, meaning it is the model with the least over-fitting. It has a 33% recall (or mean sensitivity), which is still considerably low, but is much better than both the results with random
Results and Discussion

labeling and the results without feature selection. The performance of the feed-forward neural
network is also worth noting, since it is a very close second, and its logarithmic loss is slightly
lower, which might mean it has fewer false positives.

In conclusion, the most successful method was random forest. This might be due to it being
more equipped to deal with imbalanced datasets than other methods. Despite this, it still struggled
to achieve good recall, which is to be expected with the nature of the dataset, and all the experiment
limitations. Feed-forward neural networks are also a method to consider for this task, since it
achieved high performance as well. SVM is a high-performing method as well, but it takes a lot
of experimentation with different types of kernel to achieve acceptable results.
Results and Discussion
Chapter 5

Conclusions and Future Work

This section will present the conclusions of the Dissertation that was completed. It will summarize the main results obtained, as well as describe the limitations of this study, the satisfaction of the goals that were defined, and it will offer suggestions for future work.

5.1 Results

The work developed in this dissertation is divided into data generation, and modeling and clustering.

In terms of data generation, a large dataset was obtained, with several transformations performed on it, which was the initial goal of this step. The data had an imbalance that was a consequence of the generation procedure, yet this procedure was executed in a way to obtain the most realistic progression of data. The labeling performed on the simulated data was implemented taking research into account as much as possible, and resulted in a dataset with a high prevalence of instances classified as having QoL 6 (61%), due to the imbalanced nature of the generated data.

Regarding the modeling and clustering results, despite the limitations imposed on the process, there were satisfactory results, especially when using feature selection to filter out noisy data. The method that achieved most success was random forest, having 71% accuracy, 33% recall and 65% balanced accuracy. Feed-forward neural networks also performed well, being a close second to random forest. While SVM had a notably good performance, the selection of the ideal kernel is time-consuming and did not have a significant payoff in terms of performance.
Conclusions and Future Work

5.2 Study limitations

The experiments carried out throughout this dissertation were severely limited by the lack of real-world data. The absence of patient data made it impossible to analyze the influence of each physiological signal’s features on QoL, and the objective was changed to an experimentation with the theoretical tolerance of models to simulated data and randomized labels.

5.3 Satisfaction of goals

All goals set out for this work were achieved, with the creation of a signal transformation tool, the generation of HR and RR data and the labeling of it, as well as the modeling and clustering of this data. The results obtained point towards the possibility of using the explored methods in the prediction of QoL from physiological data.

5.4 Suggestions for future work

In order to further research solutions for this problem, it will be necessary to collect real world data from patients with varied levels of QoL, ideally with a balanced dataset. Given the research mentioned in chapter 2, it would be important that this data provided accurate heart rate variability values, since these have been shown to be very informative. Another important indicator is resting HR, but in order for it to be of predictive value, the HR recorded should be measured only during rest, since measurements throughout the day could possibly include values during exertion that would influence the overall HR recorded, and therefore influence the predictive models. The collection of data besides physiological signals could also be of great use, and it could include information like age, gender, exercise frequency, previous disease history, among others. This information has an influence on how the physiological data should be interpreted and could make predictive models significantly more accurate.
References


[DNG+00] Rod K. Dishman, Yoshio Nakamura, Melissa E. Garcia, Ray W. Thompson, Andrea L. Dunn, and Steven N. Blair. Heart rate variability, trait anxiety, and perceived


46
REFERENCES


[MJH+11] Shie Mannor, Xin Jin, Jiawei Han, Xin Jin, Jiawei Han, Xin Jin, Jiawei Han, and Xinhua Zhang. K-Medoids Clustering. In *Encyclopedia of Machine Learning*, pages 564–565. Springer US, Boston, MA, 2011.


REFERENCES


Appendix A

Further statistics of simulated data

In table A.1, some statistics of all overall features of the simulated data are presented.
Further statistics of simulated data

Table A.1: Statistics of overall features of simulated data

<table>
<thead>
<tr>
<th>Feature</th>
<th>Minimum</th>
<th>First Quartile</th>
<th>Median</th>
<th>Mean</th>
<th>Third Quartile</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overall Mean HR</td>
<td>55.0</td>
<td>63.11</td>
<td>65.43</td>
<td>69.82</td>
<td>73.40</td>
<td>146.53</td>
</tr>
<tr>
<td>Overall Mean RR</td>
<td>0.2221</td>
<td>0.7368</td>
<td>0.8657</td>
<td>0.8738</td>
<td>1.0058</td>
<td>1.5710</td>
</tr>
<tr>
<td>Overall Median HR</td>
<td>55.0</td>
<td>63.0</td>
<td>63.0</td>
<td>64.15</td>
<td>63.0</td>
<td>159.0</td>
</tr>
<tr>
<td>Overall Median RR</td>
<td>0.20</td>
<td>0.8296</td>
<td>0.8794</td>
<td>0.8745</td>
<td>0.9209</td>
<td>1.60</td>
</tr>
<tr>
<td>Overall Mode HR</td>
<td>55.0</td>
<td>62.0</td>
<td>63.0</td>
<td>65.46</td>
<td>63.0</td>
<td>180.0</td>
</tr>
<tr>
<td>Overall Mode RR</td>
<td>0.20</td>
<td>0.20</td>
<td>0.8960</td>
<td>0.8909</td>
<td>1.60</td>
<td>1.60</td>
</tr>
<tr>
<td>Overall Min HR</td>
<td>55.0</td>
<td>55.0</td>
<td>60.0</td>
<td>58.11</td>
<td>60.0</td>
<td>68.0</td>
</tr>
<tr>
<td>Overall Min RR</td>
<td>0.0996</td>
<td>0.0996</td>
<td>0.0996</td>
<td>0.1227</td>
<td>0.0996</td>
<td>1.1773</td>
</tr>
<tr>
<td>Overall Max HR</td>
<td>55.0</td>
<td>74.0</td>
<td>103.0</td>
<td>110.7</td>
<td>145.0</td>
<td>180.0</td>
</tr>
<tr>
<td>Overall Max RR</td>
<td>0.916</td>
<td>1.775</td>
<td>1.775</td>
<td>1.741</td>
<td>1.775</td>
<td>1.775</td>
</tr>
<tr>
<td>Overall Std Dev HR</td>
<td>0.0</td>
<td>0.0427</td>
<td>1.9047</td>
<td>4.7143</td>
<td>8.7173</td>
<td>21.2552</td>
</tr>
<tr>
<td>Overall Std Dev RR</td>
<td>0.0</td>
<td>0.0464</td>
<td>0.0634</td>
<td>0.0588</td>
<td>0.0796</td>
<td>0.1474</td>
</tr>
<tr>
<td>Overall Spikes HR</td>
<td>0.0</td>
<td>0.0</td>
<td>4.0</td>
<td>5.711</td>
<td>10.0</td>
<td>30.0</td>
</tr>
<tr>
<td>Overall Spikes RR</td>
<td>0.0</td>
<td>20.0</td>
<td>20.0</td>
<td>19.87</td>
<td>20.0</td>
<td>20.0</td>
</tr>
<tr>
<td>Overall Dips HR</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0603</td>
<td>0.0</td>
<td>3.0</td>
</tr>
<tr>
<td>Overall Dips RR</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0268</td>
<td>0.0</td>
<td>10.0</td>
</tr>
<tr>
<td>Overall Interquartile HR</td>
<td>0.0</td>
<td>0.0</td>
<td>2.5</td>
<td>15.3</td>
<td>15.3</td>
<td>100.5</td>
</tr>
<tr>
<td>Overall Interquartile RR</td>
<td>0.0</td>
<td>0.1040</td>
<td>0.4222</td>
<td>0.3416</td>
<td>0.5408</td>
<td>0.9030</td>
</tr>
<tr>
<td>Overall LCH HR</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>93.54</td>
<td>64.75</td>
<td>1740.0</td>
</tr>
<tr>
<td>Overall LCH RR</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Overall Skewness HR</td>
<td>-37.97</td>
<td>16.86</td>
<td>25.59</td>
<td>24.73</td>
<td>37.96</td>
<td>87.15</td>
</tr>
<tr>
<td>Overall Skewness RR</td>
<td>-41.9568</td>
<td>-0.7163</td>
<td>2.4209</td>
<td>2.3698</td>
<td>5.2969</td>
<td>89.5901</td>
</tr>
<tr>
<td>Overall ASD HR</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0017</td>
<td>0.0</td>
<td>0.5622</td>
</tr>
<tr>
<td>Overall ASD RR</td>
<td>0.0</td>
<td>8.259e-05</td>
<td>8.259e-05</td>
<td>1.200e-04</td>
<td>8.259e-05</td>
<td>6.237e-03</td>
</tr>
<tr>
<td>Overall MADR HR</td>
<td>4.239</td>
<td>7.145</td>
<td>9.532</td>
<td>12.801</td>
<td>15.519</td>
<td>77.045</td>
</tr>
<tr>
<td>Overall MADR RR</td>
<td>0.1626</td>
<td>0.3134</td>
<td>0.3565</td>
<td>0.3877</td>
<td>0.4391</td>
<td>0.9710</td>
</tr>
<tr>
<td>Overall SD Std Dev HR</td>
<td>0.0</td>
<td>1.528</td>
<td>2.244</td>
<td>3.576</td>
<td>4.205</td>
<td>31.653</td>
</tr>
<tr>
<td>Overall SD Std Dev RR</td>
<td>0.0915</td>
<td>0.2163</td>
<td>0.2306</td>
<td>0.2465</td>
<td>0.2641</td>
<td>0.6307</td>
</tr>
</tbody>
</table>
Appendix B

Selected features

The features that were selected after recursive feature elimination were the following:

- "TotalMeanHR"
- "TotalMeanAbsDevFromRestHR"
- "MeanHRW9"
- "MeanHRW10"
- "MeanAbsDevFromRestHRW9"
- "MeanHRW8"
- "MeanAbsDevFromRestHRW8"
- "StdDevRRW8"
- "TotalLongestContinuousHighHR"
- "LongestContinuousHighHRW9"
- "LongestContinuousHighHRW8"
- "LongestContinuousHighHRW10"
- "LongestContinuousHighHRW7"
- "TotalSpikesHR"
- "IQRHRW10"
- "IQRHRW4"
- "MeanAbsDevFromRestHRW10"
Selected features

- "StdDevRRW9"
- "IQRHRW5"
- "IQRHRW6"
- "IQRRRW3"
- "IQRRRW4"
Appendix C

Images of developed tools

Figure C.1: Interface of signal transformation tool, used for transformation of data, visualization and data generation

Figure C.2: Interface of signal transformation tool, including filters and log of changes