FEDERAL UNIVERSITY OF TECHNOLOGY — PARANÁ
GRADUATE PROGRAM IN BIOMEDICAL ENGINEERING

RUI PIMENTEL LEITE

AUTOMATIC ESTIMATION OF CANINE HEART MURMUR WITH ELECTRONIC STETHOSCOPE AND SOFTWARE

DISSERTATION

CURITIBA
2019

RUI PIMENTEL LEITE

AUTOMATIC ESTIMATION OF CANINE HEART MURMUR WITH ELECTRONIC STETHOSCOPE AND SOFTWARE

Dissertation presented to the Graduate Program in Biomedical Engineering of Federal University of Technology — Paraná, in partial fulfillment of the requirements for the degree of "Mestre".

Concentration area: Biomedical engineering

Research line: Informatics applied to health

Advisor: Prof. Dr. Sergio Leandro Stebel
Federal University of Technology — Paraná

Coadvisor: Prof. Dr. Miguel Antonio Sovierzoski
Federal University of Technology — Paraná

CURITIBA
2019

Dados Internacionais de Catalogação na Publicação

Leite, Rui Pimentel
Automatic estimation of canine heart murmur with electronic stethoscope and software [recurso eletrônico] / Rui Pimentel Leite.-- 2019.
1 arquivo texto (55 f.) : PDF ; 2,46 MB.

Modo de acesso: World Wide Web.
Texto em inglês com resumo em português.
Dissertação (Mestrado) - Universidade Tecnológica Federal do Paraná. Programa de Pós-Graduação em Engenharia Biomédica, Curitiba, 2019.
Bibliografia: f. 50-52.

1. Engenharia biomédica - Dissertações. 2. Algoritmos. 3. Coração - Sons - Análise. 4. Sopros cardíacos. 5. Cães - Doenças. 6. Auscultação. 7. Estetoscópios. 8. Aprendizado do computador. 9. Análise numérica. 10. Métodos de simulação. 11. Medicina veterinária de pequenos animais - Instrumentos e aparelhos médicos. I. Stebel, Sérgio Leandro, orient. II. Sovierzoski, Miguel Antonio, coorient. III. Universidade Tecnológica Federal do Paraná. Programa de Pós-graduação em Engenharia Biomédica. IV. Título.

CDD: Ed. 23 -- 610.28

Biblioteca Central do Câmpus Curitiba - UTFPR
Bibliotecária: Luiza Aquemi Matsumoto CRB-9/794

Ministério da Educação
Universidade Tecnológica Federal do Paraná
Diretoria de Pesquisa e Pós-Graduação

TERMO DE APROVAÇÃO DE DISSERTAÇÃO Nº125

A Dissertação de Mestrado intitulada “Estimativa automática do grau do sopro cardíaco canino com estetoscópio eletrônico e software”, defendida em sessão pública pelo(a) candidato(a) Rui Pimentel Leite, no dia 31 de maio de 2019, foi julgada para a obtenção do título de Mestre em Ciências, área de concentração Engenharia Biomédica, e aprovada em sua forma final, pelo Programa de Pós-Graduação em Engenharia Biomédica.

BANCA EXAMINADORA:

Sergio Leandro Stebel, Dr – UTFPR

Gustavo Benvenutti Borba, Dr – UTFPR

Marcos Vinicio Haas Rambo, Dr – UFPR

A via original deste documento encontra-se arquivada na Secretaria do Programa, contendo a assinatura da Coordenação após a entrega da versão corrigida do trabalho.

Curitiba, 31 de maio de 2019.

Carimbo e Assinatura do(a) Coordenador(a) do Programa

I dedicate this research to family and friends, who have never failed to provide me with all the support I needed to continue my studies. To you, from the bottom of my heart, thank you.

ACKNOWLEDGMENT

First of all, I sincerely thank my father and mother, Paulo Roberto Leite and Liane Ambrosio Pimentel Leite, the two people who undoubtedly, through much effort, removed any obstacles to my academic, professional and personal trajectory.

It is also imperative to thank my dear Carolina Kimie Idehama, with all my heart and soul, for the immense help that sustains an entire dimension of my life (far beyond this research).

To Professors Sergio Leandro Stebel and Miguel Antonio Sovierzoski, thank you very much for the hours and hours of class and orientation. I hope to have transformed your patience into satisfaction with the result.

I thank Camila Cristina Antunes Negrão de Andrade for the help and for the frequent productive discussions about the research, and like her, I would also like to acknowledge the animals, the tutors and the professionals involved in this work.

Of course, I also thank my reviewer, Letícia Oliveti, whose experience has improved the text greatly.

And finally, I acknowledge all those who, more or less directly, have been part of this journey with me.

Although the Cosmos are unmeasurably vast and complex, building on preceding generations' explorations, humanity has been able to unlock the Universe's secrets. What will YOU discover?! (SCHMIDT, Brian P.).

RESUMO

LEITE, Rui Pimentel. Estimativa Automática do Grau de Sopro Cardíaco Canino com Estetoscópio Eletrônico e *Software*. 2019. 73 p. Dissertação – Programa de Pós-Graduação em Engenharia Biomédica, Universidade Tecnológica Federal do Paraná. Curitiba, 2019.

Os sons cardíacos carregam informações sobre a saúde das estruturas cardíacas e sobre a dinâmica da circulação sanguínea. A auscultação é a atividade dedicada à análise dos sons produzidos espontaneamente pelos órgãos — inclusive os sons cardíacos. Embora seja importante para detecção precoce de doenças e para o aprimoramento do emprego de recursos na área da saúde, tanto humana quanto animal, a prática da auscultação ainda é largamente subjetiva e dependente de fatores como experiência, técnica e habilidade auditiva do profissional, bem como de ruídos ambientais e de fricção. O presente trabalho representa um esforço no sentido de incrementar o ferramental para prática de medicina veterinária ao propor um novo método para estimar automaticamente o grau de sopro cardíaco canino numa escala de 0 a 6 a partir de uma gravação de sons cardíacos com marcação dos principais batimentos cardíacos. Tal método inclui a geração de *features* a partir dessa gravação (através de algoritmos especialmente adaptados) e o uso de técnicas de aprendizado de máquina para efetuar a estimativa. O método proposto foi desenvolvido a partir de um conjunto de gravações de 56 pacientes, tendo cada um dos quais 9 ciclos cardíacos processados. Das 4032 linhas de dados resultantes do estudo, apenas 0,35% delas não puderam ter o grau de sopro corretamente estimado. Uma análise do perfil de erro permitiu concluir que a estimativa e o grau real do paciente nunca atingem uma diferença de 2 graus ou mais, e que técnicas podem ser pesquisadas para reduzir o erro ou para aumentar a capacidade de generalização do método desenvolvido.

Palavras-chave: Auscultação. Grau de sopro. Estimativa automática. Aprendizado de máquina. Estetoscópio eletrônico.

ABSTRACT

LEITE, Rui Pimentel. Automatic Estimation of Canine Heart Murmur with Electronic Stethoscope and Software. 2019. 73 p. Dissertation – Graduate Program in Biomedical Engineering, Federal University of Technology — Paraná. Curitiba, 2019.

Heart sounds carry information about the health of heart structures and about the dynamics of blood circulation. Auscultation is the activity dedicated to the analysis of sounds produced spontaneously by organs — which includes heart sounds. Despite being important to detect diseases early and to improve the use of resources in the health sector, both human and animal, the practice of auscultation is still largely subjective and dependent on factors such as experience, technique and auditory ability of the professional, as well as environmental noise and friction. The present work represents an effort to increase tooling for veterinary medicine practice by proposing a new method to automatically estimate the degree of canine heart murmur on a scale of 0 to 6 from a record of heart sounds marked with the position of the main heartbeats. The method includes the generation of features from this record (through specially adapted algorithms) and the use of machine learning techniques to make the estimation. The proposed method was developed from a set of records of 56 patients, each of which had 9 heart cycles processed. Of the 4032 rows of data resulting from the study, only 0.35% of them could not have the degree of murmur adequately estimated. An analysis of the error profile allows one to conclude that the estimated and the actual degree of murmur of the patient never reach a difference of 2 degrees or more, and that further techniques can be researched to reduce the error even more or to increase the generalization capacity of the developed method.

Keywords: Auscultation. Degree of murmur. Automatic estimation. Machine learning. Electronic stethoscope.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

ADC	Analog-to-Digital Converter
CD	Cycle Duration
CE	Cycle Energy
CEUA	*Comissão de Ética no Uso de Animais*
CSV	Comma Separated Values
DS	Downsampling
EMD	Empirical Mode Decomposition
GB	Gigabytes (unit)
GHz	Gigahertz (unit)
HSSR.C	Heart Sound to Signal Ratio based on the Correlation method
HT	Hilbert Transform
HVD	Hilbert Vibration Decomposition
Hz	Hertz (unit)
IMF	Intrinsic Mode Function
k-NN	k-Nearest Neighbors
kHz	Kilohertz (unit)
ms	Milliseconds (unit)
MAE	Mean Absolute Error
MICE	Minimum InterCycle Error
NIMF	Number of IMFs
NNIMF.C	Number of Noise IMFs based on the Correlation method
NPM	NodeJS Package Manager
NSR.C	Noise to Signal Ratio based on the Correlation method
RA12	Relative Amplitude of the peaks of S1 and S2

s	Seconds (unit)
S1	[Heart] Sound 1
S2	[Heart] Sound 2
SRT	Systole Relative Time
UTFPR	*Universidade Tecnológica Federal do Paraná*
WAV	Waveform Audio File Format
μs	Microseconds (unit)

LIST OF SYMBOLS

c	Number of heart cycles to be combined in unique pairs in Equation (15)
CC_{XY}	Correlation coefficient between actual and expected outputs in Equation (4)
CD_i	Value of feature CD for the cycle of index i in Section 3.2
$CE.HVD_i$	Value of feature CE.HVD for the cycle of index i in Section 3.3
$CE.DS_i$	Value of feature CE.DS for the cycle of index i in Section 3.3
$d(v_A, v_B)$	Generic distance between two generic vectors v_A and v_B from the Subsection 3.5.1 onwards
D	Number of entries of $v[i]$ from the Subsection 3.5.1 onwards
E_x	Energy of a discrete signal $x[n]$ in Equations (2) and (3)
f	Function able to estimate outputs in a supervised learning problem in Section 2.5
f_{DS}	Resulting sampling frequency after downsampling in Section 2.2 and Subsection 3.1.1
f_S	Sampling frequency in Section 2.2 and Subsection 3.1.1
$h_{m;i}[j*]$	Vector of the component of index m resulting from the HVD decomposition of $x_i[j*]$ from the Subsection 3.1.1 onwards
i	Instance index in Equations (4) and (5)
i	Index of entries of $v[i]$ from the Subsection 3.5.1 onwards
j	Index of a sample in the vector of an input audio from the Subsection 3.1.1 onwards
$j*$	Index of a sample of a segment of the vector of an input audio from the Subsection 3.1.1 onwards
$j_{end;i}$	End index of the cycle of index i in the vector of an input audio from the Subsection 3.1.1 onwards
$j_{sta;i}$	Start index of the cycle of index i in the vector of an input audio from the Subsection 3.1.1 onwards

$j_{maxS1;i}$	Index of the sample of maximum value in the vicinity of the S1 of the cycle of index i from the Subsection 3.1.1 onwards
$j_{maxS2;i}$	Index of the sample of maximum value in the vicinity of the S2 of the cycle of index i from the Subsection 3.1.1 onwards
$j_{S1;i}$	Index of occurrence of S1 in the cycle of index i in the vector of an input audio from the Subsection 3.1.1 onwards
$j_{S2;i}$	Index of occurrence of S2 in the cycle of index i in the vector of an input audio from the Subsection 3.1.1 onwards
k	Index of a sample in a downsampled vector from the Subsection 3.1.1 onwards
k	Number of cross-validation sets in Section 2.5
$k*$	Index of a sample of a segment of a downsampled vector from the Subsection 3.1.1 onwards
$k_{end;i}$	End index of the cycle of index i in the downsampled vector of an input audio from the Subsection 3.1.1 onwards
$k_{sta;i}$	Start index of the cycle of index i in the downsampled vector of an input audio from the Subsection 3.1.1 onwards
$k_{maxS1;i}$	Index of the sample of maximum value in the vicinity of S1 of the cycle of index i in the downsampled vector from the Subsection 3.1.1 onwards
$k_{maxS2;i}$	Index of the sample of maximum value in the vicinity of S2 of the cycle of index i in the downsampled vector from the Subsection 3.1.1 onwards
$k_{S1;i}$	Index of occurrence of S1 in the cycle of index i in the downsampled vector of an input audio from the Subsection 3.1.1 onwards
$k_{S2;i}$	Index of occurrence of S2 in the cycle of index i in the downsampled vector of an input audio from the Subsection 3.1.1 onwards
K	Number of nearest neighbors of k-NN from the Subsection 3.5.1 onwards
l	Index of a function $y_{l;i}[k*]$ from the Subsection 3.1.1 onwards
l	Index of a row in the features table from the Subsection 3.5.1 onwards
l_{max}	Index l of the vector $y_{l;i}[k*]$ with the highest correlation coefficient $\rho_{l;i}$ with $x_{DS;i}[k*]$ from the Section 3.3 onwards
L	Number of samples of a discrete signal $x[n]$ in Equation (3)

L_i	Number of functions $y_{l;i}[k*]$ from the Subsection 3.1.1 onwards
m	Index of a component $h_{m;i}[j*]$ from the Subsection 3.1.2 onwards
M	Number of features in Figure 2
$MAE(X, Y)$	Mean absolute error between the actual and expected outputs in Equation (5)
n	Index of the samples of a discrete signal $x[n]$ in Equations (2) and (3)
n	Number of known instances in Equation (4)
N	Number of heart cycles in Figure 2
N	Number of samples of $x[n]$ to be computed in Equation (2)
$NIMF_i$	Value of feature NIMF for the cycle of index i in Section 3.3
$p(c)$	Number of combinations of heart cycles in unique pairs in Equation (15)
P	Order of Minkowski distance from the Subsection 3.5.1 onwards
$RA12_i$	Value of feature RA12 for the cycle of index i in Section 3.2
S	Downsampling factor in Section 2.2
SRT_i	Value of feature SRT for the cycle of index i in Section 3.2
t	Time instant from the Subsection 3.1.1 onwards
$t_{end;i}$	End time (in seconds) of the cycle of index i in the input audio from the Subsection 3.1.1 onwards
$t_{sta;i}$	Start time (in seconds) of the cycle of index i in the input audio from the Subsection 3.1.1 onwards
$t_{S1;i}$	Time (in seconds) of occurrence of S1 in the cycle of index i in the input audio from the Subsection 3.1.1 onwards
$t_{S2;i}$	Time (in seconds) of occurrence of S2 in the cycle of index i in the input audio from the Subsection 3.1.1 onwards
$v[i]$	Generic vector of entries indexed by i from the Subsection 3.5.1 onwards
$x[j]$	Sample of index j in the sample vector of an input audio from the Subsection 3.1.1 onwards
$x[n]$	Generic discrete signal in Equations (2) and (3)

$x_{DS;i}[k*]$	Vector resulting from the downsampling of $x_i[j*]$ from the Subsection 3.1.1 onwards
$x_{HS.C;i}[k*]$	Vector resulting from the sum of the functions $y_{l;i}[k*]$ considered heart sounds by the correlation method in Section 3.3
$x_i[j*]$	Segment corresponding to the cycle of index i in the sample vector of an input audio from the Subsection 3.1.1 onwards
$x_{N.C;i}[k*]$	Vector resulting from the sum of the functions $y_{l;i}[k*]$ considered noise by the correlation method in Section 3.3
X	Set of possible entries in a supervised learning problem in Section 2.5
X	Vector of the expected outputs in Equations (4) and (5)
X_i	Expected output for the instance of index i in Equations (4) and (5)
$\overline{X}$	Mean of the expected outputs of all instances in Equation (4)
$y_{l;i}[k*]$	Vector of a function (residual or IMF) of index l resulting from the EMD decomposition of $x_{DS;i}[k*]$ from the Subsection 3.1.1 onwards
Y	Set of possible outputs in a supervised learning problem in Section 2.5
Y	Vector of the outputs estimated in Equations (4) and (5)
Y_i	Estimated output for the instance of index i in Equations (4) and (5)
$\overline{Y}$	Mean of the estimated outputs of all instances in Equation (4)
$\rho_{l;i}$	Correlation coefficient between the vectors $y_{l;i}[k*]$ and $x_{DS;i}[k*]$ from the Subsection 3.1.1 onwards

TABLE OF CONTENTS

1 INTRODUCTION

The heart is a muscular organ, present inside the thoracic cavity, which composes the cardiovascular system together with the blood vessels. Each contraction (**systole**) and relaxation (**diastole**) sequence of heart is denominated **heart cycle** (REECE et al., 2018). Such activity imparts a series of vibrations to the surface of the thorax, of which those that are audible are called **heart sounds**. As an example, one can mention the most known pair of heart sounds: the heartbeats **S1** and **S2**, frequently described as "lub" and "dub", respectively. As a mechanical result of the dynamics between fluids, muscles, valves, blood vessels and other tissues, the heart sounds hold important information about blood circulation and the health of heart structures. Proof of this is the use, until nowadays, of the **stethoscope**, a centennial instrument dedicated to the amplification, filtering and projection of sound waves to the human ear (PAZIN-FILHO; SCHMIDT; MACIEL, 2004; TILKIAN, 2004; FEITOSA, 2008; LENG et al., 2015; REECE et al., 2018). **Auscultation** is the name given to the evaluation of sounds produced spontaneously by the organs (FEITOSA, 2008). It is used in the decision-making process of application of more expensive technologies, in guidelines for the screening of athletes and surgical patients in preoperative state (KUMAR; THOMPSON, 2012), and in clinical practice it becomes an important criterion in determining the need to consult a specialist (NASSRALLA; ZEIN; HAJJ, 2017). According to Feitosa (2008), when the anamnesis and the physical examination are well executed, 90% of the cases are correctly diagnosed. Timely detection of many forms of congenital and acquired diseases, as Kumar and Thompson (2012) state, can be made through a proficient auscultation. Thus, the cardiac auscultation must be seen as an auxiliary method to the characterization of clinical syndromes, in addition to other cardiovascular physical exams and the patient's clinical history (PAZIN-FILHO; SCHMIDT; MACIEL, 2004).

In human medicine, heart diseases are the biggest cause of deaths, corresponding to 17.5 million deaths in 2012 (NARVÁEZ et al., 2017). An even more alarming statistic is the one that shows that 75% of these deaths are located in underdeveloped countries, in whose rural areas the limited access to prevention programs, medical assistance, equipment and specialists prevails (LENG et al., 2015; NARVÁEZ et al., 2017). Therefore, it is not surprising the abundance of scientific work aiming at promoting improvements in the stethoscope (such as electronic stethoscope), in aspects of the auscultation (filtering, telemedicine etc.) and in the processing of heart sounds (automatic event detection and segmentation, among others). Most of this research comes from underdeveloped countries.

In the case of veterinary medicine, the scenario is worse. Even when specialists and instruments are available, few tutors are willing to pay for additional exams (ANDRADE, 2018). All these factors highlight the need to increase the quality of the first contact between the physician and the patient. The present work represents an additional effort to increase tooling

for the practice of veterinary medicine.

1.1 PROBLEM

One class of particularly revealing heart sounds is the one of **heart murmurs** — noises often associated to the degeneration of heart valves, to the reflux of blood within the heart and to heart failure. Due to the importance of the detection and analysis of the heart murmurs, with the advance of medical protocols, measurement scales were developed for these events. One of these scales is quoted by various authors (FEITOSA, 2008; TILKIAN, 2004; PAZIN-FILHO; SCHMIDT; MACIEL, 2004) and consists of a classification from I to VI according to the criteria described below:

- **Degree I**: heart murmur of such a low intensity that requires a long period of auscultation and high concentration.
- **Degree II**: small-intensity heart murmur that can be immediately heard by an experienced professional.
- **Degree III**: striking heart murmur but not too intense (FEITOSA, 2008).
- **Degree IV**: intense heart murmur. At this stage a palpable vibration can be present (TILKIAN, 2004), but according to Andrade (2018) it can propagate to other areas of auscultation.
- **Degree V**: very intense heart murmur. Comes with palpable vibration, but it is not audible when detaching the stethoscope from the patient's chest (ANDRADE, 2018).
- **Degree VI**: severe murmur with palpable vibration, and can be auscultated even without any contact between the stethoscope and the thorax (TILKIAN, 2004).

As can be seen, the auscultation (and, in some aspects, even the scale) is based on subjective factors. The interpretation of heart sounds itself is largely dependent on details such as experience, ability and auditory skills of the physician (PAZIN-FILHO; SCHMIDT; MACIEL, 2004; LENG et al., 2015). As Leng et al. (2015) affirm, the difficulty in the analysis of a small-intensity sound with high and low competing frequencies was, for a long time, diminished only by the use of the acoustic stethoscope (sometimes also called traditional stethoscope). Nevertheless, the author recalls the presence of noises in the signal — often caused by the friction of the equipment over the chest or by environmental factors in the consultation room, such as voices, vehicles and other equipment.

With the technological progress, other exams were developed; X-ray, electrocardiography and echocardiography contribute, each in its own way, to the confirmation of the diagnosis of cardiovascular diseases (FEITOSA, 2008). While they represent a significant advance in diagnostic capacity, imaging equipment (which enables the aforementioned examinations) have brought with them an unwanted consequence: the devaluation of the fundamental diagnostic technique (PAZIN-FILHO; SCHMIDT; MACIEL, 2004). With the advent of these new instruments, which have never extinguished the importance of auscultation, the new process of teaching medical professionals has imposed on low- and middle-income countries enormous

economic costs due to the excessive use of sophisticated diagnostic methods (PAZIN-FILHO; SCHMIDT; MACIEL, 2004). According to Feitosa (2008), about 40% to 50% of subsidiaries exams result in normal findings, which (still according to the author) often characterizes clinical error. All of these complementary exams require specific machinery (often expensive and/or with restrictions of use) that also gives rise to the need for new consultations and extra costs. Add to this the aforementioned lack of equipment in the underdeveloped countries and, in veterinary medicine, the refusal of many tutors to bear the extra costs.

The classic limitations of auscultation, the difficulty of professional training in the technique and the problems for enabling additional tests bring about, finally, the main problem to be faced in this work: the deficiency in case monitoring due to the subjectivity of the auscultation. The impossibility of regular application of imaging tests, despite the objectiveness of some of their metrics, prevents the monitoring of progress in clinical cases. The physical examination (of which auscultation is part), on the other hand, is consistently applied during clinical consultation. Such examination is considered by Feitosa (2008) as "the most useful and economical tool the physician possesses".

1.2 GENERAL OBJECTIVE

Automatically estimate the degree of canine heart murmur on a scale of 0[1] to 6 from a heart sound record marked with the time of heartbeats S1 and S2.

1.3 SPECIFIC OBJECTIVES

As specific objectives, it is listed:

- Develop a set of algorithms that generate features — numerical values that synthesize characteristics of the records.
- Apply such algorithms to a bank of heart sound records.
- Produce a table aggregating the values of the features for the set of records.
- Automatically estimate the degree of canine heart murmur from the table of features along with machine learning techniques.

1.4 JUSTIFICATION

The automatic classification of the degree of murmur introduces objectivity to the clinical analysis when it confronts the perception of the health professional. Because it consists of an immutable metric over time, it allows a patient to be evaluated throughout his or her history with reliability. The proposed methodology is based on widely reproducible artifacts

[1] The original scale, already quoted, assumes the classification from I to VI, with I being the mildest murmur and VI the most pronounced one. Here, the metric was extrapolated to start at 0, meaning no murmur; the other levels correspond to those already present in the original scale.

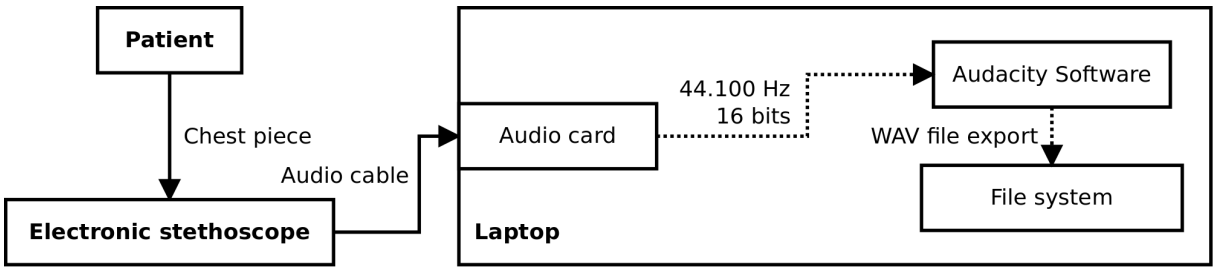
(commercial quality electronic stethoscope and software), thus enabling the comparison of different cases and even the continuity of evaluation of the same case in different clinics.

All these characteristics provide a better basis for the diagnosis and lessen damages resulting from the non-application of imaging tests. In addition, the use of the electronic stethoscope is a non-invasive, inexpensive, quick and easy procedure operated by the clinician himself, which brings with it all the benefits of digital recording of heart sounds — audio reproducibility, sharing of records and signal visualization as a graphic. It can also provide greater assistance in the training of new professionals. Studies have suggested that the electronic stethoscope has the potential to benefit substantially the learning of basic auscultation techniques (KUMAR; THOMPSON, 2012; MESQUITA et al., 2013). A new application of the stethoscope — in this case, the automatic classification of the degree of murmur — should only contribute to this scenario.

1.5 OVERVIEW

The work is based on the use of a common (commercial-line) electronic stethoscope connected to the analog audio input of a laptop. The audio signal is scanned at 44,100 samples per second, each with 2 bytes (or 16 bits). At this time, the signal is transmitted as stereo, although in practice its two channels are identical. In any case, stereo audio is recorded with the help of the open-source software Audacity[2]. This entire acquisition process is represented in Figure 1.

Figure 1 – Acquisition of heart sounds



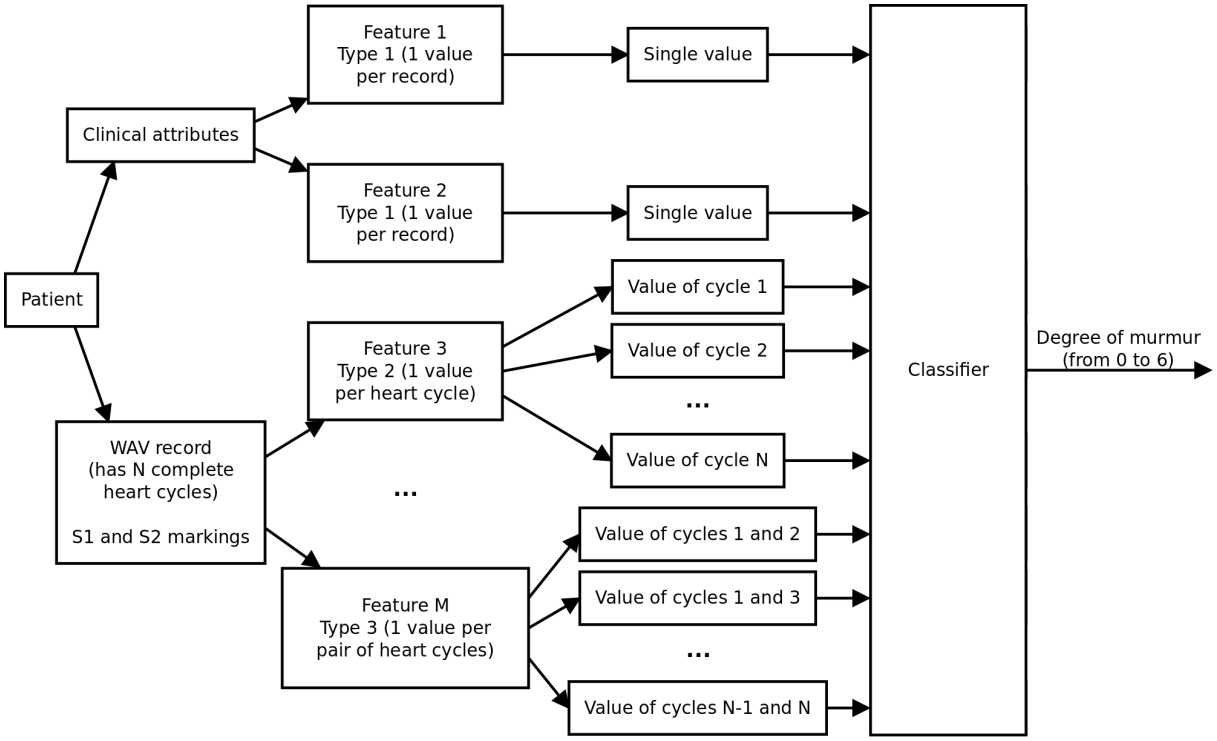
Source: Own authorship

With the WAV (Waveform Audio File Format) records of the heart sounds of the patients, their objective clinical data and the times of occurrence of S1s and S2s, features are produced to be used as input to a classification algorithm, which returns the estimate in the desired range (0 to 6) as an output.

Each feature can generate a single value per patient (**Type 1**), or a quantity of values equal to the number of heart cycles in the WAV record (**Type 2**), or even a value for each distinct pair (combination) of heart cycles in the record (**Type 3**). The mechanism is summarized in Figure 2. Each of its steps will be timely detailed in this work.

[2]Cross-platform open-source software for audio recording and editing (Audacity Team, 2019).

Figure 2 – Generation of features and automatic classification



Source: Own authorship

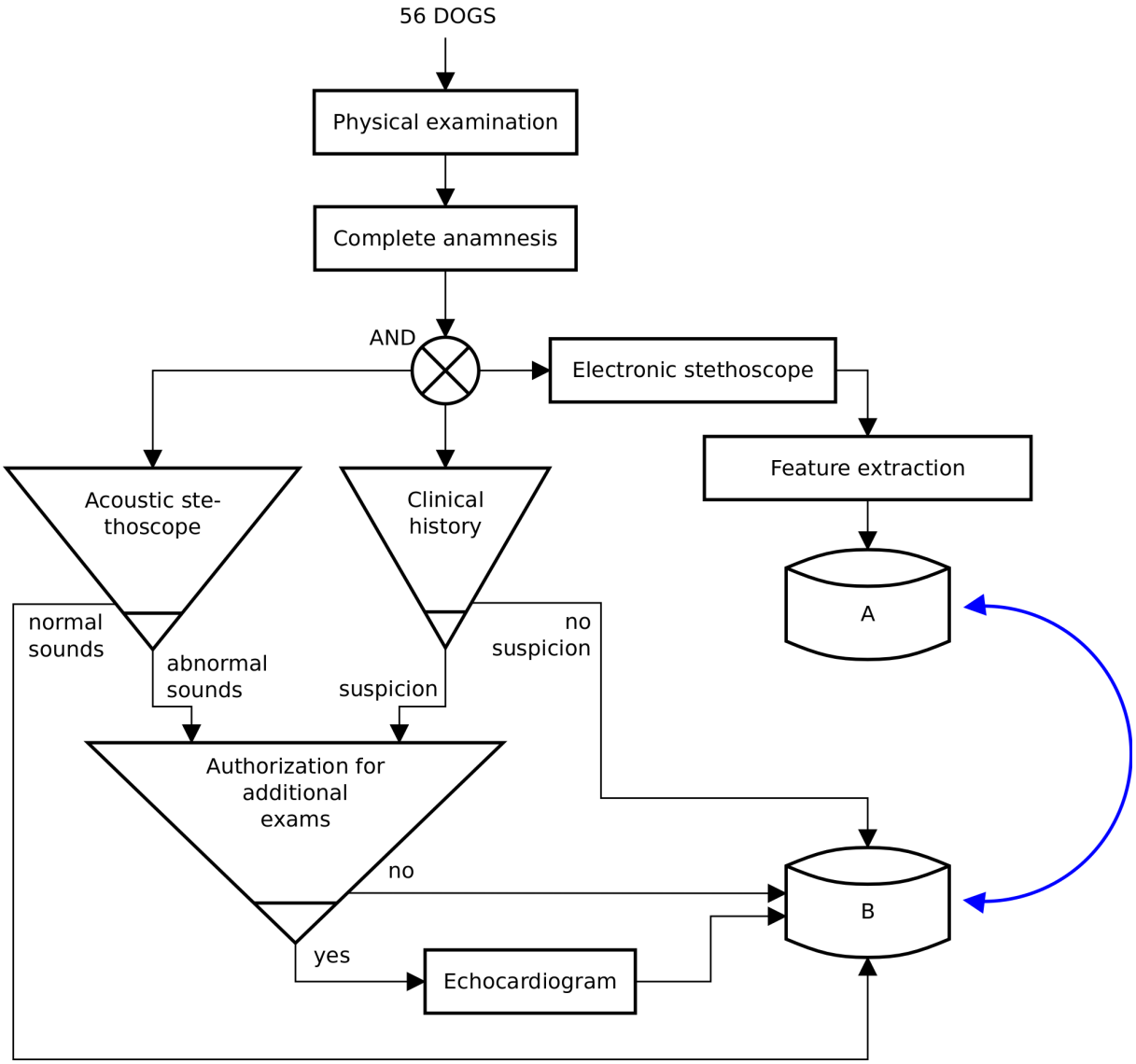
1.6 METHODOLOGY

The audio files used in this research were recorded by Andrade (2018) in a set of 56 dogs, from September 2017, at the Dog Vet Veterinary Clinic (Curitiba — PR). There was no distinction of race, gender, age or health condition. The only exclusion criteria adopted were the size of the animal and the choice of its tutor (large animals or those whose tutors chose not to participate in the research were not considered). It is emphasized that this data collection was non-invasive, did not cause suffering to animals (not even trimming of the dog's fur) and was previously authorized by the ethics committee *Comissão de Ética no Uso de Animais* of the *Universidade Tecnológica Federal do Paraná* (CEUA-UTFPR), as shown in Anexo A. The stethoscope used was the Jabes Digital Stethoscope Analyzer. The product has been set to "W" (wide range) mode, which captures sounds over the entire range from 20 Hz to 1,000 Hz. The volume for auscultation has been adjusted according to the need, patient to patient. The stethoscope was positioned at the point of greatest intensity (apex beat) of each patient.

Simultaneously with the auscultation, the stethoscope was connected to a laptop with the same audio cable that accompanies that device. This allowed the recording on file of the same sound perceived by the veterinarian. The software used, Audacity, is an open source software and has been configured for stereo digitalization at 44,100 Hz. For each patient, at least 10 complete heart cycles of good quality were recorded. Through an auscultation with acoustic stethoscope, which the veterinarian was familiar with, the degree of heart murmur (if

any) from I to VI was registered. At the same occasion, a careful anamnesis was performed and dogs that needed complementary examinations were referred. Other parameters have been cataloged, such as the name of the tutor, and the name, identification code, breed, age and weight of the patient. In the present study, these data and the degree of heart murmurs were named **database B** — a registry of clinical data of the study patients, all coming directly from the physical examination and some strongly dependent on the experience of the medical professional. Figure 3 shows the procedure performed once for each of the 56 patients.

Figure 3 – Clinical consultation and generation of databases A and B



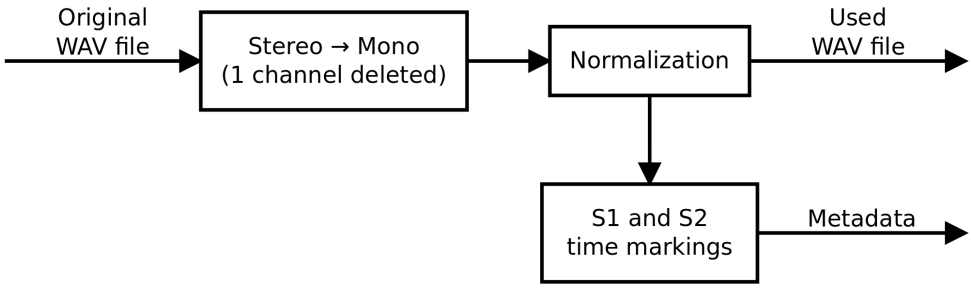
Source: Adapted from Andrade (2018)

Another aspect shown in Figure 3 is the filling of **database A**. Such a base consists of the features mentioned in Section 1.3 and in Figure 2. However, they are not generated directly from the files recorded during the examination. In fact, the files used in the present work are the post-processed ones with `".wav"` extensions as described in Andrade (2018) and summarized

in Figure 4. In this figure, it is also emphasized that the medical professional's experience has generated extremely important artifacts — the so-called "metadata", which consist of the time markings of the heart sounds S1 and S2 of each heart cycle. Each patient's metadata is saved in a `".csv"` extension file. The set of the metadata files of the post-processed records (**input audio files**) of the patients of this study form the **input data** of the present study.

It is from the newly defined input data that the features are extracted (as shown in Figure 5). The Chapter 3 describes them in more detail. With its values finally stored in database A, the next step was to analyze machine learning algorithms capable of estimating the degree of a patient's heart murmur based only on the data generated by the features.

Figure 4 – Post-processing and generation of metadata



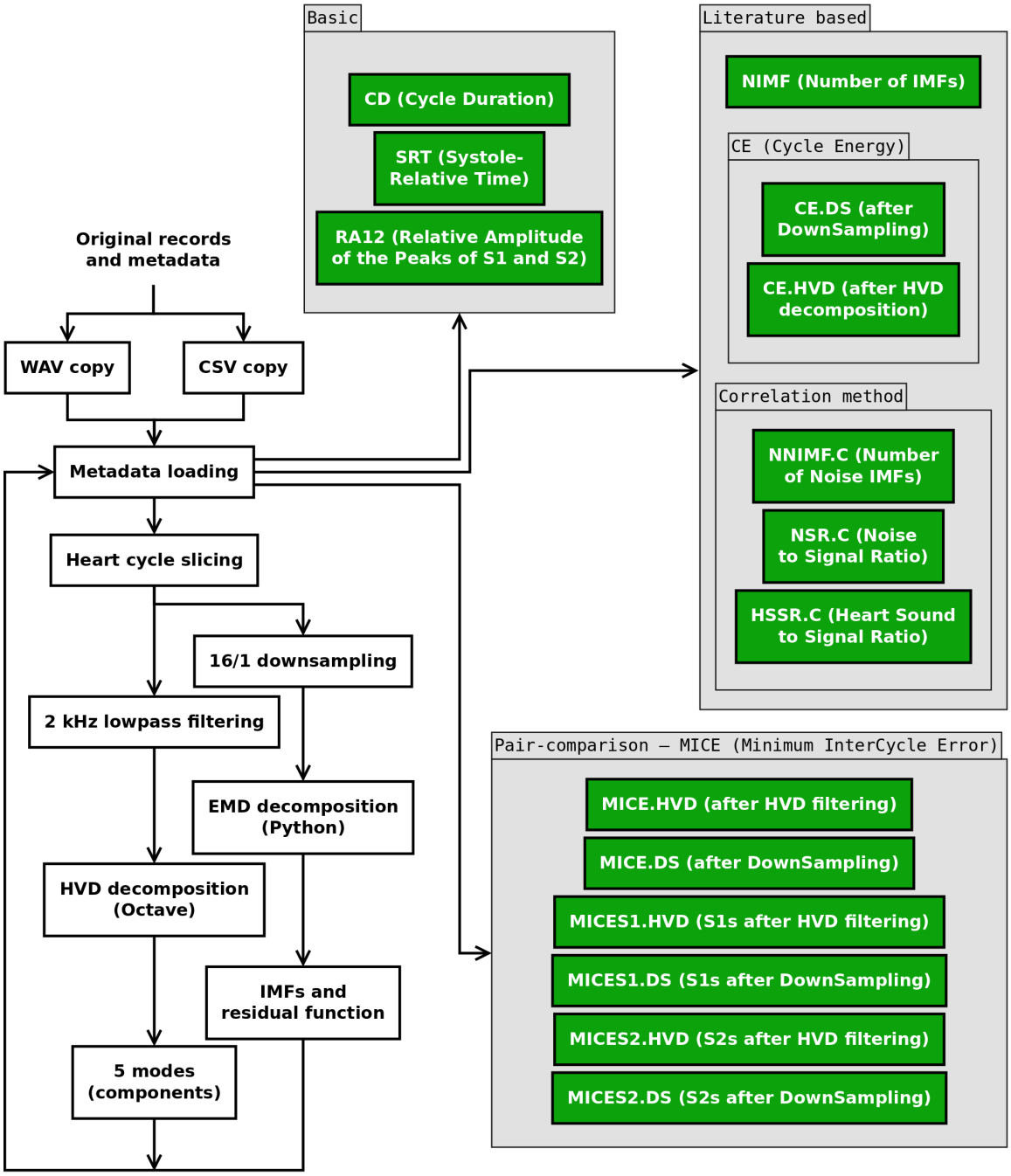
Source: Own authorship

1.7 PRESENTATION OF THE DOCUMENT

The document is organized as follows:

- Chapter 2 summarizes the knowledge needed to fully understand this research. Topics covered include the nature of heart sounds, the limits of human hearing, sound signal digitalization, and basic and elaborated methods of digital signal processing.
- Chapter 3 describes the procedures for processing the set of available records and metadata. In particular, the formal definition of the produced features is given in sections 3.2 to 3.4. The performance of such features to estimate the degree of patient's heart murmur is discussed in Section 3.5.
- The synthesis of the results achieved with this research is presented in Chapter 4.

Figure 5 – Overview of feature extraction



Source: Own authorship

2 LITERATURE REVIEW

The research presented in this document is of multidisciplinary nature, since it involves a range of subjects reaching from veterinary medicine (topics of anatomy, physiology and clinical practice), digital signal processing, to finally culminating in classifiers and machine learning. Therefore, many new knowledges were necessary to make the work feasible, and they are the key to allow full understanding of it. In the following sections, these knowledges are shared.

2.1 TOPICS OF HEART SOUND CAPTATION

In the literature, there is controversy regarding the frequency range of heart sounds. The minimum and maximum registers found are, respectively, 2 Hz (WANG et al., 2007) for low-pitched sounds and 2 kHz (WEBSTER, 2009) for high-pitched sounds. Concerning the human ear, however, there is a certain consensus that the range of perception is limited between 20 Hz and 20 kHz — but even for individuals with perfect hearing health, not all of this band has the same ease of perception (PAZIN-FILHO; SCHMIDT; MACIEL, 2004). The main range for auscultation of heart sounds, from 20 Hz to 500 Hz, has one of the highest perception thresholds (PAZIN-FILHO; SCHMIDT; MACIEL, 2004; KUMAR; THOMPSON, 2012), i.e., one of the regions of the audible spectrum that are more difficult for the listener to assimilate.

The acoustic stethoscope, traditionally utilized by physicians, uses purely mechanical phenomena to direct the sound to the ears of the specialist. The electronic stethoscope, on the other hand, makes use of electromechanical components specialized in converting the **mechanical signal** of the heart (heart sounds) into an **electrical signal** (LENG et al., 2015), both digital and analog. The stethoscope that was used in the recording of the samples for the present work, for example, has as output an analog electric signal, and has the ability to remove the frequency filters — that is, to capture the entire frequency range of heart sounds, without distinction of sensitivity (at least in theory).

In this type of analog output, the task of **digitizing** the captured audio is delegated to the computer. The digital audio can be heard, stored, transmitted, transformed to graph or processed on the computer. The digitization consists of a sampling of the electrical magnitude of the signal (usually of its electrical voltage), in a regular time interval, by a device known by the acronym ADC: Analog-to-Digital Converter (LENG et al., 2015). The regular time interval determines a frequency called **sampling frequency**: the number of samples measured per second from the analog signal (HAYKIN; VEEN, 2001). Thus, a sample rate of 44.1 kHz, for example, comes down to 44,100 values per second, and is known as "CD quality" (as it is the same on music CDs). Sampling is an operation that generates a discrete time signal from a continuous time signal (HAYKIN; VEEN, 2001).

The digitization also provides for the digital storage of the samples — that is, a coding of each input in the form of bits. This is because digital storage devices, such as the computer's hard disk, do not have the ability to write values to the decimal system — only in the binary system —, and also have finite write capability — meaning, for example, that values have to be discretized after their sampling. The amount of bits used in the digitization defines, in conjunction with the sampling frequency, the quality of the digitization (fidelity of digital audio with respect to its analog counterpart). Both impact proportionally on the transmission rate, power consumption and processing time (LENG et al., 2015). In summary, higher quality audio has more samples per second and more bits per sample, so it consumes more resources during recording and processing; in this way, the ideal bit and sampling rates are defined by the designer for the specific application in development.

At this point, the extremely important concept of the **aliasing effect** is presented. Aliasing consists of a distortion in the frequency spectrum of the discrete time signal in comparison to the spectrum of the continuous time signal, and can be avoided by choosing a sampling frequency at least two times greater than the highest frequency of the analog signal (HAYKIN; VEEN, 2001). In practice, however, the most common is to apply a high-frequency removal filter (the so-called **"low-pass filter"**) to this signal before sampling it (LENG et al., 2015). Filtering is a potentially destructive procedure over the desired data; again, it is up to the designer to study the frequencies of interest and eliminate unwanted effects on the recording. In the specific case of the recordings used in the present work, with a sample rate equal to 44.1 kHz, a maximum frequency of 22,050 Hz was captured without distortions, a frequency that gives a wide margin in relation to the limits of heart sounds.

The effects of aliasing are the apparent occurrence in the sampled signal of frequencies non-existent in the continuous time signal, and the non-equivalence (for analysis purposes) between both signals (HAYKIN; VEEN, 2001; WEBSTER, 2009). At least during the process of digitization directly on a computer, such as the audios used in the research, all necessary care for this phenomenon is automatically taken by the audio card in response to the sampling frequency chosen for recording; however, the concern with aliasing comes up again at performing a **downsampling** on the already digitized signal.

2.2 BASIC DIGITAL SIGNAL PROCESSING OPERATIONS

As already mentioned, the sampling frequency impacts the processing time. Therefore, prior to computationally costly procedures, it is very common to perform an operation known as downsampling, whose concept can be understood, in a simplified way, as a new sampling, now on the discretized signal (HAYKIN; VEEN, 2001), in order to reduce the number of samples to be processed. A key concept here is the **downsampling rate** — the factor that indicates how many samples of the input vector corresponds to a certain amount of samples in the output vector. For example, the 16/1 ("16 to 1", or simply "16") downsampling rate, used in this study, indicates that every 16 originally recorded samples match to only 1 sample in the

downsampled audio.

It is worth noting, however, that it is not always just a simple re-sampling ("choosing" some samples and "ignoring" the others). The S factor downsampling on a discrete sampling frequency signal f_S [Hz] is an operation that simulates a sampling of f_{DS} [Hz] on the original continuous signal, according to Equation (1). Thus, the discussion of aliasing in Section 2.1 remains valid: f_{DS} should be at least twice as large as the largest frequency of the original signal. However, it is very common that (as with the audios in this study) the continuous signal is no longer accessible — which is why a downsampling is performed, firstly, because in the presence of the original signal a new simple sampling at f_{DS} [Hz] would suffice. When only the discrete signal is accessible, the solution is to apply a low-pass digital filter over it prior to the downsampling, thus eliminating the frequencies that would cause the aliasing effect on the downsampled vector.

$$f_{DS} = \frac{f_S}{S} \tag{1}$$

In addition to downsampling, another operation that was widely used in the work is the calculation of the **energy of a discrete signal** $x[n]$. This energy is defined by Equation (2) (WICKERT, 2013).

$$E_x = \lim_{N\to\infty} \sum_{n=-N}^{N} |x[n]|^2 \tag{2}$$

If $x[n]$ is a vector of L real values, as is the case of the samples used in the present work, the equation can still be simplified to the form of Equation (3).

$$E_x = \sum_{n=0}^{L-1} x[n]^2 \tag{3}$$

2.3 EMPIRICAL MODE DECOMPOSITION

Empirical Mode Decomposition (EMD) is a decomposition that can be seen as an expansion of a data vector into a collection of other vectors called Intrinsic Mode Functions, or **IMFs** (HUANG et al., 1998), in the time domain — that is, it generates a series of signals (the IMFs) from the original signal, all in the time domain. According to its authors, this is a non-linear and non-stationary analysis method.

Haykin and Veen (2001) affirm that a signal is said to be "non-stationary" if its intrinsic characteristics vary over time. As for linearity, the same authors point out that a nonlinear system is one that violates the **principle of superposition** — on which they write:

> It is said that a system is *linear* if it satisfies the *principle of superposition*. That is, the response of a linear system to a weighted sum of input signals is equal to the same weighted sum of output signals, each output signal being associated to a particular input signal acting on the system independently of all other input signals (HAYKIN; VEEN, 2001, p. 69).

Heart sounds are considered non-stationary because of changes in their characteristics over time (VIKHE; HAMDE; NEHE, 2009). Many phenomena of nature are prone to non-linearity, and the production of heart sounds is no exception. According to Leng et al. (2015) and Barma et al. (2014), these sounds constitute a class of strictly non-linear biological signals. Moreover, as Huang et al. (1998) remember, even a perfectly linear phenomenon is often perceived as non-linear due to imperfections in the measuring instruments used to assess it.

For all of this, the EMD proved to be an adequate tool for computational analysis of the audio signals of the research. The effect of EMD decomposition on the heart sound record of a healthy and young patient can be seen in Figure 6. In this example, the heart cycle (represented in red) was decomposed, generating 8 IMFs (indices 0 to 7) plus one residual function, all shown in green. IMFs of indices 3, 4, 5 and 6 were hidden for space saving but, following the trend already indicated by the visible modes, they have gradually smaller amplitude and frequency. The sum of all these products (IMFs and residual function) results in the signal represented in blue, which corresponds exactly to the original signal.

2.4 HILBERT VIBRATION DECOMPOSITION

The Hilbert Vibration Decomposition (HVD) is another decomposition dedicated to the problem of non-stationary broadband vibration analysis (FELDMAN, 2006). According to the author, unlike methods such as EMD, only the Hilbert Transform (HT) and other simple mathematical tools are used, without involving advanced techniques of digital signal processing.

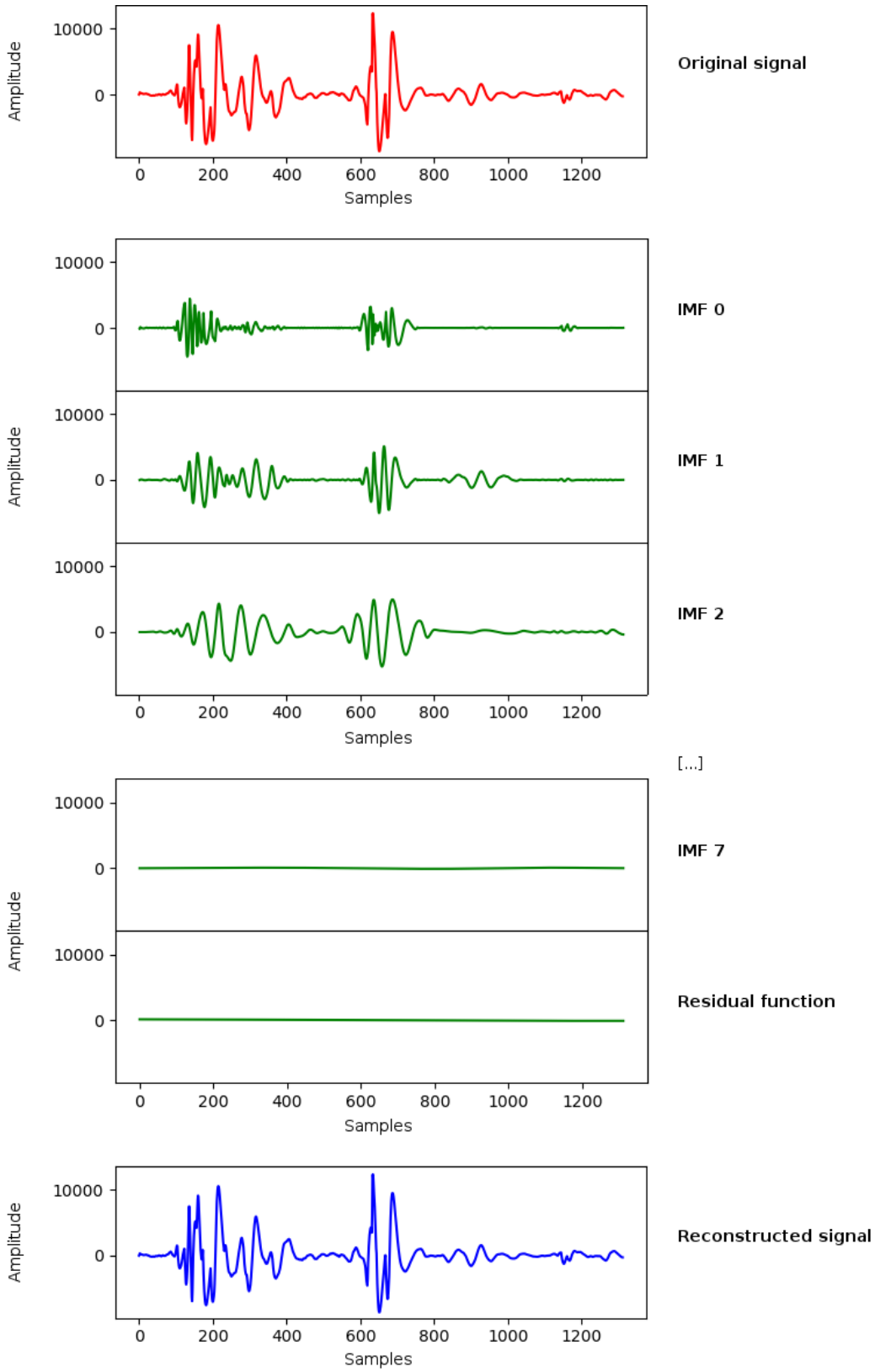
Feldman (2006) states that components extracted via EMD or HVD (called "intrinsic oscillatory modes") consist of **monocomponent signals** — those with only one maximum or minimum value between each zero crossing. As examples, slow frequency-modulated and narrow-band vibration signals are mentioned. The IMFs shown in Figure 6, being intrinsic modes, illustrate this behavior.

Figure 7 shows the HVD decomposition for the same patient. As noted, the first monocomponent isolated from the original signal contains the largest amplitude. The energy of the extracted modes drops gradually with each new mode. The residual function was not illustrated in this example because it has even lower energy than the 5th extracted mode (Mode 4), but in the same way as with the EMD, the sum of the components and the residual function reconstructs the original signal.

2.5 TOPICS OF MACHINE LEARNING

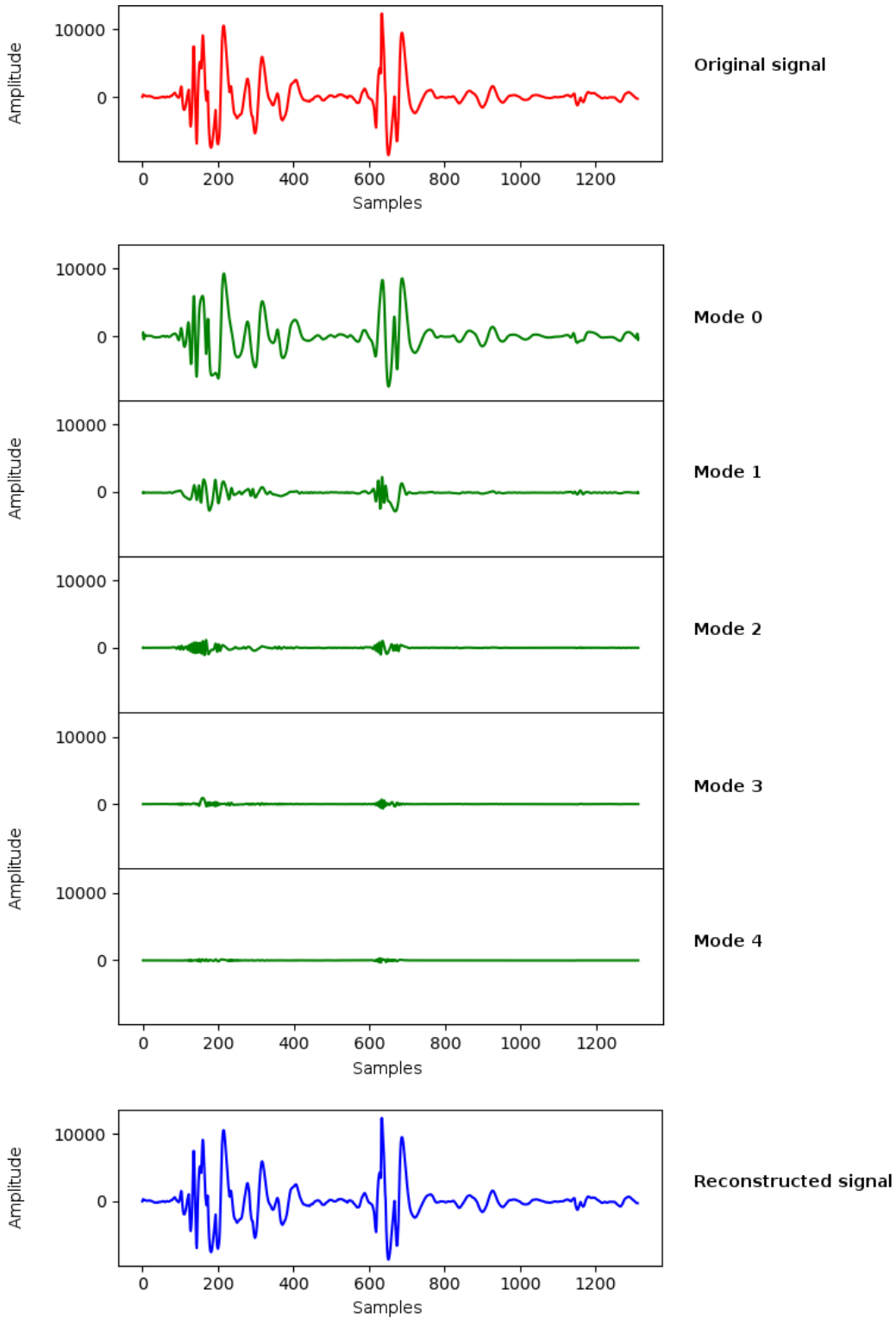
Machine learning is the area of knowledge that is concerned with the application of generic algorithms and techniques to the process of obtaining knowledge from data (KALUŽA, 2016). It includes the processing of available data to construct **models** capable of predicting new data (GOLLAPUDI; LAXMIKANTH, 2016). Models are usually representations of data,

Figure 6 – EMD decomposition of the audio record of a heart cycle



Source: Own authorship

Figure 7 – HVD decomposition of the audio record of a heart cycle



Source: Own authorship

and are a product of the application of a machine learning algorithm to the available dataset (GOLLAPUDI; LAXMIKANTH, 2016).

In the scope of machine learning, the **dataset** corresponds to a table in a database

or in a spreadsheet, because it consists of lines (**instances**) made of a group of columns (**attributes**) (BOUCKAERT et al., 2018). Each instance represents a different record — for instance, data from the acquisition of a certain product, or samplings from the climate conditions on a certain spot. Each attribute, on the other hand, represents a known variable (feature) of the instance (the price of the product, the date and time, the client's identification number, the temperature, the air humidity, and so on). "Variable", "field" or "feature" are all synonyms of attribute (GOLLAPUDI; LAXMIKANTH, 2016).

A class of problems commonly addressed in machine learning is the one called **supervised learning**, that can be summarized as the obtaining of a function f capable of mapping elements of a set X in corresponding elements of a set Y (KALUŽA, 2016) — in other words, $f : X \to Y$. In this kind of problem, the columns of the dataset are divided by the technician in two groups: the **input attributes** and the **output attributes**. Given this separation, the values of the instances of the dataset are interpreted by the machine learning algorithm as examples of the sets X and Y, respectively, and its role is to speculate a function f — that is, a model that maps combinations of input values to output values (GOLLAPUDI; LAXMIKANTH, 2016). The process of generation of the model from the instances is known as **training**.

A subclass of the supervised learning problems is the one of **regression**, in which Y consists of an attribute of continuous value. This way, according to Kaluža (2016), regression is understood as a process that allows one to comprehend how the values of a group of features (input attributes) affect a continuous target variable. The problem studied in the present work is an example of this subclass: it is desired to determine the relationship between inputs (like weight and features extracted from the heart sound records of small dogs) and a continuous output (the degree of heart murmur).

A regression model allows outputs to be estimated for arbitrary inputs, i.e., inputs that extrapolate the known dataset. This way, resuming the previous example, once determined a functional regression model for the degree of murmur, one can (at least in theory) input it with data from a dog which was never studied before and still obtain an estimate of its degree of heart murmur. However, models generated by machine learning algorithms produce errors in their estimates. In fact, an error can be observed even by feeding the model with the instances — the so called **training set** — that were used to create it, in the first place. Error can be measured in several ways, depending on the class of algorithms used. In the specific case of the present work, two metrics that were compatible with the regression problem were selected.

The **correlation coefficient** represents a measure of similarity between the estimates produced by the model and the expected outputs (known values) for the same instances. Its calculation is made possible by Equation (4) (KALUŽA, 2016), where X and Y represent the expected outputs and the estimates, respectively. The bar notation denotes the arithmetic mean of the variable for the instances, whose index (i) varies from 1 to the number of known

cases (n).

$$CC_{XY} = \frac{\sum_{i=1}^{n}(X_i - \overline{X})(Y_i - \overline{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \overline{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \overline{Y})^2}} \tag{4}$$

As it follows from the equation, if all the estimates (Y_i) are perfectly proportional to the expected outputs (X_i), the correlation coefficient will be equal to 1 (highest possible value), indicating a perfect and direct linear relationship between these variables. A correlation coefficient of -1 (lowest possible value) would also indicate a perfect linear relationship between them, but at an inverse proportion (one grows to the exact ratio at which the other reduces). Thus, it is said that a value close to the extremes signals a quasi-linear relationship between the variables, and a value close to 0 denotes absence of relation between them. Because of this, a correlation coefficient closer to 1 is almost always sought.

The second error metric for regression models that is important for the present work is the mean absolute error (**MAE**). It is defined as the mean of the absolute difference between the estimates and the expected outputs (GOLLAPUDI; LAXMIKANTH, 2016; KALUŽA, 2016). Its value is given by Equation (5).

$$MAE(X, Y) = \frac{1}{n}\sum_{i=1}^{n}|X_i - Y_i| \tag{5}$$

While many other metrics may be used, these have interpretation per se, even if there is no information on the statistical distribution of the data, although the existence of such information would allow further analysis. In addition, they are not as sensitive to outliers, that is, instances with discrepant data (GOLLAPUDI; LAXMIKANTH, 2016; KALUŽA, 2016).

The measurement of the estimation error plays an important role in the analysis of another crucial aspect of the model: its **generalization** capacity. Generally speaking, generalization consists of the ability of a model to maintain a low estimation error even when applied to inputs that do not exist in the training set. In this situation, the error is given a special name: **generalization error**. Because it depends on future data (i.e., unavailable during model training), the actual generalization error can not be determined, only an estimate of it (KALUŽA, 2016).

A common technique is to not use all instances known as a training set; a small part of them is separated into what is called the **test set**. According to Kaluža (2016), when calculating the estimation error for the test set, an estimate for the generalization error is obtained. According to the author, a common proportion for the number of instances in the training and test sets respectively is 70:30, however there is no closed formula for defining it. This approach has two problems. First, the number of instances remaining in the training set may be so low that the quality of the model produced can be affected. Second, it is necessary to use mechanisms that ensure that the test set is representative of the training set, that is, that it bears similarity with the training set (KALUŽA, 2016).

To mitigate these problems, the approach employed in this work is that of **cross-validation**. In it, the instances are divided not in two parts but in an arbitrary number k of sets of approximately the same size. The major difference of cross-validation, however, lies in the fact that k models are produced — each time using only one of the different sets as a test set, and the rest of the instances as the training set (KALUŽA, 2016). Thus, k estimate errors are calculated, and their arithmetic mean gives the estimate of the generalization error. Unlike the previous method, each instance is used both in the test step (once) and in the training step ($k - 1$ times).

3 DEVELOPMENT

Section 1.6 presented the input data of the work — post-processed WAV records and markings of the S1 and S2 — and introduced the concept of **feature type** (1, 2 or 3), which basically indicates how many values the feature produces. Chapter 2 has systematized several useful knowledges to understand the work developed, especially for the calculation of features, which will be enumerated and will have their algorithms detailed in this chapter.

However, before describing the features, it is necessary to present the software framework that allowed them to be generated; in particular, it is important to understand how the data contained in the input files is translated into variables for calculation.

3.1 PROCESSING SCRIPT

As shown in Figure 2, feature generation is a complex process executed once for each pair of input audio and metadata file.

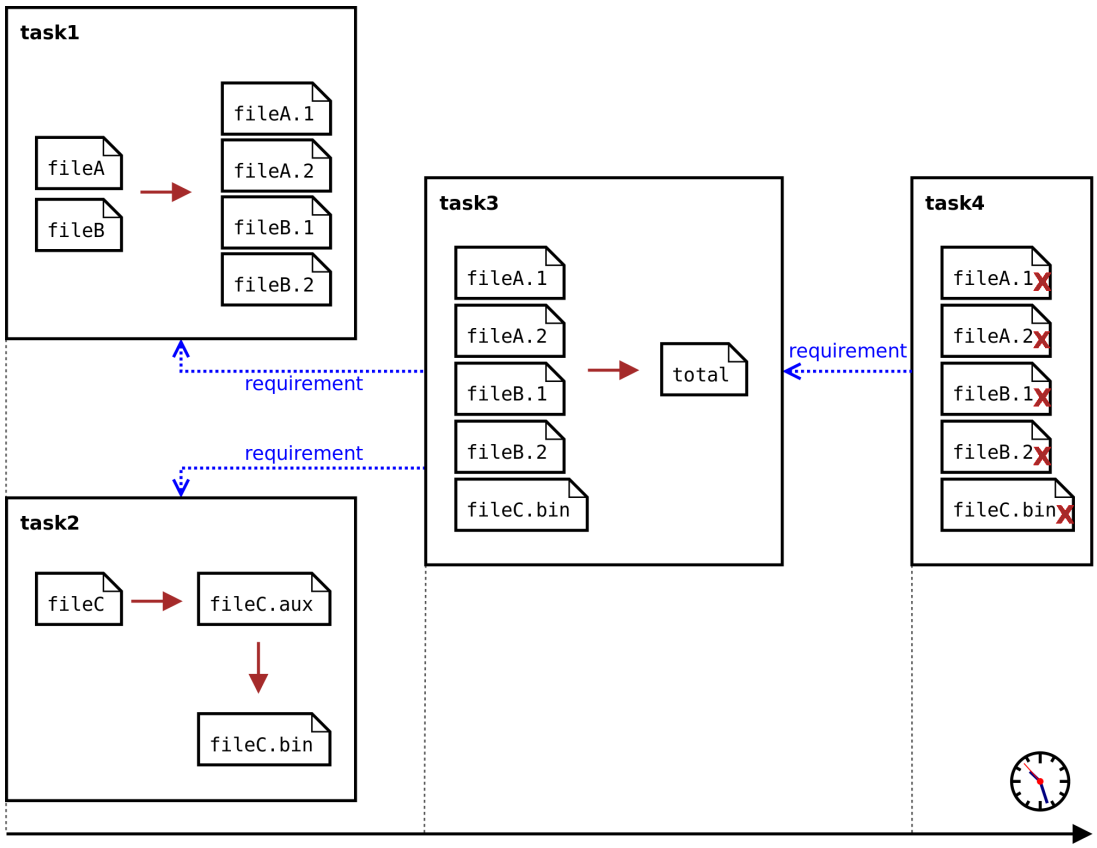
To facilitate the development of these features, **gulp** was used, which is defined as a set of tools to automate painful or time-consuming tasks in the workflow (GULP, 2019b). It is an open source software (GULP, 2019a) with the ability to integrate with many other popular technologies today, such as NodeJS, Python and the command line interface of the operating system.

The main feature of gulp is the file `"gulpfile.js"`, the **entry point** of the automation script. Usually, the **tasks** to be performed are defined directly in this file. Each task consists of a stream of one or more **content transformations** over a group of input files. A content transformation can be as simple as reading or removing certain data from the file, or as complex as compressing an image, just to mention a few. Each task handles a generally different set of input files, and usually outputs another set of files. Tasks may have other tasks as **requirements for execution**.

The concepts of tasks, requirements and transformations are illustrated in the example in Figure 8. In it, `"task1"` has a transformation that takes each input file and generates two output files. `"task2"` has two chained transformations. `"task3"`, which depends on the products of the previous tasks, transforms a set of input files into a single output file. Finally, `"task4"` waits for completion of `"task3"` and then clear temporary files. The timeline indicates the execution sequence of the tasks; it is interesting to note that since `"task1"` and `"task2"` have no dependence on each other, both are executed in parallel.

The transformations are declared in their own source code files, often (but not always) distributed in the form of NPM (NodeJS Package Manager) modules — a package manager specialized in JavaScript libraries entitled the largest software registry in the world (NPM, Inc., 2019). The versions of NodeJS and NPM used in this work are 6.9.5 and 4.2.0, respectively.

Figure 8 – Example of execution of gulp tasks with requirements



Source: Own authorship

In addition to those already mentioned, gulp characteristics that are especially useful for the present work are:

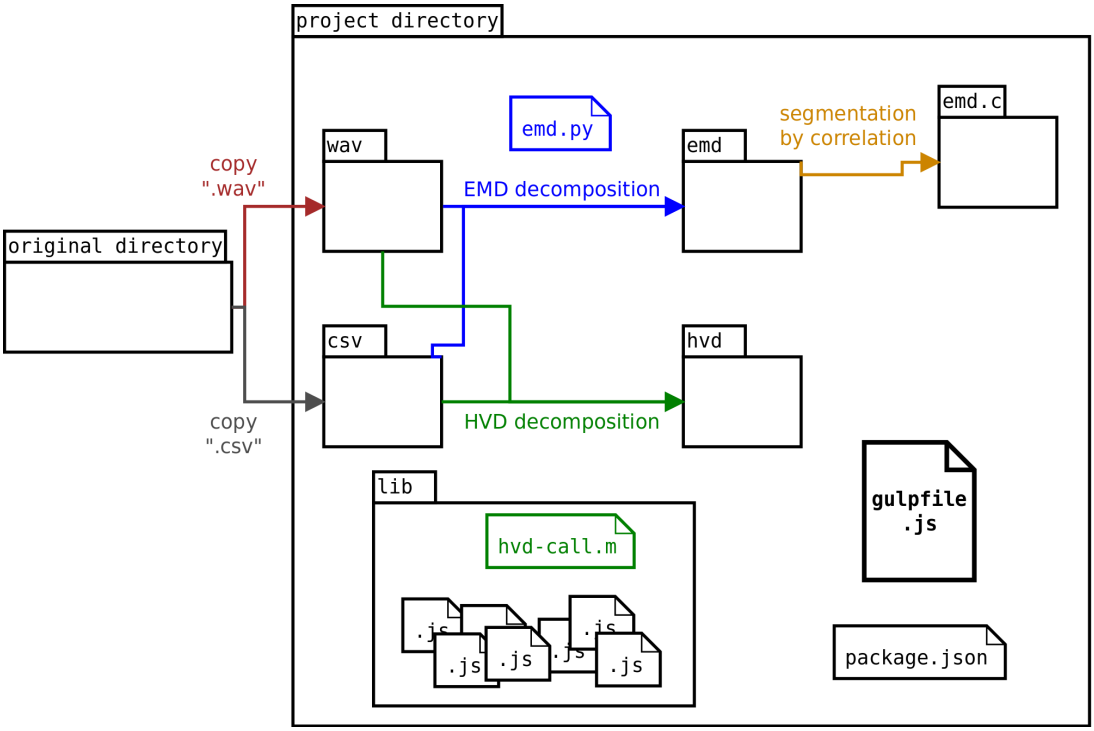
- Native functions for manipulating files, such as in-memory loading and copying to another directory.
- Possibility of creating custom transformations by the user (for example, those described in Subsection 3.1.1, which operate on the metadata).
- **gulp-text-simple** plugin (KIERTSCHER, 2018), an NPM module that makes it easy to create transformations in text files (such as the metadata files).
- **gulp-newer** plugin (SCHAUB, 2019), an NPM module that prevents repeated file processing (makes processing occur only to unprocessed files), which is particularly advantageous when new input data is added for processing.

Thus, the project corresponding to the generation of the features of the present work has the following structure (shown in Figure 9):

- `"gulpfile.js"` file (entry point).
- `"package.json"` file, which declares the list of required NPM modules for the project installation (mainly gulp plugins).
- `"wav"` directory, which contains a copy of the input audios (to avoid corruption of the original files).

- "csv" directory, which contains a copy of the metadata files (for the same reason).
- "emd.py" file, which consists of a Python script for EMD decomposition of each heart cycle in the "wav" directory.
- Directory "emd", which contains the EMD decompositions generated by the Python script.
- Directory "emd.c", which contains the audio and graphic files resulting from the segmentation of IMFs through the method proposed by Boutana, Benidir and Barkat (2014).
- "lib" directory, which contains the Octave script for HVD decomposition of each heart cycle of the "wav" directory, as well as ".js" extension files that declare common functions and file transformation functions detailed below.
- "hvd" directory, which contains the HVD decompositions generated by the Octave script.

Figure 9 – Architecture of the project for input data feature extraction



Source: Own authorship

"gulpfile.js" is the file that declares the gulp tasks that perform all the operations mentioned above:

- "copy-wav-files", which is responsible for detecting new input audios and for backing them up to the "wav" directory.
- "copy-csv-files", which requires the execution of the previous task. It is responsible for locating new metadata files, creating a backup of them to the "csv" directory and triggering the custom transformations described in subsections 3.1.1 and 3.1.2. Such transformations can be performed safely, because the task requirement ensures that the

input audios from the previous task are already available.

- `"copy-emd-files"`, which requires the execution of the previous task. It takes care of moving the new EMD decomposition files (generated by the `"copy-csv-files"` task) into the `"emd"` directory.
- `"default"` (main task), which requires the previous task — and, as a consequence, all other tasks. It triggers the custom transformations of Subsection 3.1.1, and then the main transformation of Subsection 3.1.3.

3.1.1 Metadata loading transformations

The metadata files, as presented in the work of Andrade (2018), inform the time instants (in seconds) in which heart sounds (S1 and S2) occur in their respective input audios. However, the metadata files are CSV formatted and require transformation to be usable in calculations after being loaded into memory. Another issue concerns the loading of an input audio (WAV) into memory, when it becomes a vector of numbers (samples). In this vector, the positions of their S1s and S2s are addressed through sample indices (functions of the sampling rate). Therefore, positions in seconds need to be translated into sample indices. Moreover, before the most costly transformation (described in Subsection 3.1.2), downsampling is applied on this vector in order to obtain another vector with fewer samples. In this new vector, the positions of S1s and S2s have different indices. Finally, when processing a heart cycle, it is often useful to know not only the S1 and S2 indices of this cycle, but also the S1 index of the next one — since each new S1 effectively marks the end of the previous cycle and the beginning of a new one. An example of this is the very definition of "end" of a heart cycle in the auscultation domain: each S1 marks the end of one cycle and the beginning of the next one[1].

In order to create a summary of this information useful for each heart cycle and also to indicate the 9 cycles of each patient that will be effectively processed, a set of transformations were created that act on each CSV file of the `"copy-csv-files"` task and on its input WAV audio (represented by the sample vector $x[j]$, where j indicates the sample index[2]). These transformations summarize, for each heart cycle:

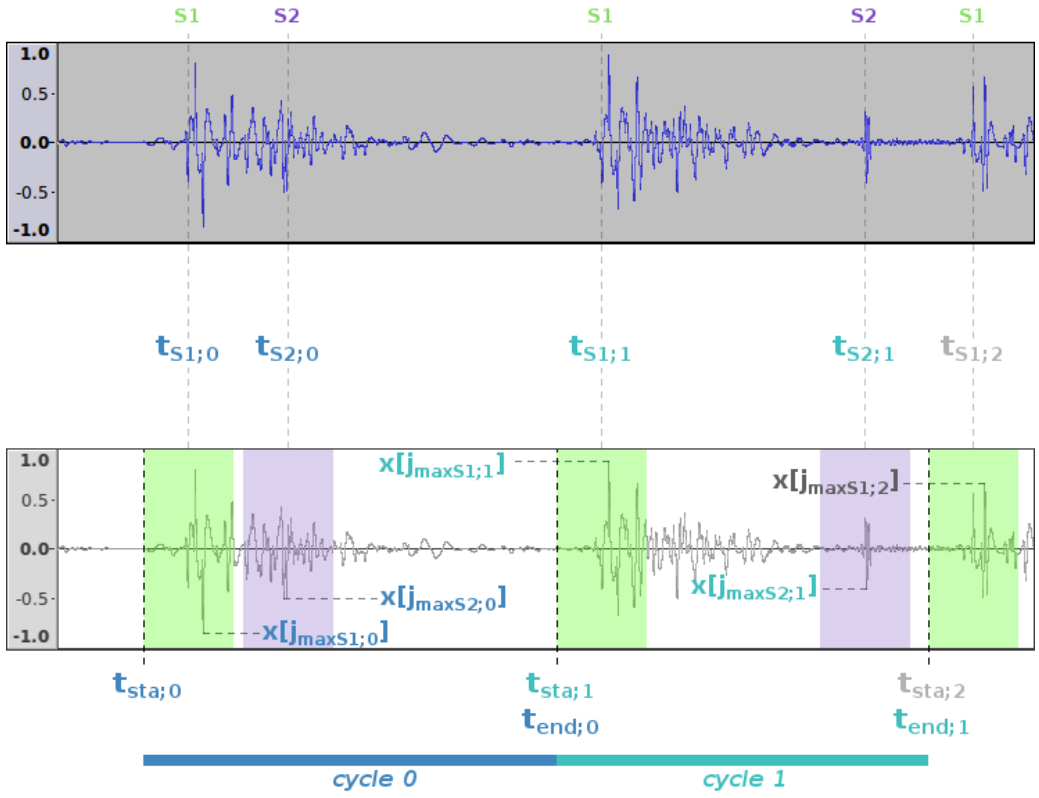
- f_S: input audio sampling rate (44,100 Hz for all audios used).
- f_{DS}: sampling rate after downsampling (2,756.25 Hz).
- $t_{S1;i}$ and $t_{S2;i}$: times of occurrence of S1 and S2 of a cycle in the input audio, where i corresponds to the cycle index (as shown in Figure 10).
- $j_{S1;i}$ and $j_{S2;i}$: corresponding indices in the input audio vector.

[1] Because of this, and since 10 or more pairs of S1 with S2 were recorded for each patient, one can only guarantee that each patient has 9 complete heart cycles.

[2] At this point, it is useful to introduce the convention adopted in the rest of this work. In addition to the letters x (always denoting samples loaded directly from a record or a segment of it) and j (index for samples under the same conditions), k was used to address samples of vectors originated by downsampling. The asterisk symbol ($*$) was used to mark indices of vectors corresponding to segments of other vectors. For instants of time, always measured in seconds, the letter t was used.

- $k_{S1;i}$ and $k_{S2;i}$: corresponding indices in the downsampled vector.
- $t_{S1;i+1}$: time (in seconds) of occurrence of S1 in the next cycle of the input audio.
- $j_{maxS1;i}$ and $j_{maxS2;i}$: indices of the samples with the maximum absolute value in the vicinity of S1 and S2. By "vicinity", it is meant the set of samples temporarily distant from $t_{S1;i}$ or $t_{S2;i}$ by a maximum of 50 ms (highlighted green and purple segments in Figure 10).
- $k_{maxS1;i}$ and $k_{maxS2;i}$: corresponding indices in the downsampled vector.
- $x[j_{maxS1;i}]$ and $x[j_{maxS2;i}]$: maximum sample value in the vicinity of S1 and S2 in the input audio vector.
- $t_{sta;i}$ and $t_{end;i}$: start and end times of the cycle in the input audio, also illustrated in Figure 10. Consider that each cycle starts 50 ms before its S1, and ends 50 ms before the next S1 (which results in $t_{sta;i+1} = t_{end;i}$).
- $j_{sta;i}$ and $j_{end;i}$: corresponding indices in the input audio vector.
- $k_{sta;i}$ and $k_{end;i}$: corresponding indices in the downsampled vector.
- $x_i[j*]$: segment corresponding to the heart cycle in the input vector, which equals the samples of $x[j]$ with $j_{sta;i} <= j < j_{end;i}$. Since it is a segment of $x[j]$, we have that $j* = j - j_{sta;i}$.

Figure 10 – Audio segment with two heart cycles and some of the summarized data



Source: Own authorship

This data will be used in the transformations described in the following subsections. In addition, if the EMD and HVD decompositions of the incoming audio heart cycles are already

available (details in Subsection 3.1.2), the summary of each of them also carries the following data[3]:

- $y_{l;i}[k*]$: vectors resulting from the EMD decomposition of the downsampled heart cycle (where $k*$ is the sample index and l varies from 0 to $L_i - 1$, with L_i equal to the number of IMFs in the cycle plus 1). Each vector $y_{l;i}[k*]$ represents a different IMF or the residual function (in the case of the index $l = L_i - 1$) and can be understood as a simple numerical vector with the same amount of elements as $x_{DS;i}[k*]$ (defined in Subsection 3.1.2).
- $\sum_{l=0}^{L_i-1} y_{l;i}[k*]$: sum of the vectors corresponding to the IMFs and the residual function, resulting in the vector of the original samples of the heart cycle after downsampling ($x_{DS;i}[k*]$, where DS stands for downsampling).
- $h_{0;i}[j*]$: vector of samples of the main component of the HVD decomposition of the heart cycle.

3.1.2 Decomposition transformations

Several of the features described in the present work are based on EMD or HVD decompositions. However, the generation of these decompositions depends on the data made available as a result of the metadata loading transformations of Subsection 3.1.1 — in particular, the start and end indices of heart cycles. In addition, decompositions are computationally costly operations, which translates into a high processing time. Thus, it was decided to implement the decomposition operations as one of the transformations of the task `"copy-csv-files"`; since such a task acts only on new input files, the decomposition transformations are performed only once for each of them.

For the EMD decomposition, a transformation was created that processes the segments corresponding to the heart cycles ($x_i[j*]$) of each input audio. The transformation sends each segment to the Python script `"emd.py"`, which decomposes the samples vector using the PyEMD library (LASZUK, 2019). The integration between the JavaScript and Python files takes place through the NPM module `pytalk` (NPM, Inc., 2016). Due to the high computational cost of the EMD, before effectively decomposing the heart cycle, the Python script **downsamples the audio with factor 16/1**, resulting in the vector $x_{DS;i}[k*]$ of sampling frequency equal to 2,756.25 Hz. Among the EMD decomposition functions available in the library, the simple `"emd"` was chosen. Of the parameters used in the call (`"std_thr"`, `"svar_thr"`, `"total_power_thr"` and `"range_thr"`), only the first parameter was modified (to "20.0") from the default value ("0.2"), as an additional measure to reduce the execution time. The remaining parameters remained unchanged ("0.001", "0.005" and "0.001", respectively). As a result of the decomposition, for each heart cycle of each incoming WAV audio file, a file (of `".npy"` extension) containing the IMFs and the residual function and a graphic (`".png"`

[3] In addition to the aforementioned convention, the letters y (to denote vectors originated by EMD decomposition) and h (similarly, to HVD decomposition) are added here.

extension) for inspection of these signals are generated. NPY is a simple format for writing Python vectors to disk (The SciPy community, 2019), and can be read again in both Python and any other programming language capable of executing the reading routine. It is through a JavaScript implementation adapted from Abdennur (2016) that the vectors ($y_{l;i}[k*]$) of the IMFs and the residual function, once created, can be made available for reading as described at the end of Subsection 3.1.1.

Similarly, a transformation was created for the HVD decomposition. However, the existing implementation of this decomposition consists of a Matlab/Octave script (FELDMAN, 2019). In this case, integration with the program was done through an Octave command line call from a `".js"` script file. Each heart cycle ($x_i[j*]$) is sent to the Matlab script for low-pass pre-filtering with cutoff frequency of 2 kHz. The filter used is a 12th-order IIR Butterworth. The beginning and end of each heart cycle are identified exactly as in the EMD transformation. Then, the HVD decomposition is called to generate 5 components ($h_{m;i}[j*]$, with m ranging from 0 to 4). The parameter `"fp"` was determined experimentally for the value that imposed maximum sensitivity ("0.0051") — that is, it kept the frequency band of the main component as small as possible, transferring the energy of other frequency bands to the following components. Finally, the decomposition of each heart cycle is saved in a WAV file containing 5 channels (one for each generated component).

3.1.3 Main task

When the main task is executed, all directories except `"emd.c"` and `"emd.se"` are already populated. Each file needed to generate the features is created, including those resulting from the decomposition transformations mentioned in Subsection 3.1.2. Therefore, all data listed in Subsection 3.1.1 is available.

Thus, the `"default"` task starts loading the metadata and sample vectors and proceeds to the **main transformation**, which effectively calculates values for the features in sections 3.2, 3.3 and 3.4.

3.2 BASIC FEATURES

All **features** whose calculation is immediate from the input audios and their metadata were considered basic. All of them are of type 2 (as defined in Section 1.5).

The first basic feature is the duration of the heart cycles (CD — Cycle Duration). Each value is given by Equation (6), is measured in seconds and is related to the patient's heart rate.

$$CD_i = t_{end;i} - t_{sta;i} \tag{6}$$

The second is systole-relative time (SRT). This is a ratio between the duration of

systole and the duration of the complete cycle. This feature is given by Equation (7).

$$SRT_i = \frac{t_{S2;i} - t_{S1;i}}{t_{S1;i+1} - t_{S1;i}} \tag{7}$$

The last basic feature concerns the ratio between the maximum values in the regions of S1 and S2 (RA12 — Relative Amplitude of the Peaks of S1 and S2). It is given by Equation (8).

$$RA12_i = \frac{x[j_{maxS1;i}]}{x[j_{maxS2;i}]} \tag{8}$$

3.3 FEATURES BASED ON THE LITERATURE

A second category of features is based on existing algorithms. During the literature review, methods to express metrics of a record (represented by the vector $x[j]$) that could be correlated to the degree of murmur were searched. The metrics found were converted into type 2 features.

The first one is the number of vectors $y_{l;i}[k*]$ (NIMF — Number of IMFs) generated in the EMD decomposition of the downsampled vector of each heart cycle ($x_{DS;i}[k*]$). Equation (9) expresses the value of the feature in the notation used in Subsection 3.1.1. Although it is an apparently trivial value, it is necessary to remember that the IMFs produced are the components found by the EMD technique in the input signal, and therefore more complex signals (with more components) tend to generate more IMFs.

$$NIMF_i = L_i \tag{9}$$

The second feature based on algorithms in the literature refers to the energy of the heart cycle (CE — Cycle Energy). Two variants were implemented: the energy of the cycle after downsampling (CE.DS), represented by Equation (10), and the energy of the main component of the cycle after the HVD decomposition (CE.HVD), of Equation (11).

$$CE.DS_i = E_{x_{DS;i}} \tag{10}$$

$$CE.HVD_i = E_{h_{0;i}} \tag{11}$$

The next features are based on an algorithm presented by Boutana, Benidir and Barkat (2014) to segment sounds and murmur in a heart cycle. For this, the correlation coefficient of each IMF with the vector that generated it (vector of the cycle before EMD decomposition) is determined by Equation (12).

$$\rho_{l;i} = \frac{\sum_k x_{DS;i}[k*] \cdot y_{l;i}[k*]}{\sqrt{\sum_k x_{DS;i}^2[k*] \sum_k y_{l;i}^2[k*]}} \tag{12}$$

Next, the index l of the IMF with the highest correlation coefficient with the cycle is found (and denominated l_{max}). Finally, IMFs of index less than l_{max} are summed to form

a vector of samples considered to contain only noise (either murmur or ambient), and the other IMFs are summed in a vector considered as containing only pure heart sounds. These two vectors are given by equations 13 and 14, respectively. At this point, WAV audios and the segmentation graph in the "emd.c" directory are also generated for inspection purposes.

$$x_{N.C;i}[k*] = \sum_{l=0}^{l_{max}-1} y_{l;i}[k*] \tag{13}$$

$$x_{HS.C;i}[k*] = \sum_{l=l_{max}}^{L-1} y_{l;i}[k*] \tag{14}$$

The following features are produced from the results of segmentation by correlation:

- NNIMF.C (Number of Noise IMFs based on the Correlation method): number of IMFs considered "noise" by the segmentation method by correlation.
- NSR.C (Noise to Signal Ratio based on the Correlation method): ratio between the energy of $x_{N.C;i}[k*]$ and the energy of the vector before decomposition.
- HSSR.C (Heart Sound to Signal Ratio based on the Correlation method): ratio between the energy of $x_{HS.C;i}[k*]$ and the energy of the vector before decomposition.

3.4 PAIR-COMPARISON FEATURES

The last features produced consist of a kind of comparison between pairs of cycles of the same input audio. More precisely, they express the minimum possible energy of error between cycles (subtraction of cycles) — hence their name, MICE (Minimum InterCycle Error). As a result, these features are of type 3. In all, 6 MICE variants were generated (MICE.DS, MICE.HVD, MICES1.DS, MICES1.HVD, MICES2.DS and MICES2.HVD), each with particularities that will be explained as the general algorithm is described next.

The first step in the calculation of the MICE consists of the choice of the **beginning and end samples** of the mandatory **comparison region** of each heart cycle. The MICE algorithm guarantees that at least this region will be included in the calculation of the error, affecting the value of the feature. In addition, a sample necessarily between the beginning and the end of the cycle is chosen as the search **pivot**. The process of comparing a pair of cycles can be understood as a sliding of one cycle over the other in search of the position in which they best overlap while keeping a maximum distance between the two pivots, as will be made clear in the sequence. The times corresponding to the beginning, pivot and end of each heart cycle for each MICE variant are listed in Table 1 and illustrated in Figures 11 and 12 for the MICE.HVD and MICES1.HVD variants, respectively. The table also shows the vectors (derived from the cycle) from which the samples are extracted for the next steps.

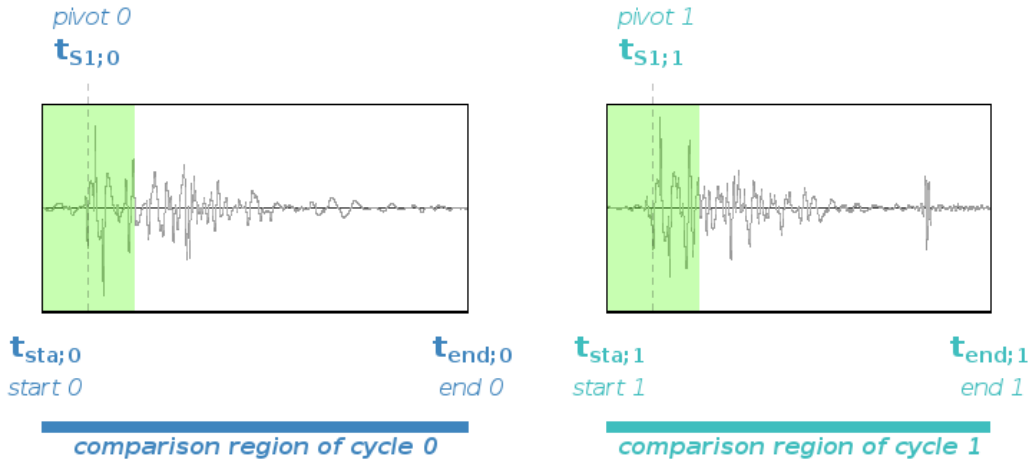
The second step is the determination of all possible pairs of heart cycles; for this, each cycle was combined once with each of its successors in the audio. The number of possible pairs as a function of the number of heart cycles in a record is given by Equation (15). Table 2

Table 1 – Parameters of the comparison region for each variant of the MICE feature.

Variant	Start [s]	Pivot [s]	End [s]	Vector	Step
MICE.DS	$t_{sta;i}$	$t_{S1;i}$	$t_{end;i}$	$x_{DS;i}[k*]$	1 sample
MICE.HVD				$h_{0;i}[j*]$	16 samples
MICES1.DS	$t_{S1;i} - 0.05$		$t_{S1;i} + 0.05$	$x_{DS;i}[k*]$	1 sample
MICES1.HVD				$h_{0;i}[j*]$	16 samples
MICES2.DS	$t_{S2;i} - 0.05$	$t_{S2;i}$	$t_{S2;i} + 0.05$	$x_{DS;i}[k*]$	1 sample
MICES2.HVD				$h_{0;i}[j*]$	16 samples

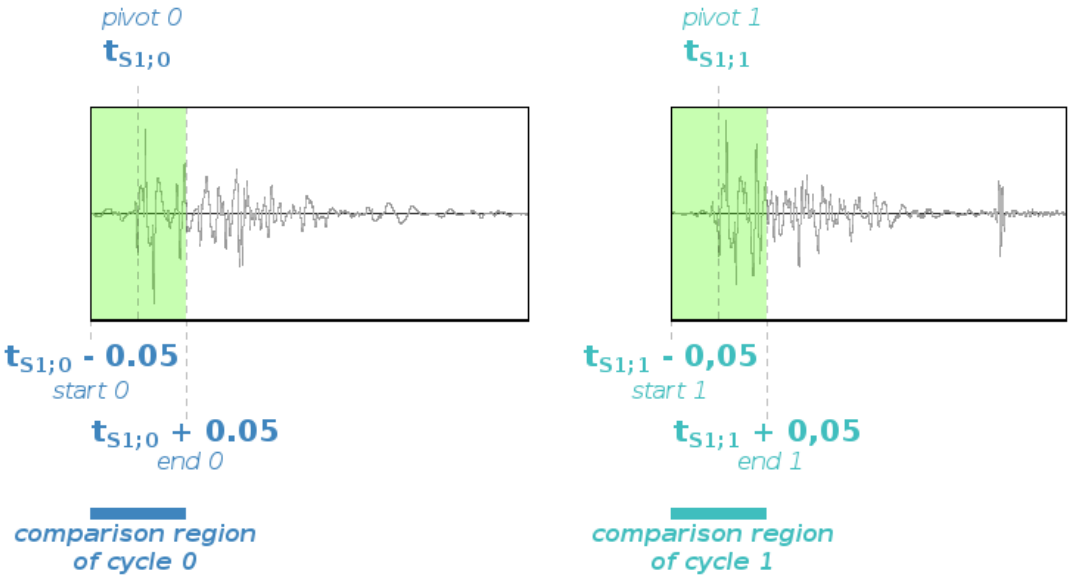
Source: Own authorship

Figure 11 – Visualization of the parameters of the MICE.HVD variant



Source: Own authorship

Figure 12 – Visualization of the parameters of the MICES1.HVD variant



Source: Own authorship

shows the pairs generated for an input audio with 9 heart cycles; the numbers in the row and column headings correspond to the indices of the cycles, while the numbered cells indicate the index of the pair formed by those cycles. Unnumbered cells indicate invalid combinations, i.e., those already produced before and those of a cycle with itself.

$$p(c) = \frac{c \cdot (c-1)}{2} \tag{15}$$

Table 2 – Example of pairs generated for an audio with 9 heart cycles.

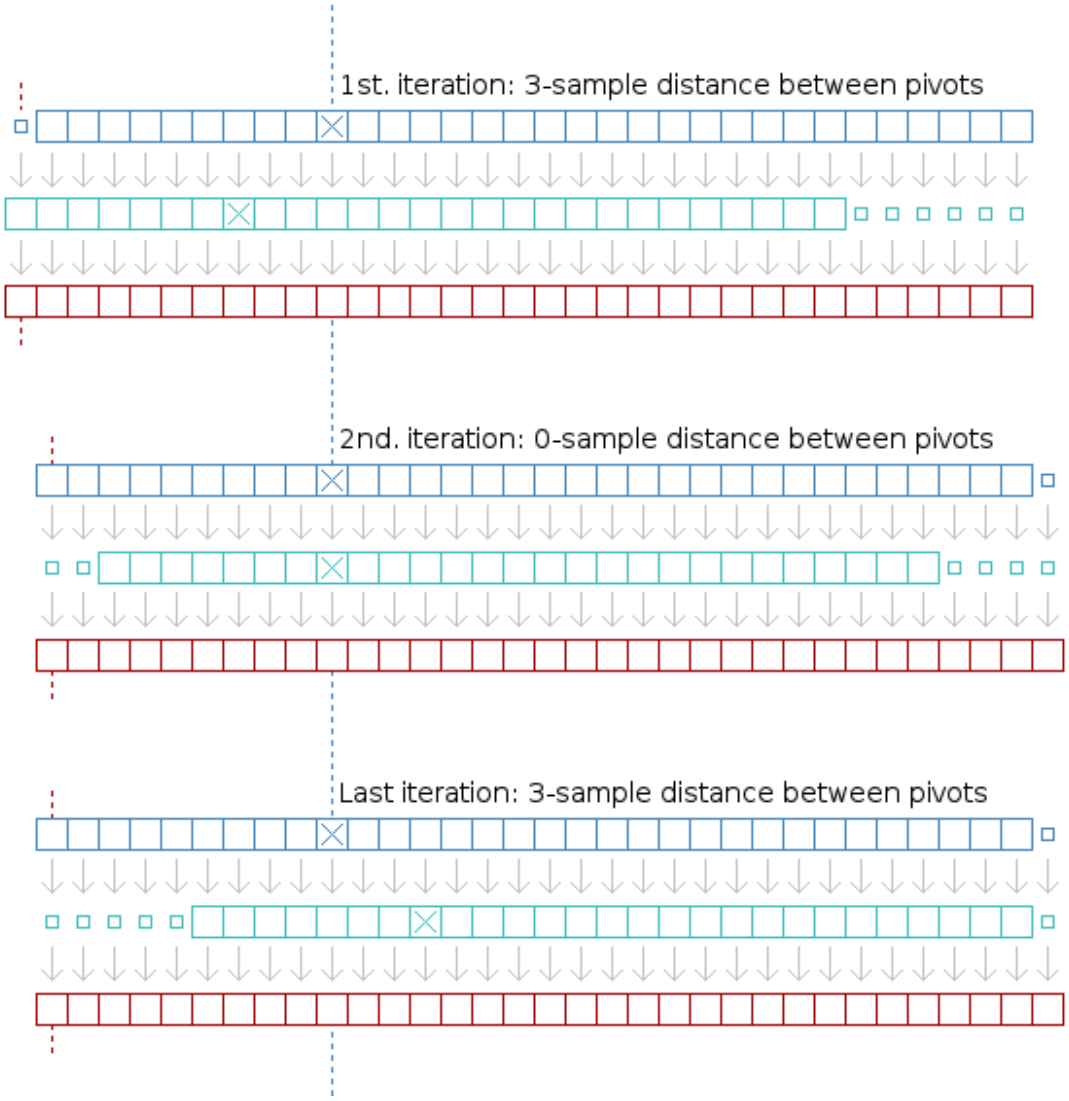
Indices	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
0	-	0	1	2	3	4	5	6	7
1	-	-	8	9	10	11	12	13	14
2	-	-	-	15	16	17	18	19	20
3	-	-	-	-	21	22	23	24	25
4	-	-	-	-	-	26	27	28	29
5	-	-	-	-	-	-	30	31	32
6	-	-	-	-	-	-	-	33	34
7	-	-	-	-	-	-	-	-	35
8	-	-	-	-	-	-	-	-	-

Source: Own authorship

For each pair of cycles generated, the MICE algorithm iterates over the distance between the two pivot points. That is, conceptually keeps one of the two vectors stopped while sliding the other from left to right — starting with a distance equivalent to 25 ms of samples between their pivots (about 1100 samples), gradually reducing that difference until the moment of their perfect alignment (0 distance samples between pivots), and then the sliding continues in the same direction (again increasing the difference to a sample amount equivalent to 25 ms). Figure 13 illustrates these iterations for a maximum distance between pivots of three samples (equivalent to only 68 μs of the input audio). The pivotal samples of the heart cycles involved in the calculation are marked with an "X". In practice, due to performance issues, the samples are not moved in memory (the distance is only counted in a variable). The amount of samples conceptually "slided" at each iteration (**step** chosen) can also be found in Table 1, and was shown in Figure 13 as being of three samples. At each iteration, the error between the two cycles (blue vectors of the figure) is established by its subtraction (resulting in the red vector). The vector of the difference between two cycles is always generated with the same length for the same pair — a number of samples sufficient to cover at least the two mandatory comparison regions (larger squares in the figure), regardless of the specific iteration. Therefore, for each pair of cycles, depending on the length of the comparison regions and the position of their pivots, samples not belonging to the comparison regions are required for the calculation of the feature. These samples were represented by smaller squares in the figure, and may even belong to adjacent heart cycles if necessary. Another observation from Figure 13 is that the

beginning of the vector of the difference must always be in line with the beginning of one of the other two vectors.

Figure 13 – Example of MICE for both step and maximum distance of three samples



Source: Own authorship

Finally, the energy of each error vector is calculated, and the lowest error energy found for the respective pair of cycles is adopted as the value of the feature.

In this way, one can summarize the process by asserting that the MICE.DS and MICE.HVD variants consist of a comparison of complete cycles for each possible pair; variants MICES1.DS and MICES1.HVD compare only the regions of S1; finally, MICES2.DS and MICES2.HVD compare only the regions of S2. The three DS variants make use of samples after downsampling (without additional filters); the other three only use samples of the main HVD component of the cycle.

3.5 RESULTS

The feature-processing gulp script was run on the input data of the 56 patients on a laptop with a Linux Mint 17.3 Cinnamon 64-bit operating system, with an Intel Core i7-3520M 2.9 GHz processor and 8 GB of memory. The total execution time was 4 hours and 25 minutes. Of these, 29 minutes were consumed by the `"copy-csv-files"` task, mostly for the EMD decomposition of downsampled audio of each heart cycle, and 2 seconds by `"copy-emd-files"` (which moves files resulting from decomposition to a directory of its own). Most of the time (3 hours and 56 minutes) was used by the main task (`"default"`) to effectively generate the features, which corresponds to about 89% of the total execution.

Since only 9 heart cycles of each patient were effectively processed, an average of 31.5 s per heart cycle was obtained. However, the runtime of the MICE feature variants increases nonlinearly with the number of cycles — in fact, the amount of generated values progresses quadratically according to Equation (15). In addition, this feature set accounted for around 97% of the main task time (73% for MICE.HVD, and an additional 24% for other variants). Thus, its share in the total execution time is approximately 86%, which means that the impact on the average execution time per cycle should change considerably if the presented features are implanted in an application with a higher or lower number of cycles per patient.

The data produced in response to the input data from the 56 patients are described in Table 3. Table 4 exemplifies the output format of the processing script for a single patient. Most of the data was omitted for space reasons, so ellipses ("...") were used to indicate missing segments.

Then, to make the visualization and the in-depth analysis of the results feasible, the need arose to convert the data to a tabular format. This means that the output format of the processing script (a list of values for each patient) must be converted to a table, in which each line must represent a vector of distinct features associated with the same heart cycle. As the script generated more than one type of value (namely types 2 and 3), and how one can understand each patient's clinical attributes — age, weight, sex, clinical signs and degree of murmur — as a feature of type 1, it was necessary to apply a criterion of union of values of features, summarized in the following set of rules:

- Each feature corresponds to a column of the table.
- Each pair of heart cycles produces two table rows (one for each cycle of the pair).
- **Feature of type 1**: Each value is repeated on all rows relative to patient cycles.
- **Feature of type 2**: Each value is repeated on all rows relative to the cycle with which the value matches.
- **Feature of type 3**: Each value is repeated only on the two lines relative to the cycle pair.

An example of union of 6 fictitious features of 4 heart cycles belonging to the same patient is shown in the diagram of Figure 14. On the left is a list of values resulting from the processing script (similar to the actual values shown in Table 4). On the right, the table

Table 3 – Quantity of values produced by feature.

Feature	Type	Values per patient	Values (total)
CD	2	9	504
SRT	2	9	504
RA12	2	9	504
NIMF	2	9	504
CE.HVD	2	9	504
CE.DS	2	9	504
NNIMF.C	2	9	504
NSR.C	2	9	504
HSSR.C	2	9	504
MICE.HVD	3	36	2016
MICE.DS	3	36	2016
MICES1.HVD	3	36	2016
MICES1.DS	3	36	2016
MICES2.HVD	3	36	2016
MICES2.DS	3	36	2016
TOTAL		297	16632

Source: Own authorship

resulting from the union of the represented features. It should be noted that the values of the two features of type 1 are repeated in all rows of the table, since all rows in the example belong to the same patient to whom those values refer. The values of the two features of type 2 appear three times each, since this is the number of pairs from which each cycle participates (in the specific case of 4 heart cycles). Finally, the values of the two features of type 3 are repeated only in the two lines referring to their respective pair of cycles.

Using this same procedure, a single table was generated, in CSV format, containing the values of the 56 patients of the study. The 20 columns corresponding to the features were:

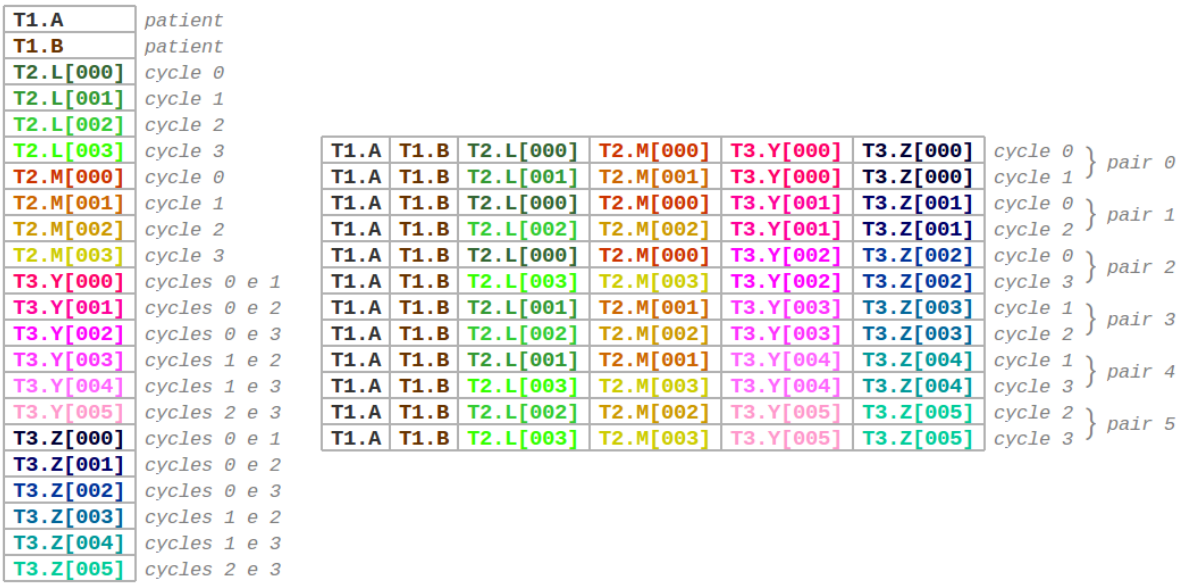
- **Weight**: in kilograms;
- **Age**: in years;
- **Gender**: 0 for "female", 1 for "male";
- **Presence of clinical signs**: 0 for "no", 1 for "yes";
- **CD**: cycle duration in seconds;
- **SRT**: systole relative time in the cycle, without unit;
- **RA12**: relative amplitude of S1 compared to S2, without unit;
- **NIMF**: number of IMFs in the cycle;
- **CE.HVD**: energy of the main HVD component of the cycle;
- **CE.DS**: heart cycle energy after downsampling;
- **NNIMF.C**: number of IMFs considered "noise" by the correlation method;
- **NSR.C**: ratio between noise energy (according to the correlation method) and signal energy;

Table 4 – Features generated for a patient.

Feature[cycle index]	**Value**	**Unit**
CD[000]	0.39473922902494335	[seconds]
...		
CD[008]	0.5166439909297056	[seconds]
SRT[000]	0.37409967980164444	ratio
...		
SRT[008]	0.2857148389245382	ratio
RA12[000]	3.155739828971236	ratio
...		
RA12[008]	1.9462574549353346	ratio
NIMF[000]	9	[IMFs]
...		
NIMF[008]	9	[IMFs]
CE.HVD[000]	143361393774	normalized energy
...		
CE.HVD[008]	145872042932	normalized energy
CE.DS[000]	13386328881.263596	normalized energy
...		
CE.DS[008]	13724180088.009422	normalized energy
MICE.HVD[000]	299054067055	normalized energy
...		
MICE.HVD[035]	99111380205	normalized energy
MICE.DS[000]	27220506308.418636	normalized energy
...		
MICE.DS[035]	9986981795.928823	normalized energy
MICES1.HVD[000]	157747451410	normalized energy
...		
MICES1.HVD[035]	23024550143	normalized energy
MICES1.DS[000]	13987740875.395214	normalized energy
...		
MICES1.DS[035]	2099924885.6990712	normalized energy
MICES2.HVD[000]	27782714021	normalized energy
...		
MICES2.HVD[035]	8774660887	normalized energy
MICES2.DS[000]	2329522992.5461106	normalized energy
...		
MICES2.DS[035]	768094647.4241908	normalized energy
NNIMF.C[000]	2	[IMFs]
...		
NNIMF.C[008]	1	[IMFs]
NSR.C[000]	0.26621420772672416	ratio
...		
NSR.C[008]	0.38460120055497865	ratio
HSSR.C[000]	0.6588263347411786	ratio
...		
HSSR.C[008]	1.283908184226331	ratio

Source: Own authorship

Figure 14 – Example of joining the values of 4 cycles of the same patient in a table



Source: Own authorship

- **HSSR.C**: ratio between the energy of the heartbeat sounds (according to the correlation method) and the energy of the signal;
- **MICE.HVD**: minimum energy between each pair of cycles after HVD filtering;
- **MICE.DS**: minimum energy between each pair of cycles after downsampling;
- **MICES1.HVD**: minimum energy between each pair of S1s after HVD filtering;
- **MICES1.DS**: minimum energy between each pair of S1s after downsampling;
- **MICES2.HVD**: minimum energy between each pair of S2s after HVD filtering;
- **MICES2.DS**: minimum energy between each pair of S2s after downsampling;
- **Degree of murmur**: 0 for "without murmur", and integer of 1 to 6 for the degree of murmur.

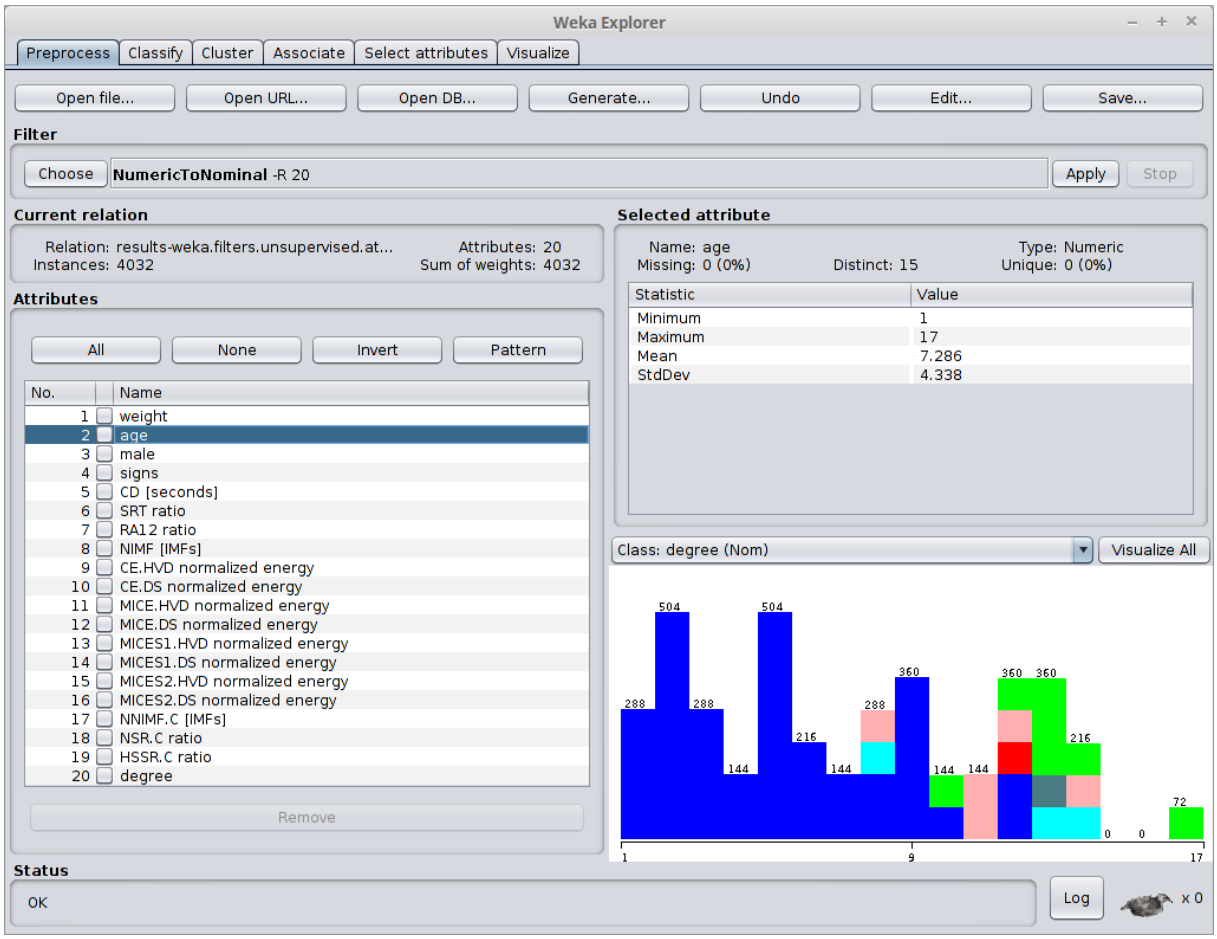
The resulting table has 4032 rows, 72 for each patient. Its CSV file was imported into the Weka software, which is defined as a collection of machine learning algorithms for data mining (FRANK; HALL; WITTEN, 2016). The version used was 3.8.3.

3.5.1 Analysis on the Weka software

After importing the CSV table, the "degree of murmur" feature was temporarily converted to a nominal type (used by Weka to represent non-numeric attributes whose values must come from a predetermined list of values). This allowed the initial inspection of the distribution of the columns, as shown in Figure 15. The upper part of the figure exposes the filter (`"NumericToNominal"`) applied on the column "degree of murmur" (option `"-R 20"`), while the lower one shows the graph of the distribution of the selected attribute ("age") degree. The blue color was chosen by the software to express degree equal to 0. Immediately, therefore,

it was perceived that the analysis should proceed with caution regarding age, since none of the studied patients with murmur was 7 years old or less — which could lead the machine learning algorithms to make improper predictions, since this is a sampling phenomenon that does not reflect the reality of the population. The other clinical attributes do not present this problem, as it is clear in the first line of Figure 16, which presents all the distributions (including age's one).

Figure 15 – Visual inspection of the attribute "age" according to the degree of murmur

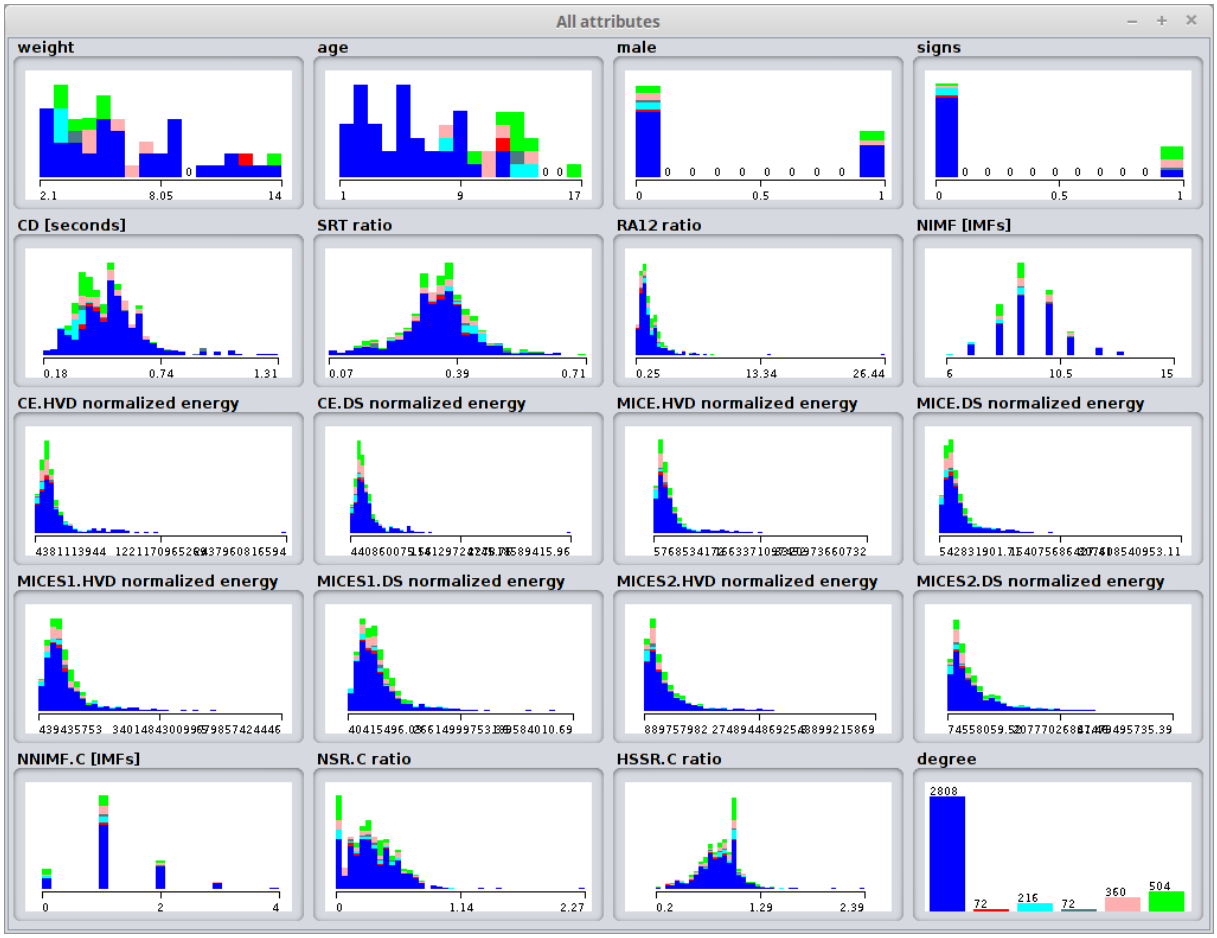


Source: Own authorship

Another common observation that can be made about the graphs presented by Weka software is that no attribute can be used alone to classify the degree of murmur (shown in the last column of the last line). This means that the problem necessarily depends on a subset of mutually complementary attributes to enable a regression to the degree of murmur. This is precisely the role of the data table generated in the present work — to act as a training set so that a regression algorithm is able to find a model capable of estimating the degree of heart murmur of new patients.

After visual inspection of the available attributes, the conversion of the column "degree of murmur" to the nominal type was undone (restoring it to the numerical type), and then proceeded to the investigation of regression algorithms. Through experimentation,

Figure 16 – Visualization of all attributes according to degree of murmur



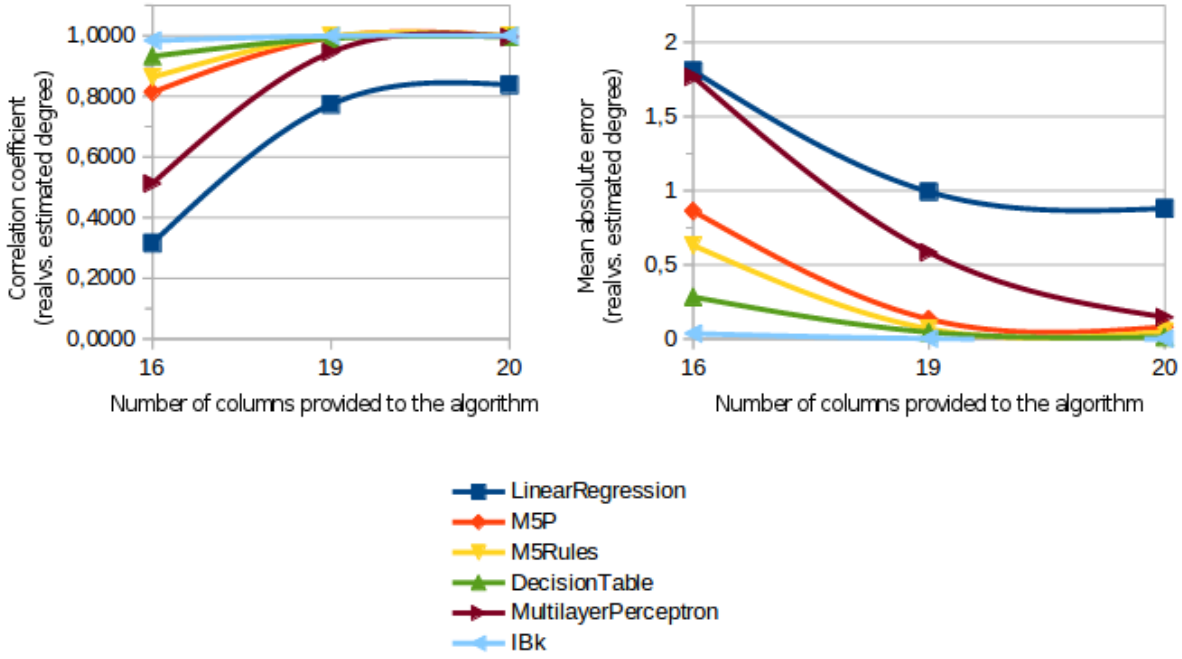
Source: Own authorship

algorithms were found that, even in the standard Weka configurations, established correlation coefficient greater than 0.99 between the degree of murmur and its estimate according to the model generated. These are: `"trees.M5P"`, `"rules.M5Rules"`, `"rules.DecisionTable"` and `"functions.MultilayerPerceptron"`. The left half of Figure 17 shows the correlation coefficients at the end of training for the algorithms searched, with default settings, by providing them with the complete table (20 attributes), the table without the "age" column (19 attributes) and without clinical indicators (16 attributes, i.e., all except weight, age, sex and presence of clinical signs). In any case, the last column fed to these algorithms is always the "degree of murmur" (variable to which regression is intended). This column acts merely as an output variable, that is, it is never used by the algorithms to produce an estimate of the degree of murmur. The right half displays the MAEs at the end of the training for the same scenarios. The higher the correlation coefficient and the lower the mean absolute error, the better the algorithm performance. The `"functions.LinearRegression"` was used as a basis for comparison. Its performance is not good by acting simply by establishing a linear combination of the input columns — that is, each column is simply associated with a scalar multiplier, and then all of them are summed and added of one final scalar to constitute the regression of the output

column.

As can be seen in the two graphs, the performance of most of the algorithms is significantly reduced in the absence of the attribute "age", which shows great dependence on the phenomena occurred during the sampling phase and may also reflect a reduced **generalization** capacity — that is, the algorithm's ability to make accurate predictions even when confronted with a completely unknown line (never used for training purposes). It is worth mentioning that whenever the table of only 16 attributes was used, the only information that the algorithm has to calculate the degree of murmur estimate are the features generated with the processing presented in this work.

Figure 17 – Correlation and error of regression algorithms for different sets of features



Source: Own authorship

The `"lazy.IBk"` classifier was one of those with a very high correlation coefficient (0.9999) in standard configurations and with all attributes (20 columns), together with mean absolute error of 0.0005 — meaning that the training was concluded with estimates of the degree of murmur (floating-point numbers) that, on average, were 0.0005 degrees above or below the actual degree (integer). This allows one to conclude that the vast majority of estimates will correspond to the true degree of murmur after rounding. However, some even more interesting properties were perceived in IBk. The first concerns maintaining its good performance by removing the "age" column — indeed, a correlation coefficient of 0.9834 and a MAE of 0.0361 were obtained even by eliminating all 4 clinical indicators. Second, good performance was also maintained by varying the parameter K, which basically reflects the generalization of the model produced by the algorithm (GOLLAPUDI; LAXMIKANTH,

2016). Finally, the IBk is actually the implementation in the Weka of the already known k-NN (k-Nearest Neighbors) algorithm, which allows the fine-tuning of the generalization not only through parameter K ("number of nearest neighbors"), but also via the distance calculation algorithm.

The k-NN works for regression by assigning as the estimated output for an input (whose actual output may be unknown) the average of the already known outputs for the K training set instances closest to that input, according to the chosen distance calculation algorithm (GOLLAPUDI; LAXMIKANTH, 2016). The K shortest instances for which the estimation is desired are called the **"K" nearest neighbors**, and consist of the training set records most similar to the new line. The distance algorithm may be that of the Euclidean distance (expressed by Equation (16)), the Manhattan distance (Equation (17)) or basically any other able to establish a metric (or pseudo-metric) between two vectors (in the formulas, $v_A[i]$ and $v_B[i]$) of the same dimension (D). The dimension, in turn, corresponds to the number of attributes (columns) whose values are always known, both in the case of the new entry and in the case of the records of the training set. For example, being $v_l[i]$ the vector of attributes (except degree of murmur) of the l-th line of the features table, the degree prediction for a new vector $v_{NOVO}[i]$ (features whose degree of murmur is unknown), for $K = 3$ and Manhattan distance is given by the mean of the degrees of the 3 lines (indices l_P, l_Q and l_R) whose respective distances $d_{MANHATTAN}(v_{NOVO}, v_l)$ are the smallest within the already known table (training set). That is, in this example, the 3 lowest distances found would be $d_{MANHATTAN}(v_{NOVO}, v_{l_P})$, $d_{MANHATTAN}(v_{NOVO}, v_{l_Q})$ and $d_{MANHATTAN}(v_{NOVO}, v_{l_R})$, and therefore the estimate of the degree of murmur for the new line would correspond to the arithmetic mean of the degrees (already known) of lines l_P, l_Q and l_R.

$$d_{EUCLIDIANA}(v_A, v_B) = \sqrt{\sum_{d=0}^{D-1} |v_A[d] - v_B[d]|^2} \tag{16}$$

$$d_{MANHATTAN}(v_A, v_B) = \sum_{d=0}^{D-1} |v_A[d] - v_B[d]| \tag{17}$$

Due to its less dependence on the clinical attributes (outcome of a greater use of the features resulting from the processing) and the possibility of adjusting parameters to obtain greater or lesser generalization, IBk was selected for application in the present work. Its default settings on Weka are $K = 1$ and Euclidean distance. Figure 18 shows the correlation coefficients (left) and the MAEs (right) of the base of comparison algorithm, `"functions.LinearRegression"` (in default settings), and the IBk with several distance calculation algorithms. In all graphs, the abscissa axis indicates the quantities of neighbors ("K", from 1 to 7) specified for the IBk runs, always in odd numbers. The first pair of graphs corresponds to runs with all 20 columns; the middle pair, to the executions in which the age of the patients was excluded; the latter, to the executions in which the regression was made

only with features coming from the processing. Again, in any case, the last column of the data table provided to the algorithm is that of the variable to be regressed ("degree of murmur").

In the figure, it is clear that any configuration of IBk has performance superior to that of the base of comparison — that is, all have higher correlation coefficient and lower MAE. Moreover, at least while all 20 attributes were supplied to the regression algorithm and the number of neighbors was limited to 7, the MAE of all IBk variations remained below 0.0819 degrees (more than ten times less than the 0.8799 degrees of the base of comparison). This means that while the default `"functions.LinearRegression"` settings are expected to cause mostly erroneous degree of murmur estimates, any IBk configuration can be used to produce estimates (floating-point) that can be rounded to the correct (integer) degree in most cases.

However, as columns are being removed from the input of the algorithms, the performance of the studied variations of the IBk deteriorates (even exceeding the baseline algorithm by a large margin). In fact, it is found in the third pair of graphs that the IBk configuration with Chebyshev distance and $K = 7$ is the first to result in MAE of more than 0.5 degrees — in practice, indicating that most of the estimates can no longer be accurately rounded to the actual degree of murmur.

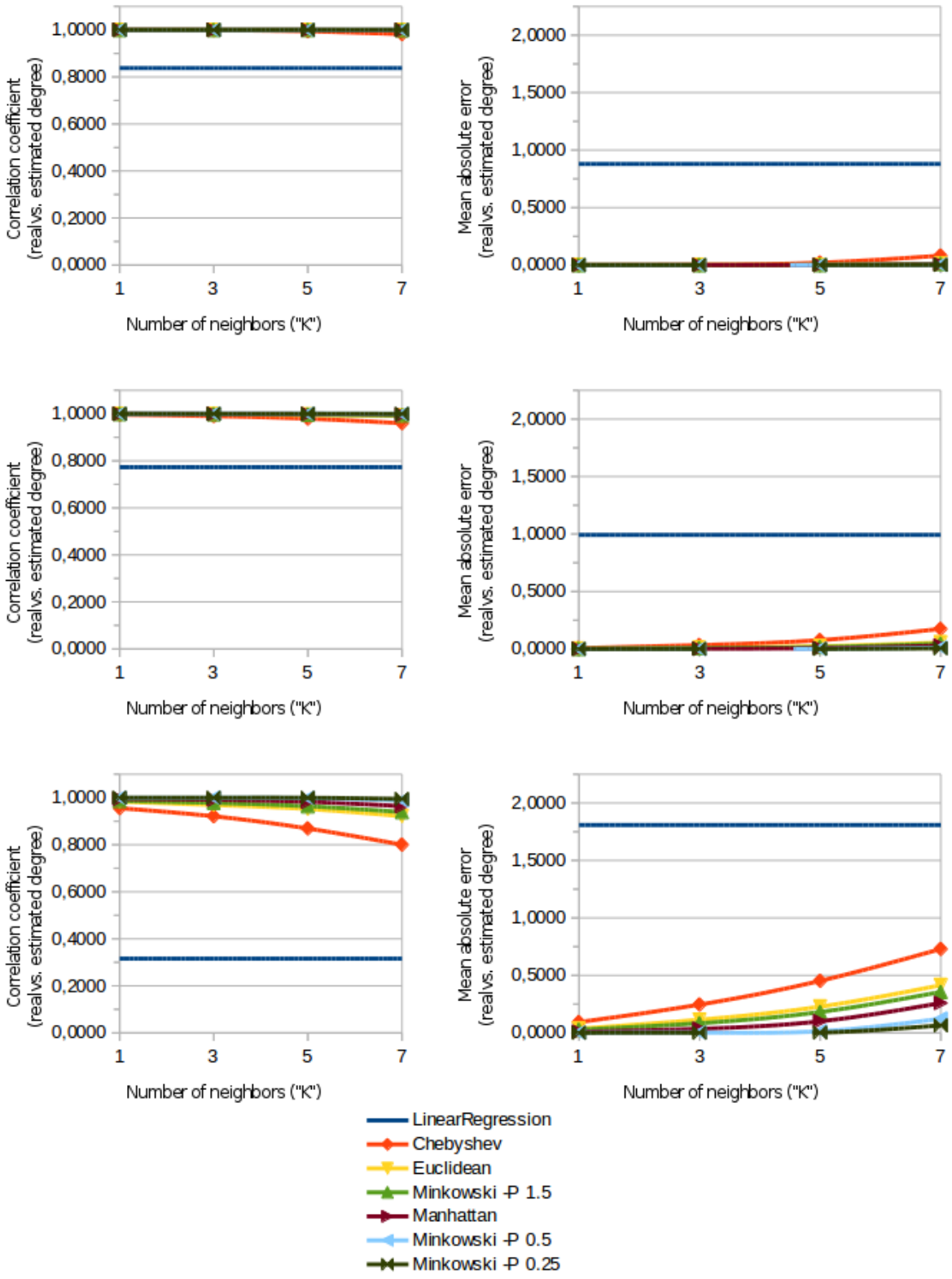
The last observation resulting from Figure 18 concerns the higher efficiency of the IBk with Minkowski distance in the tests performed. The formula of the Minkowski distance is given by Equation (18). In it, it is noted that P is a parameter represented by a real scalar. When $P = 2$, it is easy to conclude that the Minkowski distance equals Euclidean distance; when $P = 1$, the Minkowski distance equals the Manhattan distance. In particular, the graphs seem to indicate that the error is reduced as the parameter P of the Minkowski distance tends to zero.

$$d_{MINKOWSKI}(v_A, v_B) = \sqrt[P]{\sum_{d=0}^{D-1} |v_A[d] - v_B[d]|^P} \tag{18}$$

Figure 19 shows the correlation coefficient and MAE for IBk runs with Minkowski distance, with P ranging from 1 (Manhattan distance) to 0.03125. In the specific case of this figure, all executions had access the complete data table (20 columns). The number of neighbors was alternated from 1 to 15, in odd numbers. The scale was adjusted to allow differentiation between the series, which were extremely close to each other. As can be seen, with the complete table, the performance of IBk with Minkowski distance was better with $P = 0.25$ both in terms of correlation coefficient and MAE — respectively, 0.9980 and 0.0204 degrees for 15 neighbors (the worst scenario studied). The Minkowski distance with lower values ($P = 0.125$, $P = 0.0625$ and $P = 0.03125$) had intermediate performance (worse correlations than those obtained for $P = 0.5$ and $P = 0.25$ but greater than that for Manhattan distance, and mean errors similar to those observed for all executions except Manhattan distance).

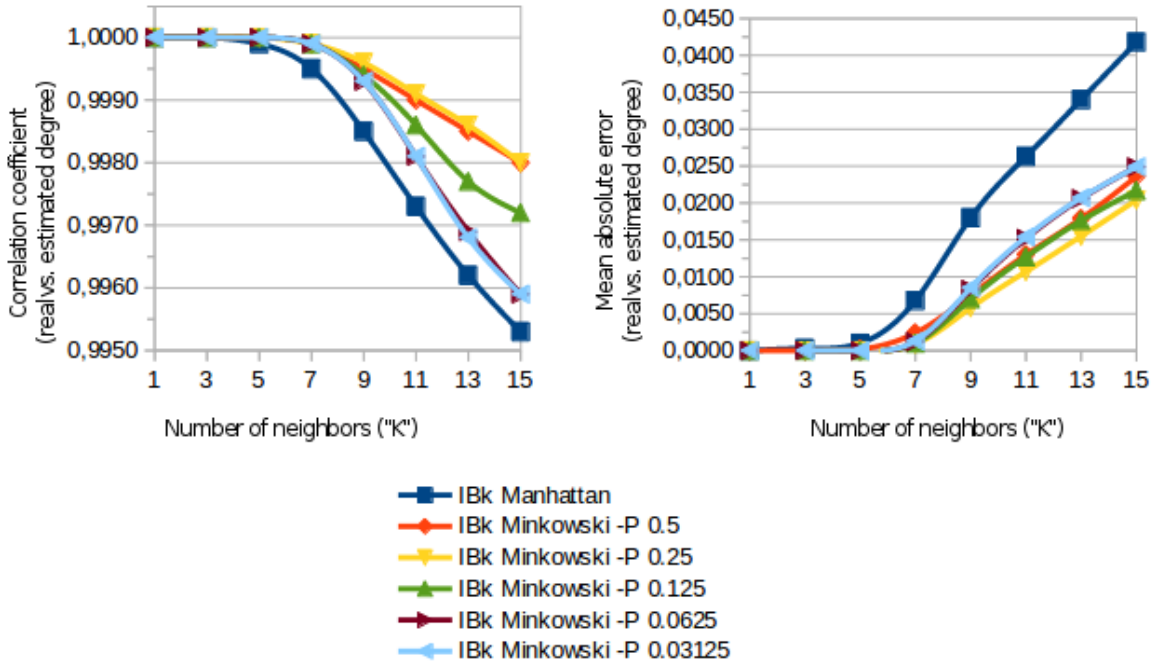
The experiment was then repeated without using the patient's age. The result is shown

Figure 18 – IBk correlation and error for 20, 19 and 16 columns according to the number of neighbors



Source: Own authorship

Figure 19 – Performance of IBk with 20 attributes and Minkowski distance of varied P



Source: Own authorship

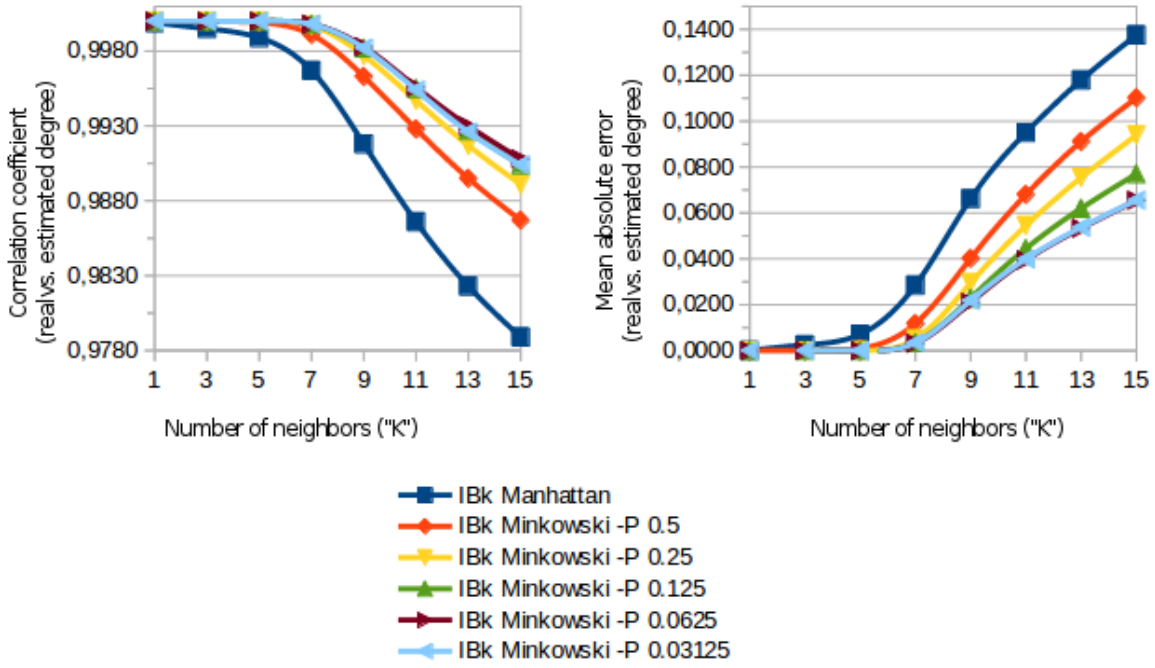
in Figure 20. In it, the performance with $P = 0.25$ was always intermediate, and with lower P values, higher correlation coefficient and lower error were obtained. With 19 columns, the best configuration of the Minkowski distance was achieved with $P = 0.0625$, with correlation coefficient equal to 0.9908 and MAE of 0.0655 degrees of murmur in the worst studied scenario.

Finally, Figure 21 shows the performance of the same IBk configurations with Minkowski distance, but now in the case of the removal of the 4 clinical attributes. In this scenario, as already commented, the regression is based only on the features generated through the processing of the WAV records and their associated metadata. Here, the performance deterioration of the configurations with $P = 0.0625$ and $P = 0.03125$ is noticeable in relation to that of the other configurations as the number of neighbors exceeds the mark of 9. With this, once again $P = 0.25$ is the adjustment with the highest correlation coefficient (0.8899) and lower MAE (0.7477 degrees), in the worst case scenario.

With these experiments, it was verified that the most consistent regression algorithm for the studied data set was IBk's, configured with Minkowski distance and $P = 0.25$.

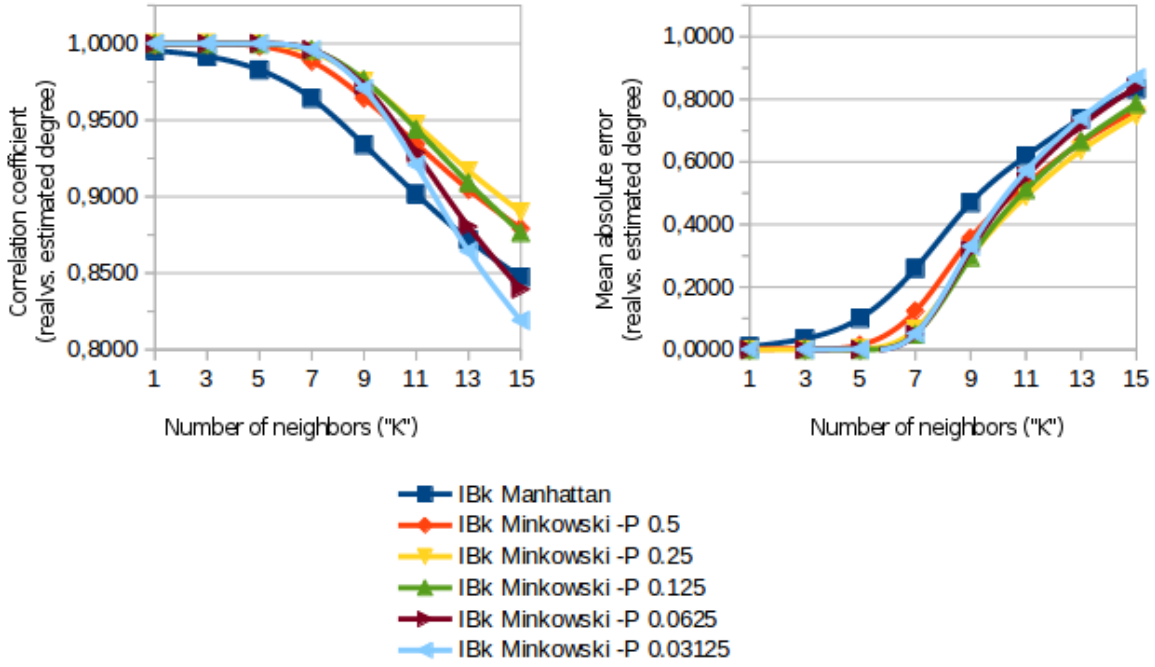
Thus, to make a final evaluation of the quality of the features generated and to also extract the best configuration for the parameter "K" (number of neighbors), the IBk with distance Minkowski and $P = 0.25$ was selected for a more detailed evaluation on Weka. By **"best configuration"** of this parameter, one can understand the maximum number of neighbors that allow the error to remain smaller than a value considered to be low — since this would mean that the generalization of the regression would be the best possible within

Figure 20 – Performance of IBk with 19 attributes and Minkowski distance of varied P



Source: Own authorship

Figure 21 – Performance of IBk with 16 attributes and Minkowski distance of varied P



Source: Own authorship

the acceptable limit. The error limit, in turn, did not have an *a priori* definition, even because the indicators provided by the Weka (correlation coefficient, MAE and some others) only have statistical significance if the studied variable (in this case, the error between the estimated degree of murmur and the actual degree) follow a normal distribution. As will be shown below, the error distribution was not normal, and therefore the error limit setting is not as straightforward as simply choosing a maximum MAE value, for example. It is enough to note that in terms of MAE, the effect of 12 errors by half-degree of murmur (all small errors) is the same as of that of a single error by 6 degrees of murmur (an unacceptable error).

Thus, the degrees of murmur were once again estimated for 20, 19 and 16 attributes from the regressions with "K" of 5, 7 and 9. Then, the estimates produced were individually analyzed. The absolute difference between each estimated value and the corresponding actual degree was determined. The mean of these errors in the first analysis has already been illustrated in Figures 19, 20 and 21. The values obtained in the reanalysis are summarized in Table 5.

Table 5 – Mean absolute error for selected IBk settings.

	5 neighbors	**7 neighbors**	**9 neighbors**
20 attributes	0.0000	0.0007	0.0057
19 attributes	0.0000	0.0039	0.0298
16 attributes	0.0010	0.0545	0.2893

Source: Own authorship

The next step was the counting of absolute errors greater than 0.5, 1.5 and 2.5 degrees. The explanation is that, for each occurrence of an absolute error greater than 0.5 degrees and less than 1.5, in the case of rounding to the nearest whole degree — for example, if a veterinarian attempts to interpret the measurement based on the original scale presented in Section 1.1 — the resulting degree will be increased or reduced by 1 in relation to the actual patient's degree of murmur. Between 1.5 and 2.5 degrees, the error will be of 2 degrees after rounding (in practice, a high difference). Estimation differences greater than 2.5 degrees will result in 3 or more degrees of difference (a completely misleading classification that should be avoided to the maximum extent).

Table 6 lists the level of error of the estimated grade after rounding. As can be seen, the total number of errors with 9 neighbors when using only the features generated via processing sum almost 30% considering all levels. 2% of the total classification errors is of 2 or more degrees of murmur — that is, extremely significant errors.

With 7 neighbors, on the other hand, the total number of post-rounding errors reduces to around 6%, with only about 0.2% of the errors reaching 2 degrees of difference. It is worth remembering, however, that this would be the case only if all the clinical attributes were neglected (that is, the most difficult classification scenario). With the clinical parameters considered (20 attributes), only 0.05% of classification error was calculated for the studied

Table 6 – Error level after rounding for the selected IBk settings.

	Error level	**5 neighbors**	**7 neighbors**	**9 neighbors**
20 attributes	**1 degree**	0.0000%	0.0496%	0.4712%
	2 degrees	0.0000%	0.0000%	0.0248%
	3+ degrees	0.0000%	0.0000%	0.0000%
19 attributes	**1 degree**	0.0000%	0.3472%	2.5298%
	2 degrees	0.0000%	0.0000%	0.1488%
	3+ degrees	0.0000%	0.0000%	0.0000%
16 attributes	**1 degree**	0.0992%	5.9772%	27.8274%
	2 degrees	0.0000%	0.1984%	1.8353%
	3+ degrees	0.0000%	0.0000%	0.1240%

Source: Own authorship

patients — once again remembering that the "age" column presented a sampling phenomenon that may be able to distort the results. Thus, by eliminating this column, a number of errors is achieved of approximately 0.35% of the estimates after rounding (all of these errors being of only 1 degree of murmur).

Table 7 lists all absolute estimation errors greater than 0.5 degrees for 19 columns, $K = 7$ and $P = 0.25$. From the table, it is easy to see the undesirable effect on the rounding — because in all the lines displayed, the rounding of the estimated degree would move it away from the actual one by exactly 1 degree (in the specific case of the mentioned configuration). It is also noted that the 14 post-rounding errors occur with 8 different patients; patients 25, 6 and 7 are the most impaired, with 2, 3 and 4 of these errors, respectively. However, since all patients are associated with 72 lines (each) in the present study, it follows that techniques such as the averaging of all estimates obtained for the same patient would have the effect of overriding the observed errors. Therefore, this configuration — IBk with 7 neighbors and Minkowski distance with $P = 0.25$ — was considered the most indicated for the studied dataset.

For comparison purposes, Table 8 lists some of the absolute estimation errors of less than 0.5 degrees. As one can see, after rounding the estimated degree, the actual degree is obtained. Similar results were obtained for 4018 of the 4032 lines of the present study.

Figure 22 shows the diagram of the estimated variable (estimated degree of murmur) by the real variable (actual degree of murmur) as generated by Weka. The vast majority of points are coincident, that is, the estimated degree is exactly the actual one; this type of point can not be easily viewed unless a small jitter (noisy displacement) is applied, as shown in Figure 23.

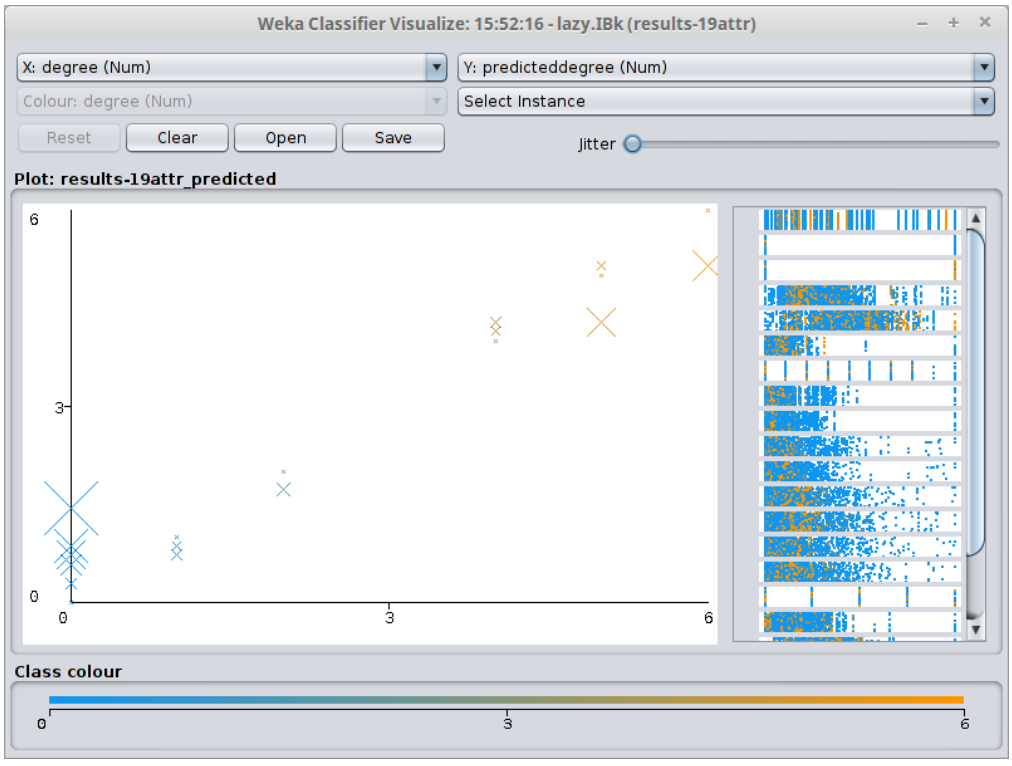
A particularly revealing way of observing errors after rounding is through Table 9, which displays the confusion matrix for the classifications using the chosen settings. As one can see, in only 5 of the 14 errors the estimated degree of murmur is less than the actual one. In the other 9, the estimated degree is greater than the actual one. These 9 instances are also

Table 7 – Absolute error per patient for regression with IBk of $P = 0.25$ and $K = 7$: all errors after rounding.

Patient	Estimated degree	Actual degree	Absolute error
30	0.57	0	0.57
25	0.71	0	0.71
	0.71	0	0.71
24	0.71	0	0.71
6	0.71	0	0.71
	0.71	0	0.71
	1.43	0	1.43
1	0.86	0	0.86
17	0.86	0	0.86
36	4.29	5	0.71
7	5.14	6	0.86
	5.14	6	0.86
	5.14	6	0.86
	5.14	6	0.86

Source: Own authorship

Figure 22 – Chart generated by Weka correlating the estimated degree with the actual one



Source: Own authorship

Table 8 – Absolute error per patient for regression with IBk of $P = 0.25$ and $K = 7$: some of the accurate classifications after rounding.

Patient	Estimated degree	Actual degree	Absolute error
10	0	0	0.00
12	0	0	0.00
16	0	0	0.00
17	0	0	0.00
19	0	0	0.00
27	0	0	0.00
34	0	0	0.00
35	0	0	0.00
38	0	0	0.00
40	0	0	0.00
43	0	0	0.00
44	0	0	0.00
47	0	0	0.00
49	0	0	0.00
50	0	0	0.00
54	0	0	0.00
23	0.857143	1	0.14
9	5.142857	5	0.14
18	5.142857	5	0.14
5	4.285714	4	0.29
13	0.285714	0	0.29
39	0.285714	0	0.29
21	1.714286	2	0.29

Source: Own authorship

called **false positives**, that is, patients for whom the proposed technique accuses murmur when, in reality, there are none.

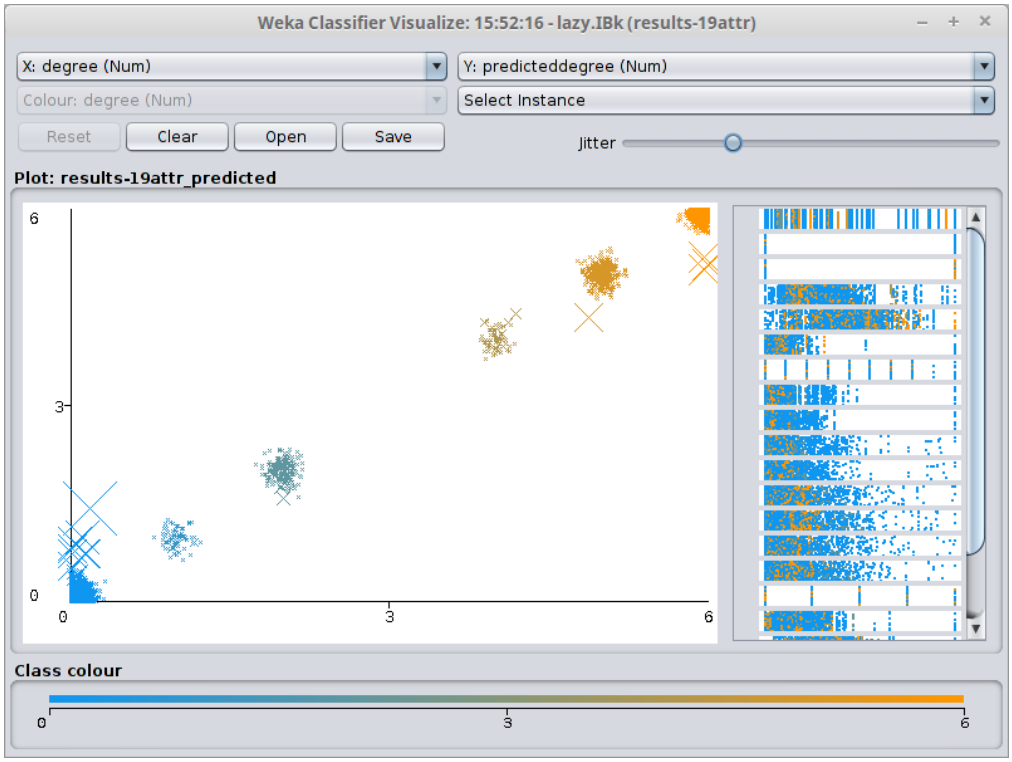
Table 9 – Confusion matrix after rounding.

Estimate / Actual	**0**	**1**	**2**	**3**	**4**	**5**	**6**
0	2799	9	-	-	-	-	-
1	-	72	-	-	-	-	-
2	-	-	216	-	-	-	-
3	-	-	-	-	-	-	-
4	-	-	-	-	72	-	-
5	-	-	-	-	1	359	-
6	-	-	-	-	-	4	500

Source: Own authorship

Finally, Figure 24 is used to demonstrate that there is no greater number of neighbors

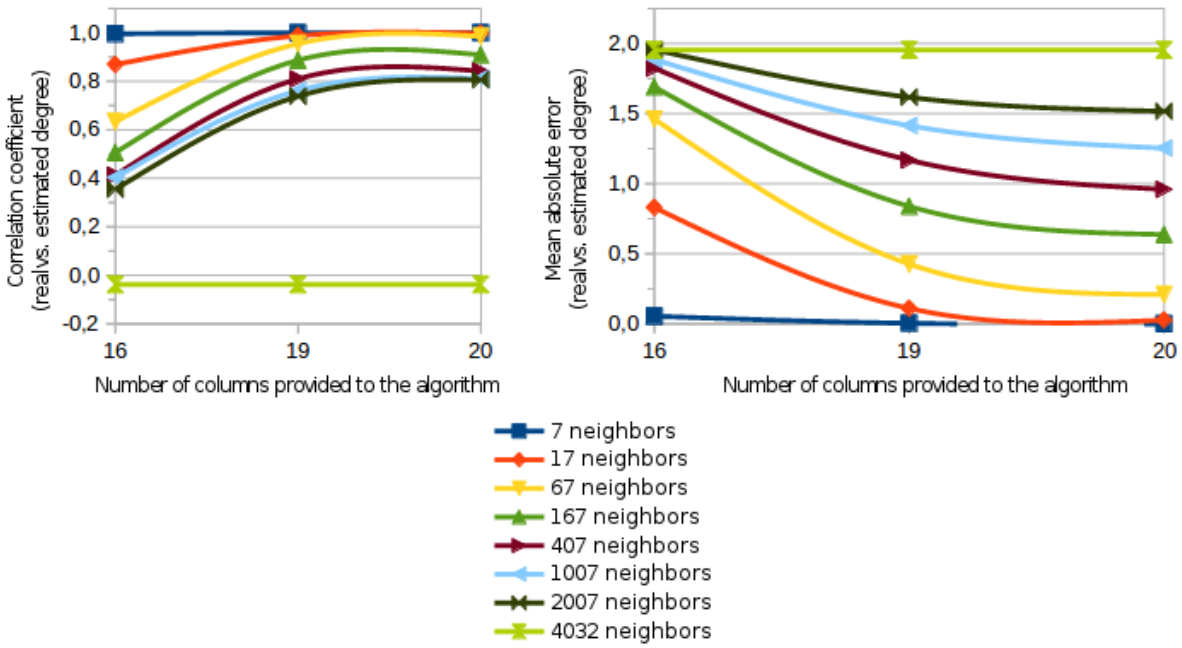
Figure 23 – Chart generated by Weka correlating the estimated degree with the actual one (with jitter)



Source: Own authorship

than the one chosen that can lead to better performance. In fact, taking the reasoning to the limit, it is easy to conclude that the error assumes a very high value when "K" is maximum (4032, the very amount of records in the training set), since it means that, regardless of the new line being evaluated, the estimate of the degree of murmur will simply equal the arithmetic mean of the degrees of all known instances. This case is illustrated in the figure; it is the lowest correlation coefficient (-0.0378) and the highest mean absolute error (1.9544) regardless of the number of columns analyzed — exactly because only one column influences the estimate: the "degree of murmur" column. Although, in a certain point of view, the generalization of the model is the maximum in this case (because it will exhibit exactly the same performance pattern during and after training), on the other hand it is proven that this performance is quite poor. This means that the criterion adopted when choosing the best configuration, the one of the largest possible amount of neighbors while keeping a margin of error considered acceptable, is correct. With the chosen configuration, the model remains as generic as possible without causing too much misclassification.

Figure 24 – IBk performance with 19 attributes and Minkowski distance of $P = 0.25$



Source: Own authorship

4 CONCLUSIONS

The present study presented the problem of the subjective measurement of heart murmurs from the limitations of the stethoscope and the difficulties of training medical professionals for proficient auscultation. However, it highlighted the importance of the technique, especially in veterinary medicine and in underdeveloped countries. As a result, the document proposed automatic classification of canine murmur on the existing scale — extrapolating it in a way that a value 0 denoted the absence of murmur — from a record made with an electronic stethoscope.

From the set of 56 audio recordings and their metadata, outcomes of the research of Andrade (2018), 16632 values of 15 different features were obtained. These values were combined with the clinical attributes of the patients (age, weight, sex, presence of clinical signs and degree of murmur) in a table of 20 columns (the 5 clinical attributes and 15 features) and 4032 lines — 72 lines for each patient. The analysis of this table on the Weka software allowed the finding of an algorithm capable of performing regression to the degree of murmur from the 19 other attributes. The IBk classifier obtained better results for the dataset in the configuration of 7 neighbors and Minkowski distance of constant $P = 0.25$; however, several other combinations are possible, so fine tuning can be accurately performed for new (and potentially larger) training sets.

Two main performance indicators were evaluated during the choice of the regression algorithm: the correlation coefficient between the produced estimate and the actual degree of murmur, and the mean absolute error of these estimates. In the chosen configuration, the IBk classifier obtained a correlation coefficient of 0.9999 when using all columns, 0.9999 when excluding the column "age" (which presented a phenomenon potentially detrimental to the generalization of the regression), and 0.9946 when using only the features produced in the present work to estimate the degree of murmur. The mean absolute error in the same scenarios was 0.0009, 0.0051 and 0.0647, respectively. The configuration was tested again (obtaining MAE of 0.0007, 0.0039 and 0.0545), and the estimation errors for 19 columns were studied in detail — revealing that the 14 absolute errors greater than 0.5 degrees existing had origin in the data of 8 different patients. In this situation, which is probably the closest to the desired use for the features and clinical attributes studied, only 0.35% of the estimates can not be associated back to the actual degree of murmur (differing by 1 entire degree on the original scale). However, it was pointed out that because of the error profile — that is, the low number of high error estimates per patient — future improvements as the average between estimates can be applied in a way to further reduce the error (or to maintain it at acceptable levels after increasing the generalization capacity, for example through the K parameter).

The chosen configuration, in the worst simulated case (in which all clinical attributes, except degree of murmur, are removed), obtained post-rounding estimation error of around

6% for 1 degree of murmur and of about 0.2% for 2 degrees. With this, it can be affirmed that the set of features produced has great impact on the regression, since even without the support of clinical attributes they could provide reasonably reliable estimates. Classification errors even smaller than these are expected in future studies with a larger set of training data.

Finally, the proposed automatic classification scale may be able to make the classification of the degree of murmur more objective, as it makes use of market equipment and easily reproducible algorithms. It bears similar resemblance to the existing scale, familiar to veterinary clinicians, and has potential for use in training. In addition, the estimates generated by it should not be influenced by factors such as time passing, professional in charge or, to some degree, physical attributes of the consulting room. Moreover, future studies can make the scale useful even in continuous mode — that is, without having to relate the reading of the estimates to the discrete degrees of the original scale. Finally, it should be noted that only free software, mostly open source, was used in the research, so that as many barriers as possible were removed for future work.

For all this, it is evaluated that the objective of the work has been reached, and that the perspectives for the technology developed are promising.

4.1 FUTURE WORKS

Several aspects of the work presented in this document can be improved.

The research made use of records and metadata previously manually processed by a veterinarian. Some steps in this preprocessing can be automated quite simply, such as mono-mode capture, since neither the recording equipment nor the feature-generating algorithms have support for 2 or more channels. Audio normalization is also a step that does not require human interference. On the other hand, marking of heart sounds is the most sensitive stage of all preprocessing, and may only be automated with the use of advanced algorithms described in the literature. Finally, the patient's input of weight, age, sex, and clinical signs is not susceptible to automation.

In the field of features, several others can be searched and the generalization of extracted features should be analyzed with more patients. All aspects of the regression, including parameters and the regression algorithms themselves, can be studied to obtain less error inside and outside the training set. As already commented, a continuous scale can be explored from the configuration suggested here and/or through unpublished techniques. Finally, the shipment of these technologies directly in the hardware can be probed, mainly for the reduction in the execution time of the applied algorithms.

Bibliography

ABDENNUR, N. **NumPy binary file parser for javascript**. 2016. Disponível em: <https://gist.github.com/nvictus/88b3b5bfe587d32ac1ab519fd0009607>. Acesso em: 10 de fevereiro de 2019. Cited on page 42.

ANDRADE, C. C. A. N. de. **Análise de sons cardíacos de cães de pequeno porte através de estetoscópio eletrônico**. 2018. 73 f. Dissertação (Dissertação (Mestrado) – Programa de Pós-Graduação em Engenharia Biomédica) — Universidade Tecnológica Federal do Paraná, Curitiba, 2018. Cited 6 times on pages 19, 20, 23, 24, 39, and 66.

Audacity Team. **Audacity**. 2019. Disponível em: <https://www.audacityteam.org/>. Acesso em: 05 de março de 2019. Cited on page 22.

BARMA, S. et al. A novel feature generation method based on nonlinear signal decomposition for automatic heart sound monitoring. In: **2014 International Conference on Orange Technologies**. [S.l.: s.n.], 2014. p. 201–204. Cited on page 30.

BOUCKAERT, R. R. et al. **WEKA Manual for Version 3-8-3"**. 2018. Disponível em: <https://www.cs.waikato.ac.nz/ml/weka/documentation.html>. Acesso em: 23 de março de 2019. Cited on page 33.

BOUTANA, D.; BENIDIR, M.; BARKAT, B. Segmentation and time-frequency analysis of pathological heart sound signals using the emd method. In: **2014 22nd European Signal Processing Conference (EUSIPCO)**. [S.l.: s.n.], 2014. p. 1437–1441. ISSN 2076-1465. Cited 2 times on pages 38 and 43.

FEITOSA, F. L. F. **Semiologia Veterinária: a Arte do Diagnóstico — Cães, Gatos, Eqüinos**. [S.l.]: ROCA, 2008. ISBN 9788572417525. Cited 3 times on pages 19, 20, and 21.

FELDMAN, M. Time-varying vibration decomposition and analysis based on the hilbert transform. **Journal of Sound and Vibration**, v. 295, n. 3, p. 518 – 530, 2006. ISSN 0022-460X. Cited on page 30.

FELDMAN, M. **Hilbert Vibration Decomposition (HVD) | Adaptive Separation of Mixed Nonstationary Signals.** 2019. Disponível em: <http://hitech.technion.ac.il/feldman/HVD.html>. Acesso em: 10 de fevereiro de 2019. Cited on page 42.

FRANK, E.; HALL, M. A.; WITTEN, I. H. **The WEKA Workbench. Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques"**. 2016. Disponível em: <https://www.cs.waikato.ac.nz/ml/weka/>. Acesso em: 10 de fevereiro de 2019. Cited on page 51.

GOLLAPUDI, S.; LAXMIKANTH, V. **Practical Machine Learning.** Packt Publishing, 2016. (Community Experience Distilled). ISBN 9781784399689. Disponível em: <http://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=1163839&lang=pt-br&site=ehost-live>. Cited 5 times on pages 30, 32, 33, 34, and 55.

GULP. **gulp / LICENSE**. 2019. Disponível em: <https://github.com/gulpjs/gulp/blob/master/LICENSE>. Acesso em: 10 de fevereiro de 2019. Cited on page 36.

GULP. **gulp.js - The streaming build system**. 2019. Disponível em: <https://gulpjs.com/>. Acesso em: 10 de fevereiro de 2019. Cited on page 36.

HAYKIN, S.; VEEN, B. V. **Sinais E Sistemas**. [S.l.]: Bookman, 2001. ISBN 9788573077414. Cited 3 times on pages 27, 28, and 29.

HUANG, N. et al. The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis. **Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences**, v. 454, p. 903–995, 03 1998. Cited 2 times on pages 29 and 30.

KALUŽA, B. **Machine Learning in Java.** Packt Publishing, 2016. (Community Experience Distilled). ISBN 9781784396589. Disponível em: <http://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=1230627&lang=pt-br&site=ehost-live>. Cited 4 times on pages 30, 33, 34, and 35.

KIERTSCHER, T. **gulp-text-simple - npm**. 2018. Disponível em: <https://www.npmjs.com/package/gulp-text-simple>. Acesso em: 10 de fevereiro de 2019. Cited on page 37.

KUMAR, K.; THOMPSON, W. R. Evaluation of cardiac auscultation skills in pediatric residents. **Clinical pediatrics**, v. 52, 11 2012. Cited 3 times on pages 19, 22, and 27.

LASZUK, D. **Python implementation of Empirical Mode Decompoisition (EMD) method**. 2019. Disponível em: <https://github.com/laszukdawid/PyEMD>. Acesso em: 10 de fevereiro de 2019. Cited on page 41.

LENG, S. et al. The electronic stethoscope. **BioMedical Engineering OnLine**, v. 14, n. 1, p. 66, Jul 2015. ISSN 1475-925X. Cited 5 times on pages 19, 20, 27, 28, and 30.

MESQUITA, C. T. et al. Estetoscópio digital como ferramenta inovadora no ensino da ausculta cardíaca. **Arq Bras Cardiol**, v. 100, n. 2, p. 187–189, 2013. Cited on page 22.

NARVÁEZ, P. et al. Classification of heart sounds using linear prediction coefficients and mel-frequency cepstral coefficients as acoustic features. In: **2017 IEEE Colombian Conference on Communications and Computing (COLCOM)**. [S.l.: s.n.], 2017. p. 1–6. Cited on page 19.

NASSRALLA, M.; ZEIN, Z. E.; HAJJ, H. Classification of normal and abnormal heart sounds. In: **2017 Fourth International Conference on Advances in Biomedical Engineering (ICABME)**. [S.l.: s.n.], 2017. p. 1–4. ISSN 2377-5696. Cited on page 19.

NPM, Inc. **pytalk - npm**. 2016. Disponível em: <https://www.npmjs.com/package/pytalk>. Acesso em: 10 de fevereiro de 2019. Cited on page 41.

NPM, Inc. **npm**. 2019. Disponível em: <https://www.npmjs.com/>. Acesso em: 10 de fevereiro de 2019. Cited on page 36.

PAZIN-FILHO, A.; SCHMIDT, A.; MACIEL, B. Ausculta cardíaca: Bases fisiológicas - fisiopatológicas. **Medicina (Ribeirao Preto. Online)**, v. 37, n. 3/4, p. 208–226, dez. 2004. Cited 4 times on pages 19, 20, 21, and 27.

REECE, W. O. et al. **Dukes | Fisiologia dos animais domésticos**. [S.l.]: Editora Guanabara Koogan, 2018. ISBN 9788527731256. Cited on page 19.

SCHAUB, T. **gulp-newer - npm**. 2019. Disponível em: <https://www.npmjs.com/package/gulp-newer>. Acesso em: 10 de fevereiro de 2019. Cited on page 37.

The SciPy community. **numpy.lib.format**. 2019. Disponível em: <https://www.numpy.org/devdocs/reference/generated/numpy.lib.format.html>. Acesso em: 10 de fevereiro de 2019. Cited on page 42.

TILKIAN, M. B. C. A. G. **Entendendo os Sons e Sopros Cardíacos: com introdução aos sons pulmonares**. [S.l.]: ROCA, 2004. ISBN 8572414843. Cited 2 times on pages 19 and 20.

VIKHE, P. S.; HAMDE, S. T.; NEHE, N. S. Wavelet transform based abnormality analysis of heart sound. In: **2009 International Conference on Advances in Computing, Control, and Telecommunication Technologies**. [S.l.: s.n.], 2009. p. 367–371. Cited on page 30.

WANG, J. et al. The design of a heart sound teletransmission system. In: **2007 IEEE/ICME International Conference on Complex Medical Engineering**. [S.l.: s.n.], 2007. p. 274–277. Cited on page 27.

WEBSTER, J. **Medical Instrumentation: Application And Design, 3Rd Ed**. [S.l.]: Wiley India Pvt. Limited, 2009. ISBN 9788126511068. Cited 2 times on pages 27 and 28.

WICKERT, M. **Signals and Systems For Dummies**. [S.l.]: Wiley, 2013. (–For dummies). ISBN 9781118475669. Cited on page 29.

Annex

ANNEX A – CEUA-UTFPR (ethics committee) approval

Ministério da Educação
UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
Câmpus Dois Vizinhos
Comissão de Ética no Uso de Animais - CEUA

PROJETO DE PESQUISA / AULA PRÁTICA

Título:	Protocolo de ausculta cardíaca em cães com uso do estetoscópio eletrônico
Área Temática:	Processamento de sinais biológicos
Pesquisador / Professor:	Sergio Leandro Stebel
Instituição:	UTFPR – Curitiba (DAELN-CT)
Financiamento:	Não há
Versão:	02

PARECER CONSUBSTANCIADO DA CEUA	**Protocolo nº 2017-006**

Apresentação do Projeto:

O projeto em apreciação visa elaborar um possível protoco de ausculta cardíaca em cães fazendo uso de equipamento eletrônico em experimentação, em comparação a técnicas de uso corrente.

Objetivo:
De acordo com os autores do projeto, a proposta deste estudo é realizar o acompanhamento clínico da evolução dos sons cardíacos em cães, através do registro gráfico, com auxílio da fonocardiografia na Clínica Veterinária 'Dog Vet' localizada na cidade de Curitiba. Mais especificamente pretendem com este experimento:

1. Avaliar os tipos de sopros cardíacos;
2. Estudar a ocorrência de disfunção diastólica e/ou sistólica, e
3. Avaliar o ritmo cardíaco com uso do fonocardiógrafo.

Avaliação dos Riscos e Benefícios:

Os riscos esperados pelos proponentes do projeto são os típicos vinculados ao uso inicial de um novo equipamento: as dificuldades iniciais quanto aos mecanismos e condições corretas de uso e também quanto ao adequado *design* do equipamento. No que tange a segurança imediata em seu uso, o equipamento que está sendo desenvolvido não apresenta riscos elétricos de choque ou descargas, seja no paciente ou no usuário do equipamento.

Quanto aos benefícios, os proponentes apontaram três aspectos distintos. Primeiramente, o estetoscópio eletrônico presumivelmente deve representar menos estresse ao paciente, uma vez que não requer imobilização, nem raspagem dos pelos do abdome, nem mesmo a postura obrigatória de decúbito, como no exame de ecocardiografia. Em segundo lugar, caso a ausculta eletrônica mostre-se eficaz, tornar-se-á um recurso significativamente mais barato que o ecocardiograma para o médico veterinário. Por fim, é esperado que a ausculta eletrônica seja mais eficiente no diagnóstico de doenças cardíacas em animais em suas fases iniciais.

Comentários e Considerações sobre a Pesquisa:

Os itens descritos nos documentos apresentados até então são compatíveis com os objetivos da proposta e a metodologia foi mais pormenorizada por parte dos proponentes.

Considerações sobre os Termos de apresentação obrigatória:

Foram anexados à presente solicitação apenas os seguintes termos e documentos: 1) o formulário unificado de

Ministério da Educação
UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
Câmpus Dois Vizinhos
Comissão de Ética no Uso de Animais - CEUA

encaminhamento ao CEUA-UTFPR, assinado pelo pesquisador responsável pelo projeto; 2) projeto de pesquisa completo no modelo da PROPPG-CEUA; 3) a declaração de não início do projeto (com assinatura e data do pesquisador responsável); 4) Declaração de médico veterinário, datado e assinado, como responsável técnico pelo bem estar dos animais utilizados durante a execução do projeto; 5) requerimento de solicitação para avaliação do projeto pela CEUA-UTFPR; 6) documento de anuência da atual Coordenadora (Profª. Drª. Leandra Ulbricht) do Programa de Pós-Graduação em Engenharia Biomédica e que inclui carimbo e assinatura de ciência e concordância do Diretor de Pesquisa e Pós-Graduação do Câmpus de Curitiba, Prof. Dr. Alexandre A. P. Pohl.
Conclusões ou Pendências e Lista de Inadequações: **Não há.**
Situação do Parecer: APROVADO
Considerações Finais a Critério da CEUA: Todos os procedimentos devem seguir a lei n° 11.794 de 8 de outubro de 2008.

CERTIFICADO

Certificamos que o projeto intitulado "Protocolo de ausculta cardíaca em cães com uso do estetoscópio eletrônico", protocolo nº 2017/006, sob a responsabilidade de Sergio Leandro Stebel - que envolve a produção, manutenção e/ou utilização de animais pertencentes ao filo Chordata, subfilo Vertebrata (exceto o homem), para fins de pesquisa científica (ou ensino) - encontra-se de acordo com os preceitos da Lei nº 11.794, de 8 de outubro de 2008, do Decreto nº 6.899, de 15 de julho de 2009, e com as normas editadas pelo Conselho Nacional de Controle da Experimentação Animal (CONCEA), e foi aprovado pela COMISSÃO DE ÉTICA NO USO DE ANIMAIS (CEUA-UTFPR) da UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ, em reunião de 04/07/2017.

Vigência do projeto:	**05 de julho de 2017 a dezembro de 2017**
Finalidade	( ) Ensino (X) Pesquisa Científica
Espécie/linhagem:	*Canis lupus familiaris* (sem distinção de raça)
Número de animais:	100
Peso/Idade:	Sem distinção
Sexo:	Sem distinção
Origem:	Proprietários particulares

Dois Vizinhos, 5 de julho de 2017.

Assinado por:
Nédia de Castilhos Ghisi
Presidente da Comissão de Ética no Uso de Animais da Universidade Tecnológica Federal do Paraná

Nédia de Castilhos Ghisi
Presidente do CEUA - UTFPR
Comissão de Ética no uso de Animais