PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA

DESIGN OF A PREPROCESSING SYSTEM FOR SOUNDS OBTAINED FROM CHEST AUSCULTATION

CHRISTIAN MATÍAS ESCOBAR ARCE

Thesis submitted to the Office of Research and Graduate Studies
in partial fulfillment of the requirements for the degree of
Master of Science in Engineering

Advisor:
PATRICIO DE LA CUADRA

Santiago de Chile, July 2021

PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA

DESIGN OF A PREPROCESSING SYSTEM FOR SOUNDS OBTAINED FROM CHEST AUSCULTATION

CHRISTIAN MATÍAS ESCOBAR ARCE

Members of the Committee:
PATRICIO DE LA CUADRA
DOMINGO MERY
GONZALO ACUÑA
HERNÁN DE SOLMINIHAC

Thesis submitted to the Office of Research and Graduate Studies
in partial fulfillment of the requirements for the degree of
Master of Science in Engineering

Santiago de Chile, July 2021

Dedicado a mis padres,
abuelos, hermano, amigos y
Anita, mi compañera de vida.

ABSTRACT

Auscultation is a simple, inexpensive, and non-invasive process for diagnosing respiratory diseases. However, its main difficulties are that these sounds are concentrated in a very low frequency band, they overlap with heart sounds and their interpretation is subject to the experience of the physician. With the advancement of technology, digital stethoscopes have been created that allow these sounds to be recorded for analysis.

In this work, the design of a system based on two stages is proposed. In the first stage, the detection of the fundamental heart sounds is performed using a CNN with encoder-decoder architecture using a database of 792 phonocardiograms from 135 different patients (from which 54 of these patients present some heart pathology). At this stage, a high performance is reported (close to 93±1.1% for different metrics such as accuracy, recall, precision and F_1 score) from a k-fold cross-validation with $k = 10$. In addition, it is shown to be a robust system that presents high performance regardless of the input features of the network.

In the second stage, different methods of source separation are proposed through a Non-negative Matrix Factorization (NMF) decomposition process. Among these, the NMF method on the entire signal and replacing segments in the heart sound positions with the estimated lung sound, in conjunction with a assignment criterion based on heart sound position correlation, give the best results in terms of reconstruction of the lung sound ($\geq 0.96 \pm 0.01$ in temporal and spectral correlation, 0.001 ± 0.0006 in Mean Square Error and 11.86 ± 1.59 dB in Signal to Distortion Ratio). From this process, a clean lung sound will be obtained that could be used in a classification system for respiratory diseases.

Keywords: Heart sound detection, heart sound segmentation, Convolutional Neural Networks (CNN), lung and heart sounds separation, source separation, Non-negative Matrix Factorization (NMF)

RESUMEN

La auscultación es un proceso simple, de bajo costo y no invasivo para realizar diagnósticos de enfermedades respiratorias. Sin embargo, sus principales dificultades es que estos sonidos se concentran en una banda de frecuencia muy baja, se traslapan con los sonidos cardiacos y su interpretación está sujeta a la experiencia del médico. Con el avance de la tecnología se han creado estetoscopios digitales que permiten registrar estos sonidos para analizarlos.

En este trabajo se propone el diseño de un sistema de dos etapas. En la primera etapa se realiza la detección de los sonidos cardiacos fundamentales usando una CNN con arquitectura encoder-decoder sobre una base de datos de 792 fonocardiogramas de 135 pacientes distintos (54 de estos pacientes presenta alguna patología cardiaca). En esta etapa se reporta un alto desempeño (cercano a $93\pm1.1\%$ para distintas métricas como exactitud, sensibilidad, precisión y valor F_1) a partir de una validación cruzada de $k = 10$ iteraciones. Además, se muestra que es un sistema robusto que presenta un alto desempeño independiente de las características de entrada de la red.

En la segunda etapa se proponen distintos métodos de separación de fuentes mediante un proceso de descomposición con Factorización No Negativa de Matrices (NMF). Entre los métodos propuestos, los que aplican NMF sobre la señal completa y reemplazan los segmentos en la posición del sonido cardiaco con el sonido respiratorio estimado, en conjunto con el criterio de asignación de componentes basado en la correlación temporal con las posiciones del sonido cardiaco obtienen mejores resultados en cuanto a la reconstrucción del sonido respiratorio ($\geq 0.96\pm0.01$ en correlación temporal y espectral, 0.001 ± 0.0006 en error cuadrático medio y 11.86 ± 1.59 dB en razón señal a distorsión). A partir de este proceso, se tendrá un sonido respiratorio limpio que podría ser utilizado en un sistema de clasificación de enfermedades respiratorias.

Palabras Claves: Detección de sonidos cardiacos, segmentación de sonidos cardiacos, redes convolucionales, separación de sonidos respiratorios y cardiacos, separación de fuentes, factorización no negativa de matrices

AGRADECIMIENTOS

En primer lugar, me gustaría agradecer al profesor Patricio de la Cuadra por haber sido un gran apoyo durante estos años de investigación. Se agradece la confianza depositada en mí para realizar este proceso y la libertad de desarrollar ideas propias, siempre ayudándome a encaminarlas de la mejor forma posible. Además, de permitirme ser partícipe de distintas experiencias como ayudantías y viajes de estudio que me permitieron desarrollar aún más habilidades que serán de utilidad en mi vida profesional. A pesar de estar en un ambiente muy académico y formal, se agradece su calidez y humanidad como profesor guía de este proceso.

Agradezco la amable ayuda de profesores como Gustavo Angulo y Domingo Mery al momento de solucionar dudas en áreas en los que ellos son expertos. Agradezco también al profesor Rodrigo Cádiz por permitirme entrenar modelos en su servidor, lo cual fue un aporte tremendo en el entrenamiento de redes neuronales en mi investigación.

Agradezco también a Gabriel Durán, querido compañero y amigo ya egresado de este programa que fue de mucho aporte en la presentación de algunas herramientas útiles, y de guía en aspectos administrativos. Quiero agradecer también a los distintos proyectos que marcaron mi paso por la Universidad como el Cuerpo de Tutores, Proyecta UC y la rama estudiantil IEEE, que pese a tener distintos enfoques, permiten generar un ambiente de comunidad muy deseable dentro de la Escuela. Así mismo, agradezco cada amigo/a que me envió sus buenos deseos para terminar de la mejor manera esta etapa.

Quiero agradecer también a mis padres por siempre alentarme a dar lo mejor de mí en cada proyecto o desafío que se me presenta. Me siento muy orgulloso de ser su hijo, y espero que este trabajo sea el fiel reflejo de los valores que me han entregado a lo largo de mi vida. A mis abuelos por toda su preocupación y cariño entregado frente a este desafío. Y por supuesto a mi hermano, siempre atento a cualquier cosa que necesité.

Finalmente, agradezco especialmente a Anita que vivió muy de cerca cada momento de este desafío, y que ha sido un gran soporte emocional y anímico. Espero que este sea uno de los muchos episodios que vivamos juntos en nuestras vidas. Gracias por tu amor y apoyo incondicional, que me hace sentir en cada momento una persona que puede lograrlo todo.

CONTENTS

ABSTRACT **II**

RESUMEN **III**

AGRADECIMIENTOS **IV**

LIST OF FIGURES **VIII**

LIST OF TABLES **XI**

1 Introduction and context of the thesis **1**
1.1 Auscultation as a diagnostic method 1
1.2 Lung sounds 2
1.3 Heart sounds 4
1.4 Research scope 6

2 Heart sound detection **8**
2.1 Related works 10
2.2 Theoretical background 13
2.2.1 CNN based Encoder-Decoder 13
2.3 Database 15
2.4 Implementation 18
2.4.1 Preprocessing 19
2.4.2 Feature settings 19
2.4.2.1 Homomorphic filters 20
2.4.2.2 Hilbert envelopes 20
2.4.2.3 Wavelets envelopes 22
2.4.2.4 Frequency bands energy 22
2.4.2.5 Variance fractal dimension 22
2.4.2.6 Spectral tracking 23
2.4.3 Envelopes resampling 23
2.4.4 Initial network design 25
2.4.5 Output modeling algorithm 26
2.4.6 Data augmentation 29

2.5 Experiments and results analysis . . . 29
2.5.1 Performance metrics . . . 30
2.5.2 Features analysis . . . 30
2.5.3 Number of filters analysis . . . 32
2.5.4 Filter length analysis . . . 33
2.5.5 Network depth analysis . . . 34
2.5.6 Analysis of the length and step of the training windows . . . 35
2.5.7 Number of labels analysis . . . 36
2.5.8 Network stability analysis . . . 37
2.6 Conclusions . . . 39

3 Source separation **41**
3.1 Related works . . . 43
3.2 Theoretical background . . . 45
3.2.1 Non-negative Matrix Factorization . . . 46
3.3 Databases . . . 51
3.4 Implementation . . . 52
3.4.1 Preprocessing . . . 53
3.4.2 Decomposition methods using NMF . . . 53
3.4.2.1 NMF on entire signal . . . 53
3.4.2.2 NMF on heart sound segments . . . 56
3.4.2.3 NMF masking heart sound positions . . . 56
3.4.2.4 NMF on entire signal & replacing in heart sound positions . . . 61
3.4.3 Components assignment . . . 61
3.4.3.1 Spectral distribution . . . 61
3.4.3.2 Pure lung sound spectral correlation . . . 63
3.4.3.3 Heart sound position correlation . . . 64
3.4.4 System parameters . . . 66
3.5 Results analysis . . . 67
3.5.1 Selection of assignment criteria . . . 68
3.5.2 NMF parameters . . . 69
3.5.2.1 β-divergence . . . 69
3.5.2.2 Number of components K . . . 70
3.5.3 Methodology with NMF analysis . . . 70
3.5.4 Best results . . . 74
3.6 Conclusions and future work . . . 74

4 General conclusions and future work **78**
4.1 Conclusions . . . 78
4.2 Future work . . . 79

Appendix **82**

A Heart sound detection: More analysis results **82**
A.1 CNN based on classical architectures 82
A.2 Initial network design 84
A.3 Architecture analysis 87
A.4 Class balance analysis 88
A.5 Skipping connections analysis 89

B Source separation: More analysis results **92**
B.1 Spectrogram parameters 92
B.2 Simulation results over all parameters 94
B.3 Simulation results restricted to heart sound position correlation criterion 94

C Theory and analysis of input features at CNN **97**
C.1 Features implemented 97
C.1.1 Homomorphic filters 97
C.1.2 Hilbert transform 98
C.1.3 Wavelet transform 99
C.1.4 Multi-scale Wavelet product 100
C.1.5 Variance fractal dimension 101
C.1.6 Frequency bands energy 103
C.1.7 Spectral tracking 103
C.2 Pearson correlation coefficient analysis 104

D NMF theory: more about **108**
D.1 Results interpretation 108
D.2 Binary Mask 110
D.3 NMF properties 110
D.3.1 Intuitively interpretable results 112
D.3.2 Bounded solution 112
D.3.3 Number of components to decompose K 112
D.3.4 Non-unique solution 113
D.3.5 Cost function selection 113
D.3.6 β-divergence and from statistics 115
D.4 Solution algorithms 116
D.4.1 Multiplicative update (MU) algorithm 116
D.4.2 Gradient descent algorithm 117
D.4.3 Newton descent algorithm 117

REFERENCES **119**

LIST OF FIGURES

1.1 Lung sound taxonomy. 3
1.2 Electrocardiogram (ECG) illustrative signal. Extracted from SMART. 5
1.3 General diagram of the system implemented in this work. 7

2.1 Diagram of a CNN-based encoder-decoder architecture. 14
2.2 Diagram of the use of information from the encoder pooling layers in the inverse pooling implemented in the decoder of SegNet network. . 14
2.3 PCG and its labels. 17
2.4 Database file division for network training and testing. 17
2.5 Windowing process of the network input signal. 18
2.6 General scheme of the design of the heart sound detection system. . . 19
2.7 Graph of the binary sequence $y(n)$ for the heart sound positions. . . . 20
2.8 Features used in this work for a PCG extract. 21
2.9 Step diagram to obtain the modified Hilbert envelope. 21
2.10 Diagram of the resampling process for features that apply windowing processes. 24
2.11 Frequency band energy envelope resampled. 25
2.12 Proposed CNN based on encoder-decoder architectures. 27
2.13 Output of networks defined with 3 classes and their equivalent representation of 4 classes. 28
2.14 Variations in the number of classes used for the design of the encoder-decoder network. 37

3.1 NMF decomposition example. 47
3.2 Results of the NMF decomposition (in dB) using Wiener filter for $K = 3$ components. 49
3.3 General diagram of the NMF decomposition process for source separation. 50
3.4 Noise reduction process in segments free of heart sound. 52
3.5 General diagram of the source separation. 53
3.6 Diagram of the base method of NMF decomposition. 54
3.7 Heart sound segmentation graphs for NMF on heart sound segments method. 55

3.8 Diagram of the NMF decomposition method on heart sound segments. 57
3.9 Transition zones between different signal segments. 58
3.10 Diagram of the NMF masking heart sound positions method. 59
3.11 Heart sound segmentation plots for the NMF masking heart sound positions method. 60
3.12 Diagram of the NMF on entire signal & replacing in heart sound positions method. 62
3.13 Graphs of cardiac activations. 65
3.14 Cardiac activation in specific heart sound segment. 67
3.15 Histogram of results for assignment criteria (restricted to $N_{wind} = 2048$ and 90% overlap). 68
3.16 Histogram of results for β (restricted to $N_{wind} = 2048$ and 90% overlap). 70
3.17 Histogram of results for the number of components K (restricted to $N_{wind} = 2048$ and 90%). 71
3.18 Example of a spectrogram of a heart sound segment with $N_{wind} = 2048$ and 90% overlap. 72
3.19 NMF decomposition example on a heart sound segment with $K = 2$. . 73
3.20 Separation of the original cardio-respiratory signal into its lung and heart sound components using NMF replacing segments with $\beta = 2$, $K = 2$, $N_{wind} = 2048$, 90% overlap and heart sound position criterion. 75
3.21 Comparison between original lung sound and obtained by NMF replacing segments with $\beta = 2$, $K = 2$, $N_{wind} = 2048$, 90% overlap and heart sound position criterion. 75
3.22 Comparison between the PSDs of the original lung sound and the obtained by NMF replacing segments with $\beta = 2$, $K = 2$, $N_{wind} = 2048$, 90% overlap and heart sound position criterion. 76
3.23 Comparison of the spectrograms between the original cardio-respiratory sound and the lung signal obtained by NMF replacing segments with $\beta = 2$, $K = 2$, $N_{wind} = 2048$, 90% overlap and heart sound position criterion. 76

A.1 General operation scheme of a CNN based on classical architectures. . 83
A.2 Proposed CNN based on classical architectures. 85
A.3 Diagram of the network using multiple channels. 86
A.4 Connections between the corresponding encoder and decoder layers. . 90

B.1 Histogram of results for N. 93
B.2 Histogram of results for overlap. 93
B.3 Histogram of results for assignment criteria. 94
B.4 Histogram of results for β. 95
B.5 Histogram of results for the number of components in the decomposition K. 95

B.6 Histogram of results for β. 96
B.7 Histogram of results for the number of components in the decomposition K. 96

C.1 Phase frequency response of the Hilbert transform. 98
C.2 Esquema de descomposición DWT en múltiples niveles 100

D.1 Illustrative NMF decomposition example. 109
D.2 Results of the NMF decomposition (in dB) with binary mask for $K = 3$ components. 111
D.3 Effect of K on NMF. 114
D.4 β-divergence plots 115

LIST OF TABLES

2.1 Time duration for every set and label in the database. 16
2.2 Results of different combinations of features on the input of a CNN-based encoder-decoder architecture. 31
2.3 Detail of the abbreviations of the features presented in the table 2.2. . 31
2.4 Results of the increase in the number of filters as one deeps over a CNNs-based encoder-decoder architecture. 33
2.5 Results of the analysis of the length of the filters as one deeps over a CNNs-based encoder-decoder architecture. 33
2.6 Results of the analysis of the number of convolutional layers (depth) used in CNNs-based encoder-decoder architecture. 34
2.7 Results of the analysis of the training windows length N_x and the step τ_x for the inputs of the CNN-based encoder-decoder architecture. . . 35
2.8 Results of the analysis of the number of labels to be used in the CNN-based encoder-decoder architecture. 36
2.9 Results of k-fold cross-validation with $k = 10$ for the CNN-based encoder-decoder architecture configured from the previous steps. . . . 38

3.1 Results of the NMF methods comparing the original lung signal and the lung signal obtained. 71

A.1 Results of CNN based on classical architectures. 87
A.2 Results of CNN using encoder-decoder architectures. 88
A.3 Results of class balancing analysis for the weights in the objective function of the CNN-based encoder-decoder architecture. 89
A.4 Results of the analysis of the skipping connections between the encoder and decoder layers of the CNN based encoder-decoder architecture. 90

C.1 Study of the correlations between the binary signal of heart sound positions and the different levels of detail of a DWT with mother Wavelet *db6*. 104
C.2 Study of the correlations between the binary signal of heart sound positions and the product between different levels of detail of a discrete Wavelet transform with mother Wavelet db6. 104

C.3 Study of the correlations between the binary signal of heart sound positions and the energy for different frequency bands with different window length and step parameters. 105
C.4 Study of the correlations between the binary signal of heart sound positions and the VFD for different parameters of window length and step. 106
C.5 Study of the correlations between the binary signal of heart sound positions and the spectral tracking for different frequencies with different window length and step parameters. 106
C.6 Table of descriptors ordered by correlation (Pearson's coefficient) with the heart sound positions. 107

Chapter 1

Introduction and context of the thesis

1.1 Auscultation as a diagnostic method

Auscultation is the process of listening and interpreting sounds generated by the internal organs of the body. In 1816, Rene Laennec invented the stethoscope, an instrument that to this day has been used to diagnose different types of diseases (Welsby et al., 2003). Among the main fields in which this method has been used is in the diagnosis of heart and lung diseases (Mondal, Saxena, et al., 2018). Physicians use auscultation as a simple, non-invasive and low-cost method to obtain relevant information about the state of internal organs (Fernando et al., 2020; Gavrovska et al., 2014; Gupta et al., 2007; Oliveira et al., 2019; Saha and P. Kumar, 2004; Shah, Koch, et al., 2015; Shah and C. Papadias, 2013; S. Sun, Jiang, et al., 2014; Tang et al., 2012; Vepa et al., 2008; Yuenyong et al., 2011), being one of the most reliable, effective and easily repeatable method to make an early diagnosis without generating risks in the patient (Bahoura, 2009; Bardou et al., 2018; Demir et al., 2020; Dokur, 2009; Gill et al., 2005; Kandaswamy et al., 2004; D. Kumar, Carvalho, Antunes, Gil, et al., 2006; Palaniappan et al., 2013; Sankur et al., 1994; Sengupta et al., 2016; Xie et al., 2012).

However, one of the main difficulties of diagnosis by auscultation is that the interference generated between heart and lung sounds can hide/mask characteristics of each one, which can lead to erroneous diagnoses (Mondal, P. S. Bhattacharya, et al., 2011). In addition, many of these sounds are in a low frequency band, in which the human ear is not very sensitive (Ari et al., 2008; Bahoura, 2009; S. Banerjee et al., 2016; Bardou et al., 2018; Kandaswamy et al., 2004; Mondal, P. S. Bhattacharya, et al., 2011; Omari and Bereksi-Reguig, 2015; Palaniappan et al., 2013; S. E. Schmidt et al., 2010; Sengupta et al., 2016). Finally, the interpretation of these sounds is subject to the experience, skills and auditory training of the physician who performs the auscultation (Ali et al., 2017; S. Banerjee et al., 2016; Chen et al., 2017; Gill

et al., 2005; Huiying et al., 1997; Liang et al., 1997; Moukadem, S. Schmidt, et al., 2015; Mubarak et al., 2018; Nazeran, 2007; Oliveira et al., 2019; Papadaniil and Hadjileontiadis, 2014; Pedrosa et al., 2014; Sedighian et al., 2014; Shanthakumari and Priya, 2019; Tang et al., 2012; Yuenyong et al., 2011).

In addition, in (Hafke-Dys et al., 2019) it is mentioned that "the medical community recognizes the problem of the ambiguous nomenclature used in the classification of respiratory sounds", which is an additional difficulty when communicating a patient's diagnosis. Along the same lines, the study of (Hafke-Dys et al., 2019) reports that medical students achieve an average of 24.1% correct diagnoses in the auscultation of respiratory sounds using the appropriate nomenclature, while pulmonologists only 36.5%.

With the development of the digital stethoscope, considerable progress has been made in the analysis, study and processing of sounds, allowing easier diagnoses of certain diseases (Várady, 2001). Indeed, with these types of tools it is possible the study of the occurrence times and the relative intensities of each sound in a visual representation, allowing to reveal information that even the human ear cannot process (Messer et al., 2001; Omari and Bereksi-Reguig, 2015).

1.2 Lung sounds

Lung sounds are produced by a turbulent flow of air within the respiratory tract during inhalation and exhalation processes (T. E. A. Khan and Vijayakumar, 2010; Mondal, P. Bhattacharya, et al., 2013; Potdar et al., 2012), mainly in the bronchi and trachea (Welsby et al., 2003). Product of the vibrations generated by turbulence, this flow propagates in form of sound through the lung tissue which can be heard on the chest wall (Nersisson and Noel, 2017). In the process, the tissue involved acts as a low-pass filter (Welsby et al., 2003) that varies depending on possible changes in the structure, generated by pathologies that affect the lung physiology and/or respiratory airways (Pourazad, Z. Moussavi, and Thomas, 2006; Tsalaile et al., 2008). Therefore, lung sounds auscultation provide signs of excessive secretions or evidence of inflammation of the lungs (Jones et al., 1999), which can be related to diseases such as asthma, tuberculosis, chronic obstructive pulmonary diseases (COPD), pneumonia and bronchiectasis (Ahlstrom et al., 2005; T. E. A. Khan and Vijayakumar, 2010).

The lung sounds of a normal patient generally occurs in the frequency band between 20-2000 Hz (Canadas-Quesada et al., 2017; Dokur, 2009; N. Gavriely et al., 1981; Kandaswamy et al., 2004; Sankur et al., 1994), whose energy peak is located between 20-100 Hz. From 200 Hz the signal presents an abrupt drop in energy (Ghaderi et al., 2011; Nersisson and Noel, 2017; Pasterkamp et al., 1997; Yadollahi and Z. M. Moussavi, 2006).

In general, lung sounds can be classified into two classes, as presented in the

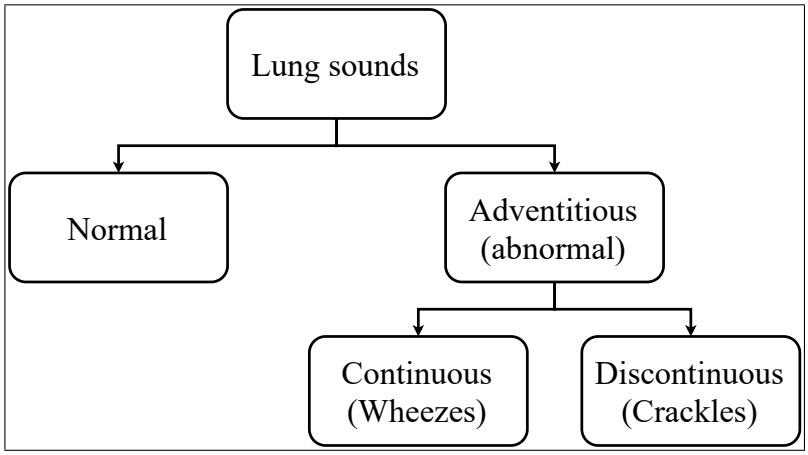

Figure 1.1: Lung sound taxonomy.

figure 1.1 (Bardou et al., 2018; Kandaswamy et al., 2004; Sankur et al., 1994; Xie et al., 2012): normal lung sounds and adventitious lung sounds (abnormal).

Normal lung sounds have the characteristics just mentioned. On the other hand, adventitious lung sounds are sounds superimposed on normal lung sounds, and can occur on both inhalation and exhalation (Dokur, 2009). In general, they are symptoms of some pulmonary disorder, and can be divided into two main classes: continuous and discontinuous (Bahoura, 2009; Bardou et al., 2018; Sankur et al., 1994; Sengupta et al., 2016; Xie et al., 2012).

Continuous adventitious lung sounds may be caused by inflammation or obstruction of the respiratory lumen (Dokur, 2009; Sengupta et al., 2016; Xie et al., 2012). Within this category, wheezing are tonal sounds whose duration is greater than 100 ms, and whose predominant frequency range is greater than 100 Hz (Sankur et al., 1994; Sengupta et al., 2016). They can be monophonic if it contains only one frequency, or even polyphonic if it contains more than one frequency (Bardou et al., 2018). They are known for the vibration of the bronchi (Bahoura, 2009), and for this reason, they are usually associated with obstruction of the bronchial passages due to bronchospasm, mucosal edema or external compression by a tumor mass (Bahoura, 2009; Sengupta et al., 2016). These types of symptoms are associated with diseases such as asthma and COPDs such as pulmonary emphysema, chronic bronchitis and cystic fibrosis (Sengupta et al., 2016).

Discontinuous adventitious sounds are caused by explosive openings of small airways that are abnormally closed by fluid (bubbling of air through mucus), inflammation, or infections of small bronchi, bronchioles, and alveoli (Dokur, 2009; Sankur et al., 1994; Sengupta et al., 2016; Xie et al., 2012). Crackles are discontinuous adventitious sounds that are characterized by being short, explosive and non-tonal whose duration is typically less than 20 ms (Kandaswamy et al., 2004; Xie et al., 2012). They tend to appear on inhalation and to a lesser proportion on exhalation

(Bardou et al., 2018; Sengupta et al., 2016). Its frequency spectrum is located between 60-2000 Hz, predominating below 1200 Hz (Sarkar et al., 2015). This type of sounds is related to restrictive lung diseases such as pneumonia, bronchiectasis, pulmonary fibrosis and fibrosing alveolitis (Sankur et al., 1994).

1.3 Heart sounds

The cardiovascular system is made up mainly of the heart and blood vessels, and is responsible for transporting blood throughout the body in 2 main circuits: the pulmonary (to the lungs) and the systemic (to the rest of the body). The heart is a four-chamber pump: two atria that receive blood from the veins, and two ventricles that pump blood into the arteries (Gill et al., 2005). The mechanical activities of the heart are always caused by its electrical activity (Sepehri et al., 2010).

Heart sounds are quasi-periodic signals originated by the flow of blood that circulates through the heart in conjunction with the movement of its own structure (T. E. A. Khan and Vijayakumar, 2010; Mondal, P. Banerjee, et al., 2017; Nersisson and Noel, 2017). The heart sounds recordings are called phonocardiograms (PCG). They contain the acoustic waves produced by the opening and closing of the heart valves, and even sounds produced by the turbulence generated by the blood flow (Gamero and Watrous, 2003; Haghighi-Mood and Torry, 1995; T. E. A. Khan and Vijayakumar, 2010; Mondal, P. Banerjee, et al., 2017; Nersisson and Noel, 2017; Varghees and Ramachandran, 2014). The PCG is a non-stationary signal that contains valuable information for the diagnosis of certain diseases, showing heart murmurs, arrhythmias and other types of aberrations originated from its structure or heart activity (ARI and SAHA, 2007; Chen et al., 2017; Fernando et al., 2020; Gupta et al., 2007; Haghighi-Mood and Torry, 1995; Huiying et al., 1997; A. Iwata et al., 1980; Lima and Barbosa, 2008; Nazeran, 2007; Nivitha Varghees and Ramachandran, 2017; Noman et al., 2020; Sedighian et al., 2014; Várady, 2001; Varghees and Ramachandran, 2014; P. Wang et al., 2005).

The main components of the PCG are the first heart sound (S_1) and the second heart sound (S_2). S_1 is generated during ventricular systole (closure of the atrioventricular valves: mitral/bicuspid and tricuspid), in which the ventricles contract and allow blood to be pumped from the heart to the rest of the body through the aorta and pulmonary arteries (Hamza Cherif et al., 2008; Mondal, P. Bhattacharya, et al., 2013; Papadaniil and Hadjileontiadis, 2014; Sepehri et al., 2010). Each of the phases of the heart sound is closely related to events generated by the electrical activity of the heart, which can be presented in the electrocardiogram (ECG). The example of a cardiac cycle seen on an ECG is presented in figure 1.2. When studying the cardiac activity using an ECG, it is possible to notice that S_1 in general coincides with the beginning of the R wave of the QRS complex (Haghighi-Mood and Torry, 1995; Lima and Barbosa, 2008; Moukadem, S. Schmidt, et al., 2015; Sepehri et al., 2010). On

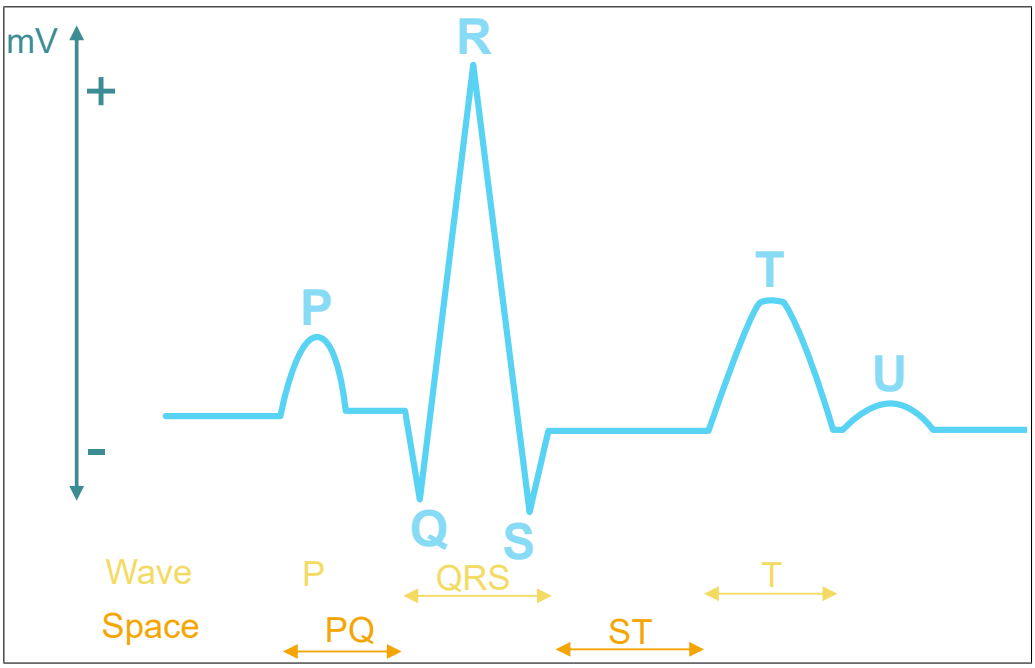

Figure 1.2: Electrocardiogram (ECG) illustrative signal. Extracted from SMART.

the other hand, S_2 occurs during ventricular diastole (closure of the sigmoid/semilunar valves: aortic and pulmonary) in which the ventricles relax and allow the entry of blood from the atria (Hamza Cherif et al., 2008; Mondal, P. Bhattacharya, et al., 2013; Papadaniil and Hadjileontiadis, 2014; Sepehri et al., 2010), and it happens right after the T wave on the ECG (Haghighi-Mood and Torry, 1995; Moukadem, S. Schmidt, et al., 2015; Sepehri et al., 2010). Sounds like the third (S_3) and fourth (S_4) heart sounds appear to a lesser extent (Ali et al., 2017; Choi and Jiang, 2008; Omari and Bereksi-Reguig, 2015). S_3 occurs shortly after S_2 and is associated with premature diastolic filling of the ventricle. While S_4 occurs just before S_1 and is associated with late filling of the ventricle, being common in children and adults over 50 years old (Iaizzo, 2005; Malarvili et al., 2003; Messner et al., 2018; Mondal, P. Bhattacharya, et al., 2013; Reed et al., 2004).

Their frequency spectra typically lies between 10-200 Hz (Choi and Jiang, 2008; Lehner and Rangayyan, 1987; Naseri and Homaeinezhad, 2013). In comparison, S_1 is a lower-pitched sound with a longer duration, while S_2 is a higher-pitched sound with a shorter duration (Chen et al., 2017; Iaizzo, 2005). In (C. Liu et al., 2016) it is indicated that S_1 predominates in the 10-140 Hz frequency band and S_2 in the 10-200 Hz band. In turn, the S_3 and S_4 sounds in general have very low amplitude and low frequency (between 20-70 Hz) (C. Liu et al., 2016). When the cardiac system presents some dysfunction such as murmurs or mitral valve stenosis, the spectral content could even expand to 600-700 Hz (Gharehbaghi et al., 2011; Lehner and Rangayyan, 1987; Mondal, P. Banerjee, et al., 2017; Naseri and Homaeinezhad, 2013; Nazeran, 2007).

The typical duration of fundamental sounds is between 80-170 ms (Luisada et al., 1949). In relation to the length of the intervals, the diastolic interval is typically longer than the systolic interval. Furthermore, the length of the systolic period is relatively constant compared to the diastolic. Indeed, with increasing heart rate, the duration of systole tends to decrease while diastole decreases significantly (Ari et al., 2008; Castro et al., 2013; Chen et al., 2017; Huiying et al., 1997; Nivitha Varghees and Ramachandran, 2017; Rajan et al., 2006; Saha and P. Kumar, 2004; P. Wang et al., 2005; Yuenyong et al., 2011).

Abnormalities such as heart murmurs are generated by turbulent blood flow in blood vessels (Dinesh Kumar et al., 2006). They can be due to thin pulmonary valves with stenosis, whose structure can generate regurgitation causing the blood flow to take a direction opposite to normal. In the case of mitral regurgitation, the valve does not close properly causing regurgitation in the left atrium when the left ventricle contracts, resulting in a characteristic murmur (Boutana et al., 2011; Yuenyong et al., 2011). The presence of murmurs is a good indicator of valve disorders (Yuenyong et al., 2011).

1.4 Research scope

With the growing boom of machine learning methods in areas such as image processing, speech recognition, natural language processing, economics and finance among others. It has been shown that these types of algorithms can be extremely robust to solve various problems.

Several authors have raised the challenge of designing an algorithm that allows objective and quantitative diagnoses from auscultated sounds using methods based on machine learning (Bardou et al., 2018; Demir et al., 2020; Palaniappan et al., 2013; Sengupta et al., 2016). However, mutual interference between heart and lung sounds could be detrimental to the performance of the lung sound classification system.

The objective of this work is to design a preprocessing system that allows to obtain a clean lung sound. The resulted sound can be used as an input signal in a lung sound classification system. In particular, it will be sought to separate the heart sound from the lung sound, trying to preserve the properties of each sound. The design of the proposed system is presented in the figure 1.3.

Because the heart is very close to the lungs in the chest, there may be the presence of heart sounds in the auscultated lung signal. This is not desirable since it can mask properties of interest of the lung sound or generate false correlations between the signals

Therefore, and in order to reduce the presence of heart sound in the auscultated signal, a prior process of source separation will be performed. In the following chapters, the implementation of the heart sound detection algorithm (chapter 2),

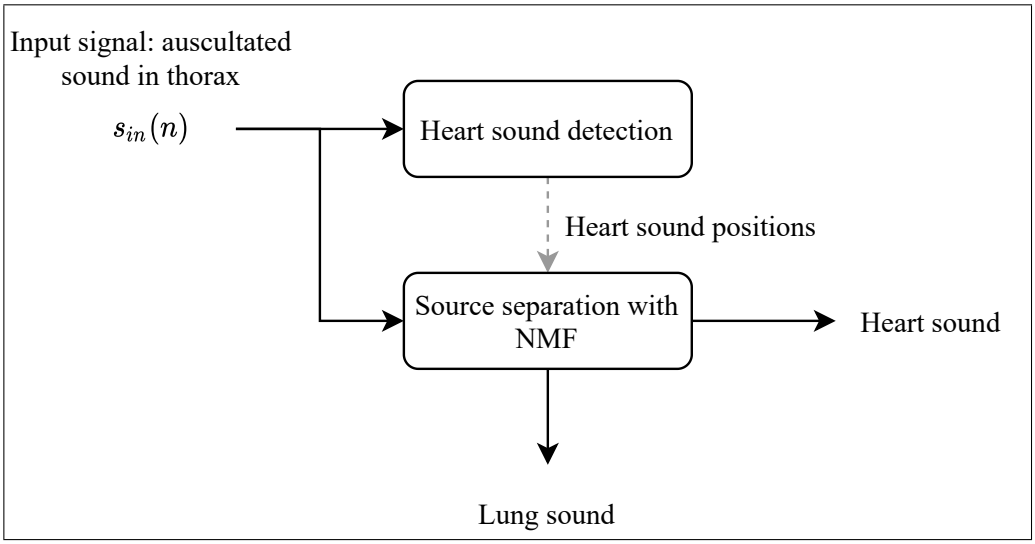

Figure 1.3: General diagram of the system implemented in this work.

and source separation between heart and lung sounds (chapter 3) will be presented in detail. Both chapters correspond to independent articles that were sent to specialized journals in their areas.

Chapter 2

Heart sound detection

Knowing the heart sound positions could be useful when implementing source separation methods on chest auscultated signals. Indeed, this would allow to separate the lung sound from the heart sound applying algorithms that act only on the segments where the fundamental heart sounds are located. In addition, a correct detection and identification of these sounds could allow the development of an automatic system that detects cardiac diseases or dysfunctions through auscultation (Sepehri et al., 2010), since it becomes relatively easy to predict any pathology by studying the presence and severity of murmurs or sounds such as S_3 and S_4 in each segment, which provide valuable information about the patient's diagnosis (Ari et al., 2008; Gavrovska et al., 2014; Gupta et al., 2007; Moukadem, Dieterlen, et al., 2013; Mubarak et al., 2018; Nivitha Varghees and Ramachandran, 2017; Renna et al., 2019; Springer et al., 2016; Varghees and Ramachandran, 2014; Vepa et al., 2008). The results of the *Physionet* challenge on 2016 revealed that algorithms that used a previous segmentation stage generally performed better in classifying pathologies from a PCG (Fernando et al., 2020).

However, the automatic cardiac cycle segmentation without the use of guide signals such as the ECG or the carotid pulse is a complicated task (Gill et al., 2005; Sepehri et al., 2010). Also, there are some factors that make it difficult to segment heart sounds. Indeed, the captured recordings of the chest wall are a mixture of respiratory sounds, sounds produced by intestinal activity, external noises, artifacts produced by movements or frictions (such as clicks), variations of the heart rate, scenarios of low SNR and even the presence of other heart sounds such as murmurs, S_3 and S_4 (Chen et al., 2017; Gamero and Watrous, 2003; Ghaderi et al., 2011; Golpaygani et al., 2015; Hassani et al., 2014; Mubarak et al., 2018; Nivitha Varghees and Ramachandran, 2017; Rajan et al., 2006; S. E. Schmidt et al., 2010; Sedighian et al., 2014; Sepehri et al., 2010; P. Wang et al., 2005). All these events can vary in duration, amplitude and spectral characteristics from one cycle to another, between recordings, and even between different patients (Huiying et al., 1997). Due to this, the detection of fundamental sounds using only the PCG is a complex problem

(Huiying et al., 1997; Rajan et al., 2006).

Due to the growing rise of Convolutional Neural Networks (CNN) and their good results in object detection in images, the study of these networks in the detection and identification of fundamental heart sounds in a PCG is proposed. One of the advantages of CNNs is that they allow independence of the temporal relationships of the signal, since each convolutional layer can be understood as a filter that is adjusted to detect the segments of interest within the heart sound. The detection will depend on the intrinsic characteristics of each segment, which will be exploited by the neural network.

In this chapter we propose the study of a CNN based on an encoder-decoder architecture that allows segmenting and detecting heart sounds in a PCG. The main contributions of this work are:

- A study and proposal of other features used to characterize heart sounds. In works such as (Renna et al., 2019; S. E. Schmidt et al., 2010; Springer et al., 2016) the typical features used are the outputs of homomorphic filters, Hilbert envelopes, Discrete Wavelet Transforms and PSD on certain bands. In this proposal we will analyze the combination of features that offers the best results among several proposed descriptors such as the Variance Fractal Dimension, Multi-scale Wavelet Product and Spectral Tracking, which until now have been used in envelopes and peaks detection based methods.

- A systematic study of the different parameters that define the network, such as the number and length of the filters in each layer, the depth of the network, and the size of the window used in the process of windowing the input signals for network training.

- An analysis of the number of classes to define for the design of the network. Indeed, typically in the literature 4 classes are used: S_1, systole, S_2 and diastole. However, we will study whether it is convenient to use 2, 3 or 4 classes depending on the performance of the network.

Next, one of the articles prepared in this research and sent for publication entitled "*Heart sound segmentation using Convolutional Neural Networks based encoder-decoder*" will be presented. It should be noted that for this chapter the introduction of the article will be omitted, since it overlays the information in chapter 1.

The CNN study was also carried out using classical architectures (AlexNet, LeNet) and techniques such as class balance and skipping connections which was not included in the article. However, the detail of the analysis are available in the chapter A of appendix. Furthermore, the detail on the features used in this section, is available in chapter C in the appendix.

The rest of this chapter is organized as follows: section 2.1 explains different methods implemented in the literature. In section 2.2 the general theory of the CNN

to be used is presented. In section 2.3 the database and its division for training and testing the network is detailed. In the section 2.4 the parameters to be used in the preprocessing, the initialization of the network and the descriptors parameters are detailed. In the section 2.5 the experiments to be performed are presented, analyzing on the results of each one. And finally, in the section 2.6 the main conclusions of this chapter are presented.

2.1 Related works

To solve the problem of heart sound segmentation using a PCG, various methods have been proposed. They can be classified into 5 categories (C. Liu et al., 2016; Renna et al., 2019): envelope and peak detection; synchronized external signal; feature extraction and classification; sequential models; and neural networks.

The envelope based methods are the most used in the different researches of the last decades. In these methods, it is sought to obtain a characteristic representation that allows to emphasize the heart sounds (Moukadem, Dieterlen, et al., 2013; Renna et al., 2019). Once the characteristic are obtained, algorithms for peaks detection or use of thresholds, algorithms for edge detection and criteria for eliminating false detection are applied. To identify and classify the heart sounds segments, typically they make use of the fundamental sounds durations, and the systolic/diastolic intervals durations. Generally, these decisions are based on the fact that the duration of diastole is longer than that of systole (Ari et al., 2008). Sometimes it is necessary to remove extra peaks and retain those that correspond to the fundamental sounds through a process called peak conditioning. To obtain this representation, authors such as (Ari et al., 2008; ARI and SAHA, 2007; Hamza Cherif et al., 2008; Liang et al., 1997; Yamaçh et al., 2008) use Shannon energy to obtain a characteristic envelope, while in (Martinez-Alajarin and Ruiz-Merino, 2005; Saha and P. Kumar, 2004; Vepa et al., 2008) the energy over the time is used. Some more sophisticated methods such as (Carvalho et al., 2005; J. Gnitecki and Z. Moussavi, 2003) make use of the Variance Fractal Dimension (VFD), which provides a measure of the inherent complexity of a signal in terms of its morphology. In (Nigam and Priemer, 2005; Vepa et al., 2008) a method to obtain the envelope using the inherent complexity of the PCG based on the theory of system dynamics is proposed. Hilbert transform is another widely used method to address this problem, where the magnitude envelopes (Mondal, P. Bhattacharya, et al., 2013) and the instantaneous frequency (Martinez-Alajarin and Ruiz-Merino, 2005; S. Sun, Jiang, et al., 2014; Varghees and Ramachandran, 2014) defined from the analytic signal are used. In (S. Sun, Haibin Wang, et al., 2014; Yan et al., 2010) the use of a method based on obtaining the variance of the signal in a time window is presented, and that provides reference points that allow characterizing each cycle. Based on spectral features (A. Iwata et al., 1980) uses spectral tracking on a specific frequency, while in (Haghighi-Mood

and Torry, 1995) energy is tracked in certain frequency bands of the signal. Authors such as (Boutana et al., 2011; Liang et al., 1998) use spectrograms to make a time-frequency representation of heart sound. The Discrete Wavelet Transform (DWT) is a technique widely used in the literature to preprocess PCGs (Vepa et al., 2008; Yamaçh et al., 2008) and to obtain envelopes from their decomposition levels (Castro et al., 2013; Golpaygani et al., 2015; Huiying et al., 1997). In works such as (Meziani et al., 2012; H. Sun et al., 2013; P. Wang et al., 2005) the DWT is used to suppress PCG noise by applying thresholds on certain detail levels. In (Yadollahi and Z. M. Moussavi, 2006) the Multi-scale Wavelets Product is presented, offering good noise suppression properties. Another less common variant of Wavelets is proposed in (Nivitha Varghees and Ramachandran, 2017) using an Empirical Wavelet Transform approach for noise suppression and discrimination of cardiac murmurs. Although these methods are relatively simple to implement, the main difficulty they present is that in the presence of noise, false detections can occur due to unwanted peaks or undetected heart sounds, which depend on the position and how much the stethoscope is pressed on chest (Yuenyong et al., 2011). Furthermore, most of these methods use the systole and diastole durations to classify S_1 and S_2, which may be useless in cases of clinical symptoms such as tachycardia, arrhythmia, patients under cardiac stress activity or erroneous segmentations in which this criterion is invalidated (Moukadem, Dieterlen, et al., 2013; Moukadem, S. Schmidt, et al., 2015).

Methods based on the use of external signals require the use of additional synchronized instruments to determine the times of key events in the cardiac cycle. Authors such as (El-Segaier et al., 2005; Gharehbaghi et al., 2011; Malarvili et al., 2003) use the ECG as an additional signal, while (Lehner and Rangayyan, 1987) uses ECG and the carotid pulse to recognize fundamental heart sounds. However, one of the main problems with this type of technique is that the instruments necessary to complement the PCG information cannot always be available. Also, in the case of the ECG, the time between electrical and mechanical activities may vary between patients due to the existence of possible pathologies, which would make its use as a guide for segmentation less reliable (Haghighi-Mood and Torry, 1995; Malarvili et al., 2003; Nigam and Priemer, 2005; Varghees and Ramachandran, 2014; Vepa et al., 2008).

The features extraction and classifications based methods seek to find features that allow to characterize the PCG, which will be used in classification systems to determine what state of the cardiac cycle each one corresponds to. In works such as (D. Kumar, Carvalho, Antunes, Henriques, et al., 2006; Dinesh Kumar et al., 2006) a spectral energy distribution criterion is used to classify S_1 and S2. In (Tang et al., 2012) a decomposition method is applied using a Gaussian modulation, performing a dynamic grouping in the time-frequency plane and using a weighted density function as a threshold for detecting fundamental sounds. In (Papadaniil and Hadjileontiadis, 2014) a kurtosis based criterion for the detection of heart sounds is used. In (Ghaderi et al., 2011) a Singular Spectral Analysis (SSA) is performed to classify

based on the eigenvalues of the decomposition. Different types of classifiers such as fuzzy c-means (Carvalho et al., 2005), bagging classifiers (Yuenyong et al., 2011), k-means (Gavrovska et al., 2014; Pedrosa et al., 2014; Tseng et al., 2012), Support Vector Machine (SVM) (Mubarak et al., 2018; S. Sun, Haibin Wang, et al., 2014) and k-Nearest-Neighbors (kNN) (Moukadem, Dieterlen, et al., 2013; Moukadem, S. Schmidt, et al., 2015) have been used in the literature offering competent results. However, this kind of heart sound classification methods suffer similar problems to peak detection methods since they are dependent on the correct segmentation of the heart sound. If the segmentation is not performed correctly, the classifiers do not have correction algorithms to amend the problem.

The use of sequential models such as Hidden Markov Models (HMM) have generated a great advance in solving this problem. Works such as (Gamero and Watrous, 2003; Gill et al., 2005; Lima and Barbosa, 2008) use first-order HMMs to model each segment of interest in the PCG. A limitation of the HMM is that in these models the transition probability to a new state is independent of the time the model has remained in the same state. As a result of this, (S. E. Schmidt et al., 2010) proposes the use of a Hidden Semi-Markov Model (HSMM), which allows modeling the expected duration of the heart sound in each of the states, consolidating itself as the state of the art at that moment. In (Springer et al., 2016) a logistic function is also incorporated to estimate the probability of emission between states, and the classic Viterbi algorithm is adapted by correcting problems at the edges that the original algorithm could not solve, consolidating itself as the state of the art of heart sound segmentation (Renna et al., 2019). In (Oliveira et al., 2019) a method similar to (Springer et al., 2016) is proposed, but with the ability to adjust the sojourn time of the states considering the characteristics of the input PCG. In recent works such as (Noman et al., 2020), HMM is used with Switching Linear Dynamics System (SLDS) which allows each state to be modeled as a linear dynamic process.

Finally, the methods based on neural networks use features or signal envelopes as input to a neural network, which classifies the PCG segments into the different interest labels. In (Oskiper and Watrous, 2002) a Time Delay Neural Network (TDNN) is implemented to detect S_1 with a single hidden layer. In work such as (Chen et al., 2017; Tsao et al., 2019) Deep Neural Networks (DNN) are used, incorporating in (Tsao et al., 2019) a spectral restoration algorithm that allows to reduce noise in the frequency domain. In works such as (Fernando et al., 2020; Messner et al., 2018) the implementation of Recurrent Neural Networks (RNN) is studied, testing with different architectures such as *Vanilla RNN*, *Long Short Term Memory Networks* (LSTM), *Gated RNN* (GRNN), *Bidirectional RNN* (BiRNN), and even the new attention blocks in (Fernando et al., 2020). The RNNs allow processing sequential inputs of variable length, and automatically learn the temporal relationships between them. In (Renna et al., 2019) Convolutional Neural Networks (CNN) are used, whose architecture is inspired by the U-Net network originally used in image segmentation. In general, neural network-based methods allow to avoid the typical

problems of envelope analysis and peaks detection. However, one of the difficulties they present is that it is necessary to develop a training set large enough for the development of these models (Yuenyong et al., 2011).

2.2 Theoretical background

2.2.1 CNN based Encoder-Decoder

One of the main characteristics of CNNs is their ability to exploit the temporal or spatial correlation of data due to its convolutional nature (A. Khan et al., 2020). The convolution relations in a CNN are given by:

$$\begin{aligned} \tilde{x}_k^{[l]}(n) &= b_k^{[l]} + \sum_{j=1}^{n_c^{[l-1]}} \sum_{m=1}^{n_N^{[l-1]}} h_j^{[l]}(n-m+1) \cdot x_j^{[l-1]}(m) \\ x_k^{[l]}(n) &= g^{[l]}\left(\tilde{x}_k^{[l]}(n)\right) \end{aligned} \tag{2.1}$$

Where $h_k^{[l]}(n)$ corresponds to the k-th filter of the l-th convolutional layer, $b_k^{[l]}$ to its bias parameter, $n_c^{[l]}$ is the number of filters defined in the l-th layer, $n_N^{[l]}$ is the length of the output of the l-th convolutional layer, $g^{[l]}(\cdot)$ is the activation function and $\tilde{x}_k^{[l]}(n)$ is the output of the k-th filter in the l-th convolution layer.

CNNs based on classical architectures such as the LeNet-5 (Yann LeCun et al., 1998) or the AlexNet (Krizhevsky et al., 2012) are typically composed of a series of convolutional layers, a flattening layer, and finally a perceptron network from which a single output class is obtained. In the case of the heart sounds segmentation, this could generate resolution problems since when entering a segment of the PCG, the network will reduce that entire segment to a single label, which is not desirable.

For this reason, this work proposes the use of a CNN based on the SegNet network (Badrinarayanan et al., 2017), an encoder-decoder architecture with CNN that presents good results in image segmentation applications, which can be divided into 3 main sections as shown in figure 2.1. This network will be adapted to operate with PCG audio signals (1D).

The first section is the encoder network, and it is composed of a series of encoders that make it possible to generate a low-resolution representation of the input signal (Badrinarayanan et al., 2017). Each encoder consists of a series of convolutional layers, followed by batch normalization, activation function (typically with ReLU function) and a pooling function (typically maxpooling), which allows to downsample the signal throughout the network (Renna et al., 2019). Although maxpooling allows for better results and translation invariance on small shifts in the input signal, the use of many layers of pooling causes a loss of resolution.

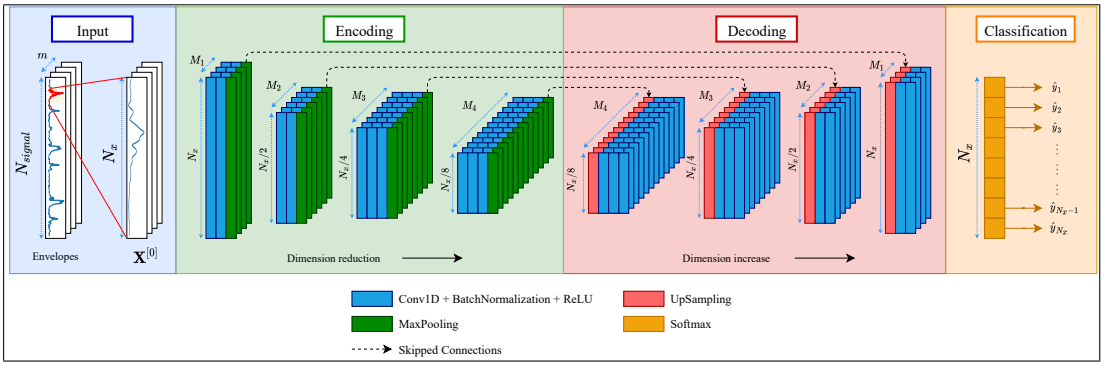

Figure 2.1: Diagram of a CNN-based encoder-decoder architecture.

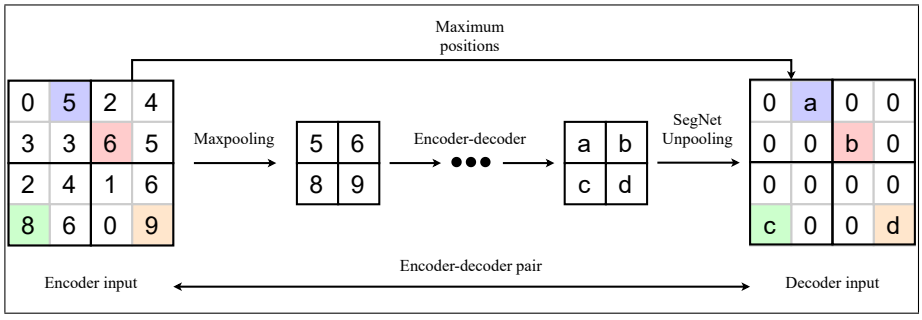

Figure 2.2: Diagram of the use of information from the encoder pooling layers in the inverse pooling implemented in the decoder of SegNet network. This is applied on a 4×4 dimension image.

The second section is the decoder network, and it is composed of a series of decoders that allow generating multi-dimensional characteristics from the low resolution representations obtained from the encoder (Badrinarayanan et al., 2017). Each layer provides a higher resolution representation that will serve as input to the next decoder, and consists of a series of convolutional layers with batch normalization and ReLU, just like the encoders. However, instead of using a pooling function, an inverse function is implemented that increases the number of samples. In this network, each encoder has a corresponding decoder that allows the reverse process to be performed up to the classification layer (Ye and Sung, 2019). In the SegNet network, for each corresponding encoder-decoder pair, it is proposed to use the position of the maximum value obtained in the maxpooling layers in the encoders to implement the inverse process, as can be seen in the figure 2.2.

Finally, at the output of the last decoder, a classifier based on the softmax function is used that classifies each sample of the input signal independently, indicating the probability that each sample belongs to any of the K classes through the expres-

sion:

$$\hat{y}_i(n) = \Phi_i\left(\mathbf{X}^{[L]}(n)\right) = \frac{\exp\left(x_i^{[L]}(n)\right)}{\sum_{k=1}^{K}\exp\left(x_k^{[L]}(n)\right)}\,, \tag{2.2}$$

Where the sum of the outputs of this layer is 1, $\forall n = \{1, ..., K\}$. Therefore, this type of network allows classifying while maintaining the length N_x of the input signal. To infer the class of each point, the class that has the maximum estimated probability among all the classes is chosen.

Finally, to train this network a cross entropy cost function is used, which is defined as (Aggarwal, 2018b):

$$\mathcal{L}_\Theta\left(y(n), \hat{y}(n)\right) = -\sum_{i=1}^{K} y_i(n)\log\left(\hat{y}_i(n)\right) \tag{2.3}$$

Where Θ represents the set of parameters that defines the network, $y_i(n) \in \{0, 1\}$, $\forall i = \{1, ..., K\}$ corresponds to the observed value or label, and $\hat{y}_i(n) \in \{0, 1\}$, $\forall i = \{1, ..., K\}$ is the probability obtained from of the softmax layer for the i-th class.

The base design of the proposed network is presented in figure 2.1. As can be seen, and unlike the original SegNet network, the encoder consists of 10 convolutional layers with batch normalization and ReLU activation, using 4 maxpooling layers after the second, fourth, seventh and tenth convolutional layers. In convolutional layers, zero padding will be used to ensure that the same number of points is maintained at the output as at the input. In addition, maxpooling will be used with length and step 2, which will allow the number of points to be cut in half each time it is implemented. For the upsampling process in the decoder, the values from the previous layer will simply be duplicated. This choice is due to the fact that performing the proposed operation in the SegNet network (see figure 2.2) on a one-dimensional signal does not generate a great difference with replicating the value, compared to signals in two dimensions. Therefore, in order to keep the network simple, the value will simply be repeated. The output of this network is a classification sample by sample, obtaining the same dimension N_x of the input signal to the network.

2.3 Database

To perform this study, we use a database of heart sounds available in the Springer implementation (Springer et al., 2016), presented for the heart sound segmentation stage in the context of the 2016 PhysioNet/CinC challenge (Goldberger et al., 2000). This dataset contains 792 audio records obtained from 135 different patients, which

Table 2.1: Time duration for every set and label in the database.

Label	**Duration (minutes)**			
	Train	**Validation**	**Test**	**Total label**
S_1	22.642	2.037	2.603	27.283
Systole	30.244	2.931	3.760	36.936
S_2	16.069	1.444	1.843	19.356
Diastole	68.186	6.229	8.252	82.667
Total set	137.142	12.642	16.458	166.242

are auscultated in different positions, resulting in a total of 166.24 minutes of recording. Each of these audio files is sampled at 1000 Hz, and has labels sampled at 50 Hz indicating 4 possible states: S_1, systole, S_2 and diastole (table 2.1 shows the detail of the duration of each set and label in this database). These labels are defined with the R peak and the end of the T wave of an ECG synchronized with the stethoscope used for recording the heart sounds. However, none of the labels have human correction. In addition, each file contains the ID number and the diagnosis of the patient, indicating whether or not they have the presence of a pathology (the most common are mitral valve prolapse).

To coordinate the sample rate of the PCGs with the sample rate of the labels, each point on the labels is simply repeated 20 times. With this, it is possible to obtain a representation of the auscultated signal synchronized with its respective labels, as can be seen in the figure 2.3.

For this work, the dataset will be divided into 90% (712 PCG recordings of 120 patients) for training and validation of the system, while the remaining 10% will be used for testing (80 PCG recordings of 15 patients). It should be noticed that in this division the patients belonging to the training and validation set are different from the patients in the testing set. This helps to ensure that the results obtained really indicate how the system is performing in segmenting the heart sounds in a general situation, and not the heart sounds of a particular person.

In turn, the set designated for training and validation is divided into 90% for training (640 files) and 10% for validation (72 files) without considering the division of patients. That is, it is possible for patients to be repeated in the training and validation sets. The reason for this decision is based on contrasting the results of the validation and testing set against the adjustment of the network, since the performance and behavior of the network on PCGs of patients who are in the training set could be studied, in comparison to PCGs of patients that the network does not have integrated. These divisions are illustrated in the diagram of figure 2.4.

Finally, for each of the audio files in this dataset, windows of length N_x with step τ_x will be used. In the extreme case of the edge of the signal, if the number of points

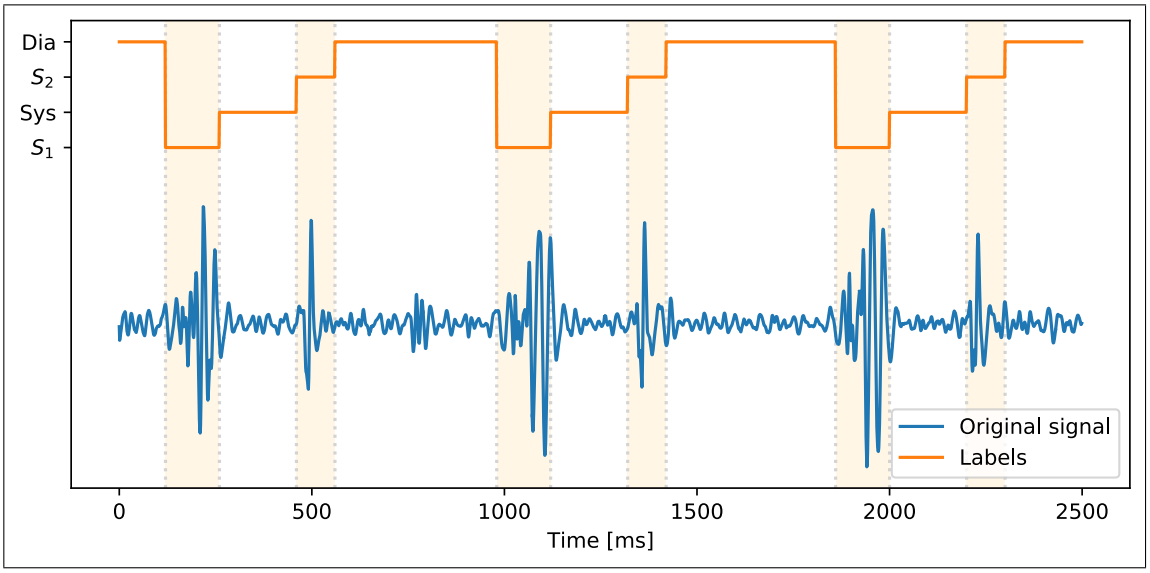

Figure 2.3: PCG and its labels. In this example, the labels are corrected to $f_s = 1000$ Hz.

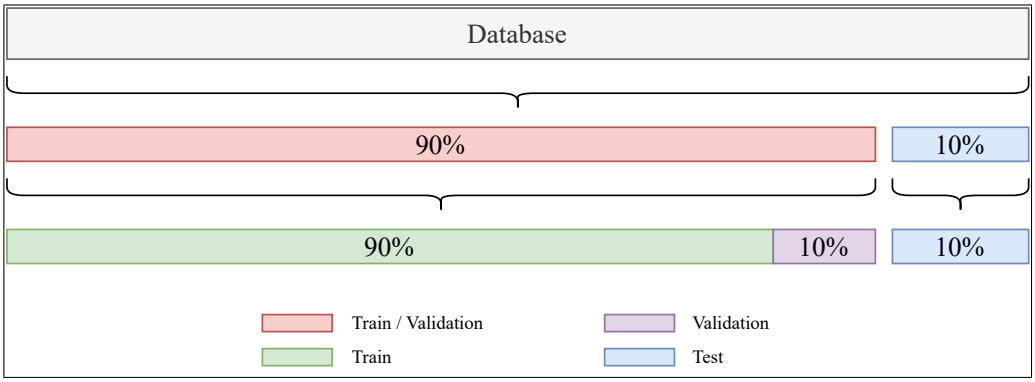

Figure 2.4: Database file division for network training and testing.

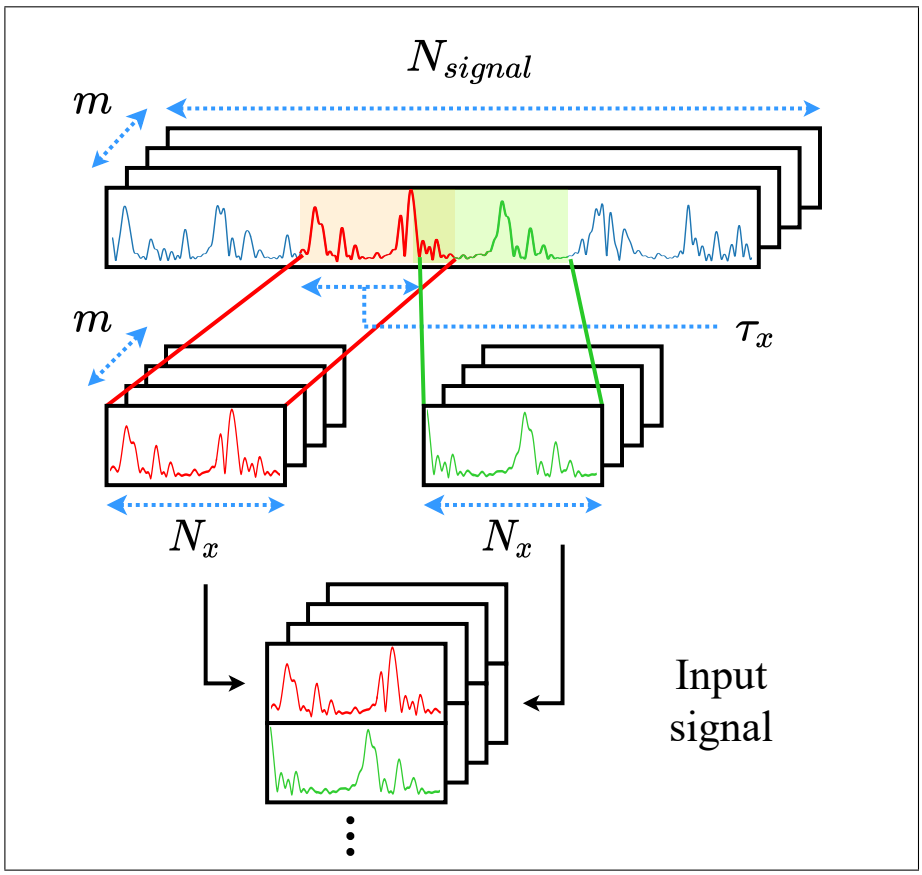

Figure 2.5: Windowing process of the network input signal. This figure shows two consecutive windows of length N_x, step size τ_x and m descriptors.

remaining is less than N_x, it will be zero padded to the desired length. Each of these windows will be grouped into a matrix that will constitute the input matrix of the network. The diagram of this process is illustrated in figure 2.5.

2.4 Implementation

For the network implementation, the system is divided into 3 main blocks connected consecutively: preprocessing, feature extraction and classification through the neural network in which the number of classes used will be varied, as can be seen in the diagram of figure 2.6. The implementation of each block and the alternatives proposed for each of them will be explained in detail below.

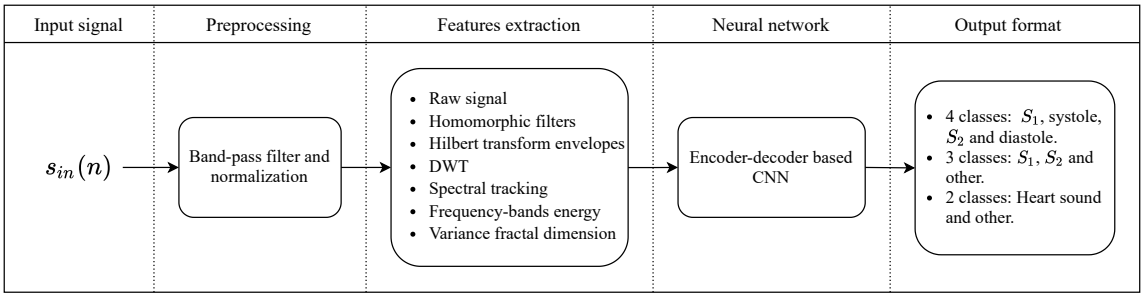

Figure 2.6: General scheme of the design of the heart sound detection system.

2.4.1 Preprocessing

Sometimes PCGs are contaminated by low-frequency sounds such as those caused by handling the stethoscope and muscle or pectoral vibrations, or high-frequency sounds such as murmurs. Also, the sound heard can vary by factors such as the stethoscope's capture position, instrument gain, age, gender, and physiology of the patient.

For this reason, in this stage, a bandpass FIR filter between 20-180 Hz will be used. Then, the filtered signal will be normalized to reduce the effect of the factors mentioned above, using the expression:

$$s_{norm}(n) = \frac{s(n)}{\max|s(n)|}, \quad \forall n \tag{2.4}$$

2.4.2 Feature settings

Several features are extracted and analyzed as possible candidates for the network, and there are different parameters that define its shape. Due to this, a preliminary study is performed that allows to find the parameters that achieve the best correlation between the envelope/feature and the heart sounds positions. Pearson correlation coefficient is calculated between the labels provided and the descriptor obtained, which is defined as:

$$\rho = \frac{E\left[(y(n) - \mu_y)(d(n) - \mu_d)\right]}{\sigma_y \sigma_d} \tag{2.5}$$

Where $y(n)$ is the binary sequence that indicates the position of S_1 and S_2 based on the labels provided; $d(n)$ corresponds to the envelope of interest to compare; and (μ_x, σ_x) corresponds to the mean and standard deviation of a signal $x(n)$ respectively. The Pearson coefficient satisfy that $\rho \in [-1, 1]$, where $\rho = 1$ indicates total positive correlation, while $\rho = -1$ indicates total negative correlation. This indicator will allow quantifying the similarity between the descriptor and the heart sounds positions given by $y(n)$, which is presented in figure 2.7.

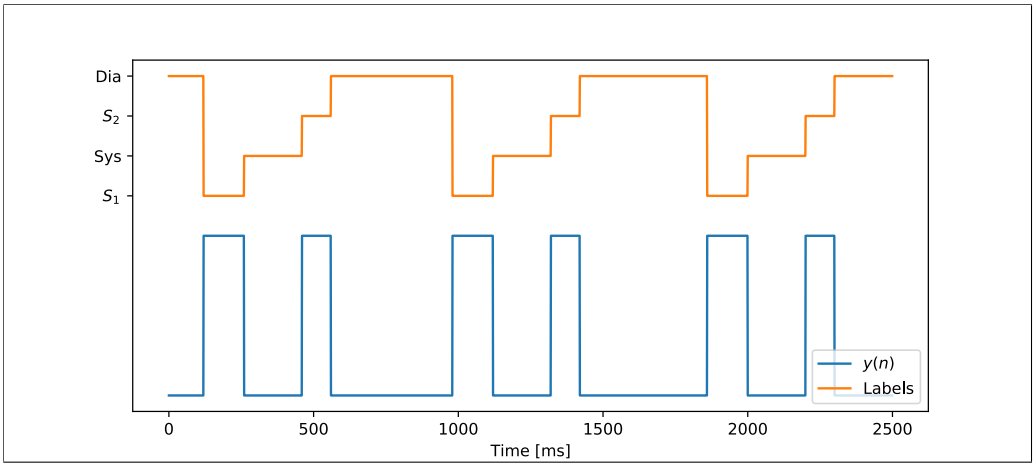

Figure 2.7: Graph of the binary sequence $y(n)$ for the heart sound positions.

Each of the calculated descriptors will be processed in such a way that all the information is contained in the range $0 \leq d(n) \leq 1$. Calling $\tilde{d}(n)$ to the raw descriptor obtained, the normalized descriptor $d(n)$ between 0 and 1 is defined as:

$$d(n) = \frac{\tilde{d}(n) - \min_n \tilde{d}(n)}{\max_n |\tilde{d}(n) - \min_n \tilde{d}(n)|} \tag{2.6}$$

Finally, in figure 2.8 an extract of a PCG together with each of the features chosen, whose description is shown below.

2.4.2.1 Homomorphic filters

To obtain the homomorphic filter envelope, the parameters of the low-pass filter applied to the logarithm of the signal must be defined. In this work, a FIR low-pass filter is used applying a Kaiser window with a cutoff frequency of 10 Hz, and a transition band of 5 Hz. This can be seen in plot a) of figure 2.8.

2.4.2.2 Hilbert envelopes

As input feature, the magnitude of the analytic signal derived from Hilbert transform will be used, which is presented in plot b) of the figure 2.8. Furthermore, a modified Hilbert envelope inspired by the work of (Varghees and Ramachandran, 2014) is proposed, which is summarized in the diagram of figure 2.9. First, the absolute value of the signal is obtained. Then a hard threshold stage using the 10% of the maximum value is applied. From this, the Shannon energy is calculated, which is defined as:

$$s_e(n) = -s(n)^2 \cdot \log(s(n)^2) \tag{2.7}$$

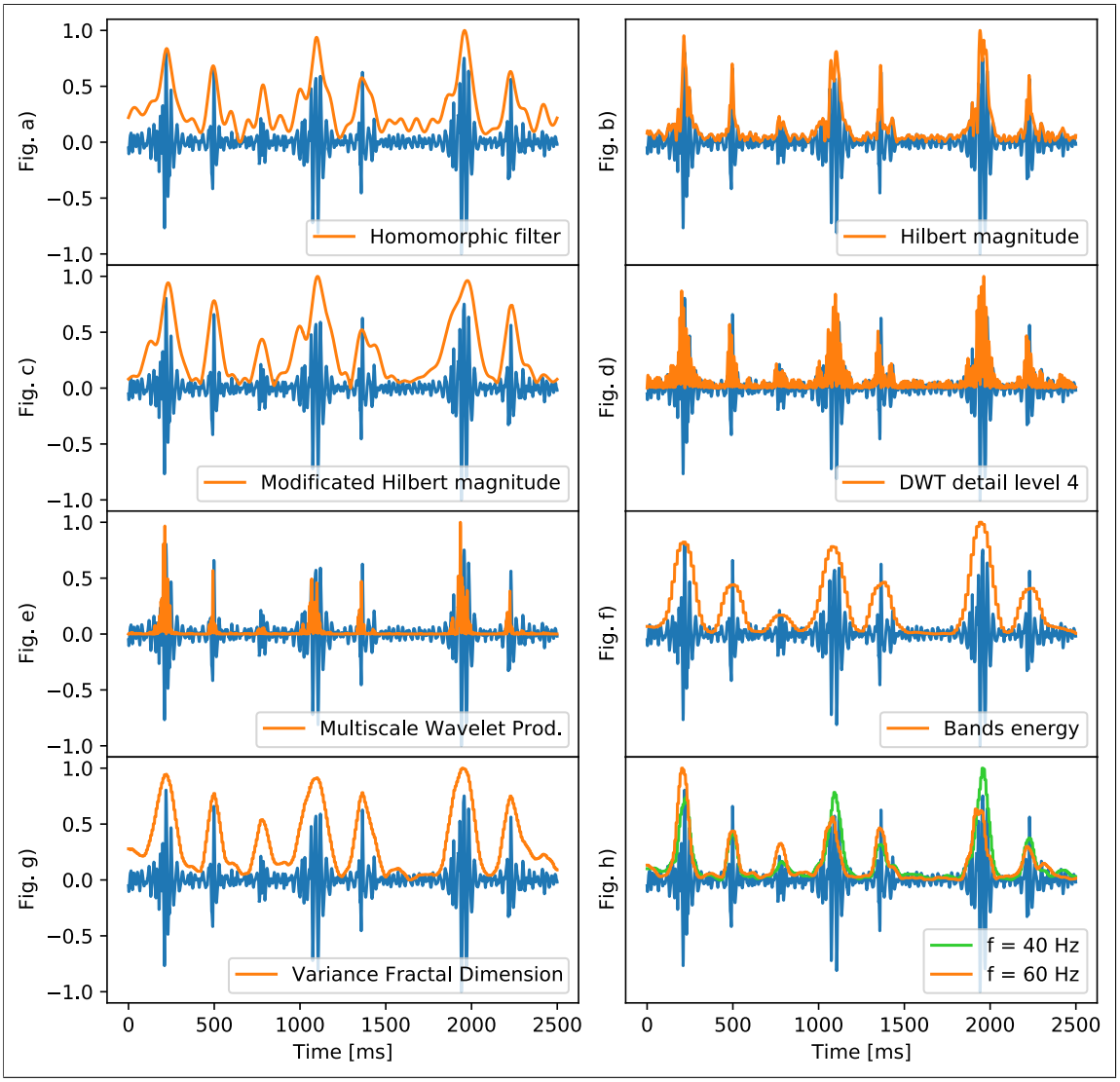

Figure 2.8: Features used in this work for a PCG extract. For each plot, the original signal and the corresponding feature indicated in the legend are attached.

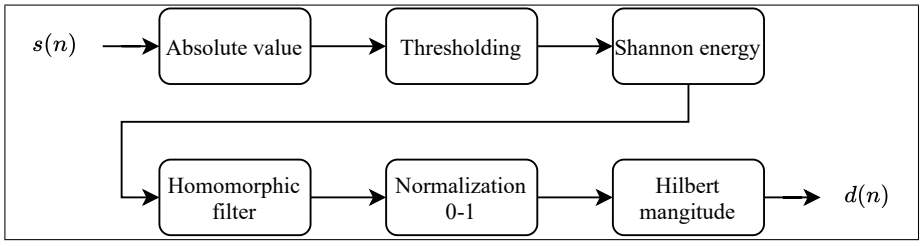

Figure 2.9: Step diagram to obtain the modified Hilbert envelope.

This allows to emphasize the medium intensity values of the signal and attenuate the effect of the low and high intensity values, which are associated with silences and clicks within the signal (ARI and SAHA, 2007; Liang et al., 1997). A homomorphic filter is applied to this result using the same parameters presented in section 2.4.2.1. Finally, the normalization between 0 and 1 presented in (2.6) and the Hilbert magnitude envelope is applied to obtain the result. The result of this process can be seen in plot c) of figure 2.8.

2.4.2.3 Wavelets envelopes

Considering the results presented in the literature (Castro et al., 2013; Huiying et al., 1997), in this work a *db6* will be used as the mother wavelet to obtain the magnitude of the 4th level detail coefficients (see details of the analysis in table C.1 available in the appendix) using the Stationary Wavelet Transform (SWT), as can be seen in plot d) of figure 2.8. This allows the length of the selected detail coefficient to be the same as the original signal.

Furthermore, the magnitude of a multi-scale Wavelet product is defined between the 4th and 5th detail level coefficients (see analysis detail in table C.2 available in the appendix), obtaining the envelope presented in plot e) of figure 2.8.

2.4.2.4 Frequency bands energy

For this feature the energy comprised in the 30-120 Hz band is calculated, using the magnitude of a spectrogram obtained with a Hann window with length $N_{wind} = 128$ points and overlap of $N_{step} = 16$ points (see details of the analysis in the table C.3 available in the appendix). The result can be seen in plot f) of figure 2.8.

2.4.2.5 Variance fractal dimension

To obtain this envelope, the algorithm proposed in works such as (Carvalho et al., 2005; J. Gnitecki and Z. Moussavi, 2003) is implemented. In the calculation, only a step size $k = 4$ was used, and the signal was divided into small windows of length $N_{wind} = 128$ points and step $N_{step} = 16$ points (see detail of the analysis in table C.4 available in the appendix). Once the VFD has been obtained, and normalizing it between 0 and 1 with (2.6), an additional step is added in which this envelope is inverted through the expression:

$$d(n) = 1 - FD_{\sigma_{norm}}(n) \tag{2.8}$$

Where $FD_{\sigma_{norm}}$ corresponds to the VFD normalized between 0 and 1. This is done because heart sound has a slightly more regular structure than white noise, and therefore its fractal dimension is smaller. Therefore, with the aim of highlighting the position of the heart sounds, this modification is performed, which result can be seen in plot g) of figure 2.8.

2.4.2.6 Spectral tracking

To obtain the spectral tracking envelope, the magnitude of a spectrogram obtained with a Hann window of length $N_{wind} = 128$ points and overlap of $N_{step} = 16$ points is used (see detail of the analysis in the table C.5 available in the appendix). The frequencies of interest to be tracked will be $f = 40$ Hz and $f = 60$ Hz. The result of the spectral tracking for both frequencies can be seen in plot h) of figure 2.8.

2.4.3 Envelopes resampling

Features such as VFD, spectral tracking, and frequency band energy have fewer points than the original signal. This is because they use techniques based on the windowing of the original signal, obtaining a single value from a window of N_{wind} points.

To recover the number of points from the original signal, a resampling process is proposed based on the diagram in the figure 2.10. As can be seen, for each windowed segment of a signal $s(n), n \in \{0, ..., N\}$ a feature $d(l)$ is obtained that returns a single point from a window of length N_{wind} points, where $l \in \{0, ..., L\}$ and $L < N$. For each point $d(l)$, it is repeated in such a way as to obtain a segment of length N_{wind} at the position of the original segment from which this value is obtained.

Once all the repeated segments have been obtained, a vertical sum of each segment is made in its corresponding position, obtaining a signal $x(n)$ with length $(N + 1)$ points. However, due to the fact that two or more segments are added in the overlap area (only two in the case of figure 2.10), it is necessary to correct them since the energy of the segments that provide information to that set of points is accumulating, which could generate a distortion of the feature. For this, it is necessary to know how many segments are overlapping for some point $n \in \{0, ..., N\}$ of the signal $x(n)$. With this information, a sequence $v(n)$ is made by making a vertical sum of the rectangular windows that describe each segment, which allows to show how much is the overlap for each point.

Finally, the resampled signal $r(n)$ is defined as:

$$r(n) = \frac{x(n)}{v(n)} \tag{2.9}$$

Where a element-wise division is made between both sequences, $\forall n = 0, ..., N$. It should be noticed that this resampling method generates a staircased signal whose segments have the length of the number of overlapping points, except for the final segment whose length is the window length minus the overlap points. This can be seen in figure 2.11.

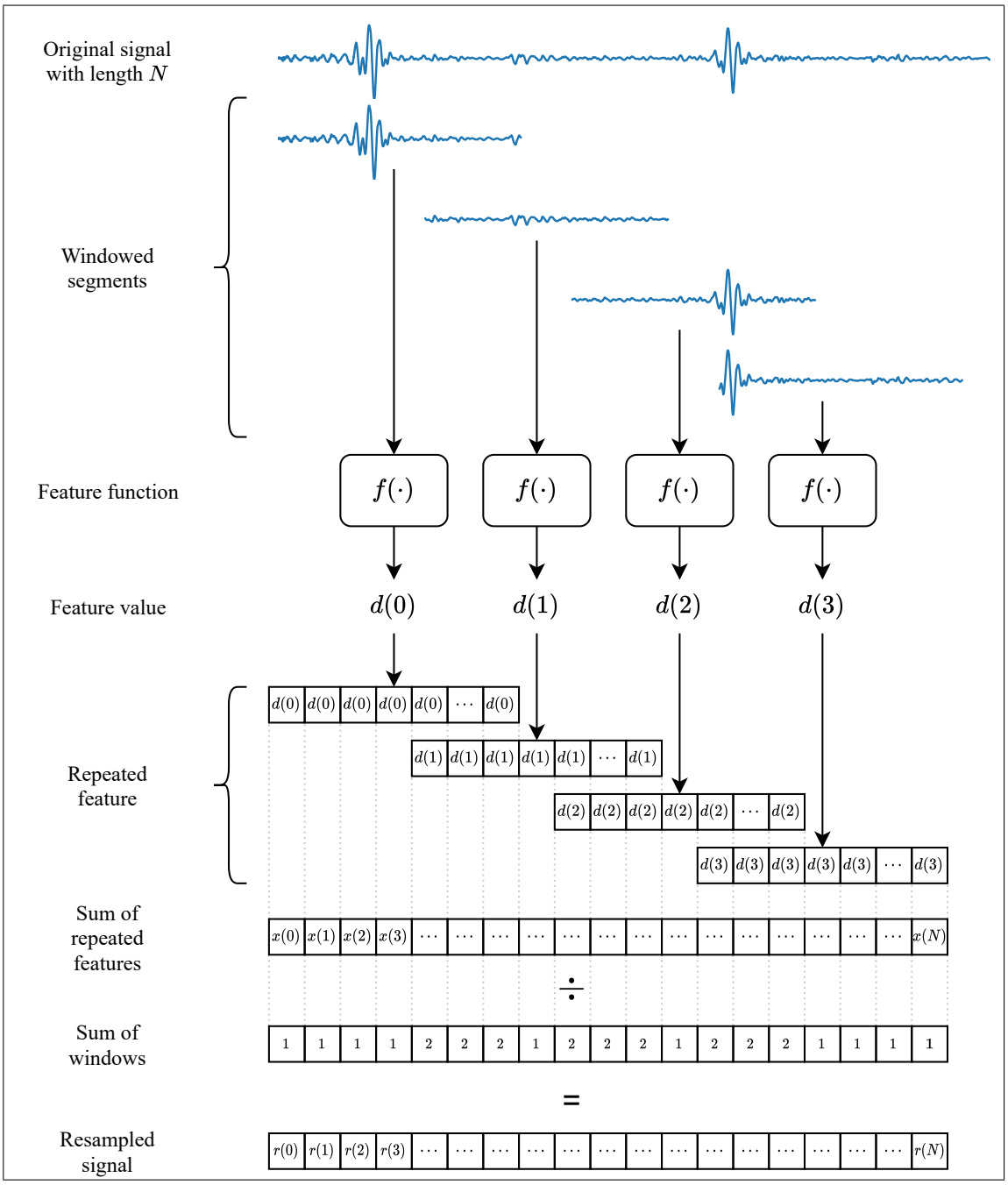

Figure 2.10: Diagram of the resampling process for features that apply windowing processes. In the case of the sum of the windows presented in the diagram, it should be noted that it is only an illustrative example, and it may change depending on the length of the window and the overlap. In this work, a rectangular window is used to re-scale the sum of the segments.

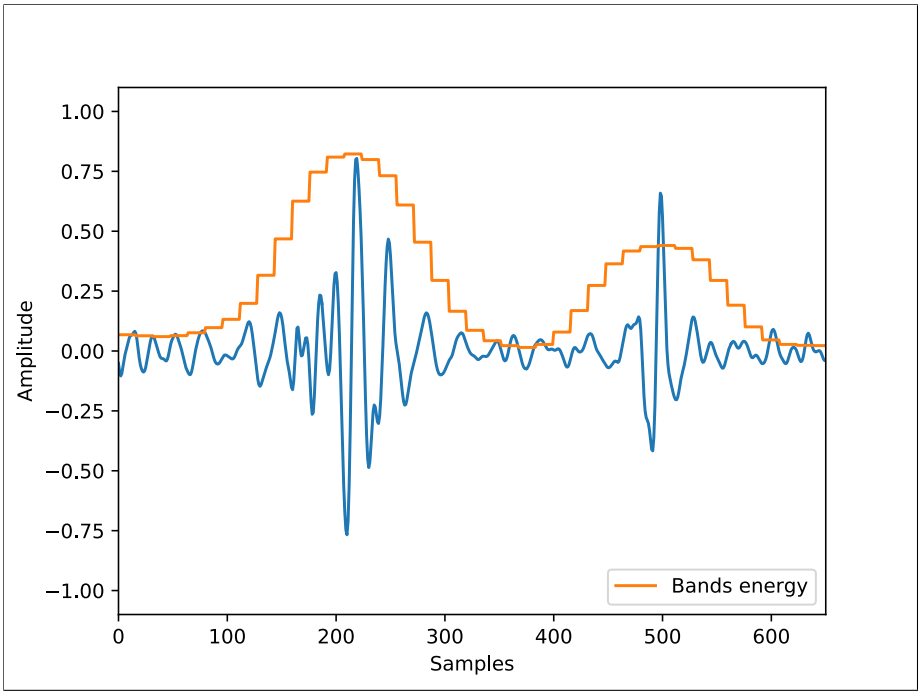

Figure 2.11: Frequency band energy envelope resampled. This example uses $N_{wind} = 128$ points and $N_{step} = 16$ points.

2.4.4 Initial network design

In this section, a base design and initial parameters will be proposed from which the variations of interest in the study will be made. Each of these networks was built using the Keras API built into Python's Tensorflow library (Abadi et al., 2016).

All the parameters of convolutional layers (defined by the `Conv1D` function of the `tensorflow.keras` environment) and perceptrons layers (defined by the `Dense` function of the `tensorflow.keras` environment) presented in this work will be initialized using the `'he_normal'` method described in (He et al., 2015).

To train the networks, the Adam optimization algorithm (Kingma and Ba, 2015) is used on a cross-entropy cost function. For all implementations of this work, $\beta_1 = 0.9$ (associated to momentum), $\beta_2 = 0.999$ (associated to RMSprop) and a learning rate of $\alpha = 0.001$ are used. Also, each network will be trained for 20 epochs using batches of 70 PCG segments with dimension (N_x, m), where N_x corresponds to the length of each segment and m to the number of features to be used.

Initially, in each convolutional layer, filters of length $H_l = 200$ will be defined. This definition is based on the fact that the duration of the heart sounds is between 100 and 150 ms. Therefore, to highlight the presence of a heart sound, it is necessary to use a filter that at least include its entire content. However, in section 2.5.4 results for other values of H_l will be studied.

The detail of the network based on the encoder-decoder architecture is shown in the figure 2.12. In the encoding stage 4 main blocks are used. The first 2 blocks consist in 2 consecutive convolution layers with a bank of $n_c^{[l]} = 13$ filters with length $H_l = 200$, batch normalization and ReLU activation; which are connected with a maxpooling layer that halves the number of points. The next 2 blocks use the same parameters as the first 2 blocks, but are constituted of 3 consecutive layers of convolution before maxpooling layer.

In the decoding stage they use 4 blocks that, correspondingly with each of the encoder blocks, carry out the reverse process up to the classification layer. The first 2 blocks are made up of an upsampling layer that replicates each sample twice at the entrance of this layer, generating a signal twice as long; which is connected with 3 consecutive layers of convolution with a bank of $n_c^{[l]} = 13$ filters with length $H_l = 200$, batch normalization and ReLU activation. The next 2 blocks use the same parameters, but are defined with 2 consecutive convolution layers after the upsampling layer instead of 3. In general, and for all variations of this type of network, it will be ensured that the decoder is symmetric with the encoder.

Finally, a softmax layer with $K = 3$ classes (S_1, S_2 and non-heart sound S_0) is used at the output. It should be noticed that all the convolutional filters implemented in this network also pad the signal so that the output maintains the length of the input, in such a way that the only stages that reduces/increases the dimension of the signal would be the maxpooling/upsampling layers.

2.4.5 Output modeling algorithm

Because in the initial design of the network presented in section 2.4.4 3 classes are used for the output of the network (S_0, S_1 and S_2), it is necessary to develop an algorithm that allows classifying the systolic and diastolic intervals from the samples identified as S_0.

In the first plot of the figure 2.13 an extract of the PCG is presented together with each of the 3 outputs of the network, indicating the probability of occurrence of each class. As mentioned in the section 2.2, for each point obtained, the class with the highest probability of occurrence is chosen, obtaining the output signal $y(n)$ presented in the second plot of the figure 2.13.

From this, an algorithm that uses the limits of the S_0 segments with S_1 and S_2 segments is proposed. Let us consider a non-heart sound segment S_0 whose left limit is at point n_L, while the right limit is at point n_R. From these points, the classification criterion of the non-heart sound segment is defined as:

$$y'(n) = \begin{cases} \text{Sys} & \text{if } y(n_L - 1) = S_1 \ \wedge \ y(n_R + 1) = S_2 \\ \text{Dia} & \text{if } y(n_L - 1) = S_2 \ \wedge \ y(n_R + 1) = S_1 \\ \text{Undef.} & \text{Otherwise} \end{cases} \tag{2.10}$$

Where $n = \{n_L, n_L + 1, ..., n_R - 1, n_R\}$ and $\wedge$ corresponds to the logical AND

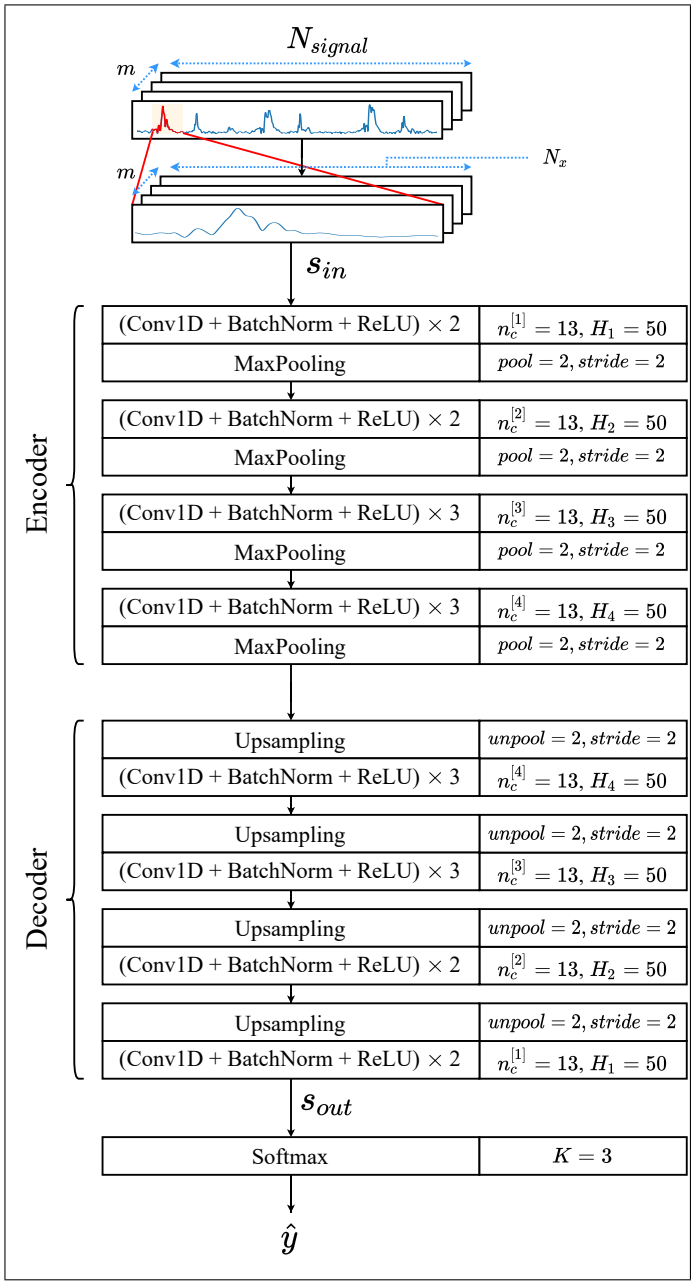

Figure 2.12: Proposed CNN based on encoder-decoder architectures. In this network, a encoder stage with successive convolutional and maxpooling layers is considered; and a decoder stage with successive upsampling and convolutional layers. The output of the last decoding layer is connected to a softmax layer.

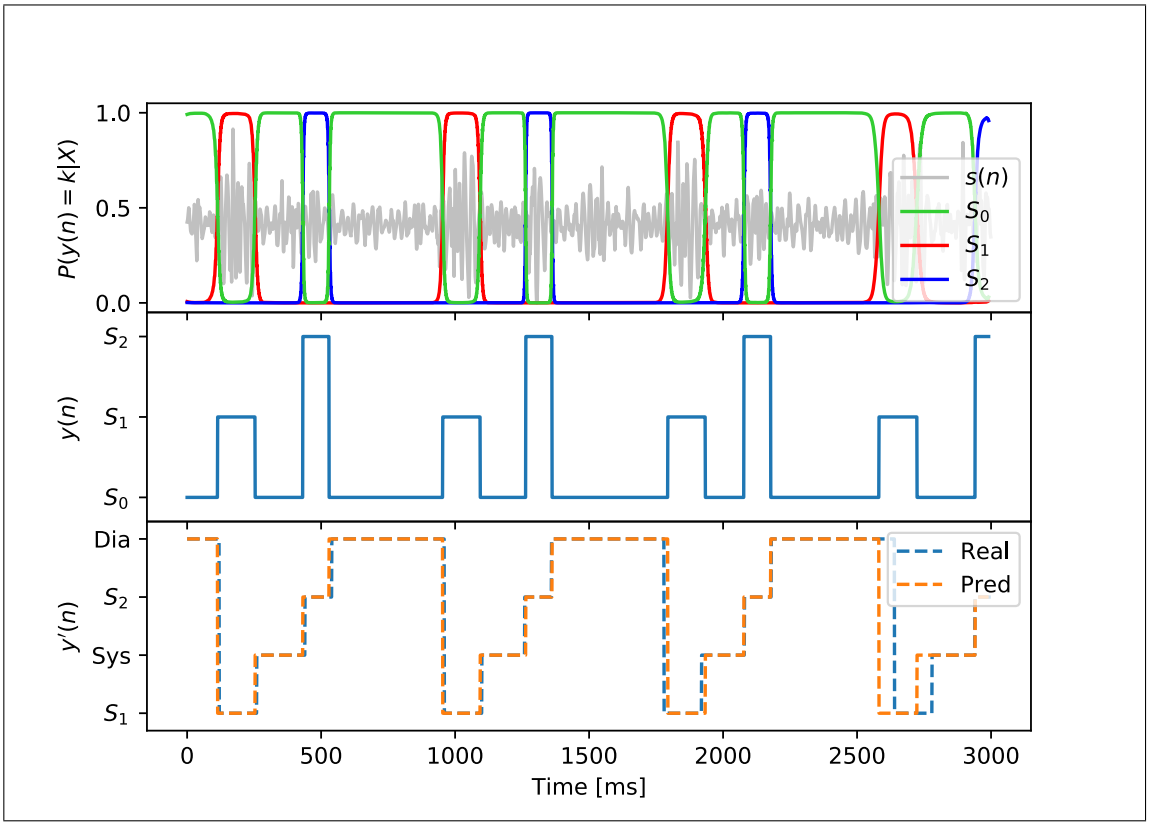

Figure 2.13: Output of networks defined with 3 classes and their equivalent representation of 4 classes.

operator. The result of this implementation can be seen in the third plot of the figure 2.13, where the segments identified as S_0 are classified as systole or diastole, while the heart sound segments S_1 or S_2 are kept with the same values.

2.4.6 Data augmentation

The noise injection is one of the typical methods used for data augmentation to the input features, which improves the robustness to random noise and reduces overfitting (Aggarwal, 2018a; Goodfellow et al., 2016).

In this work, Additive White Gaussian Noise (AWGN) will be added to the original PCG based on a Signal to Noise Ratio (SNR) defined in decibels. For this, the modified signal $\tilde{s}_{in}(n)$ is defined as:

$$\tilde{s}_{in}(n) = s_{in}(n) + \tilde{w}(n) \tag{2.11}$$

Where $s_{in}(n)$ corresponds to the original signal and $\tilde{w}(n)$ to the AWGN modified to meet the desired SNR specification (in decibels). Given a specification SNR_{db} for the ratio between the original signal and the noise in decibels, the desired energy of the noise signal is defined as:

$$E_{noise} = \frac{E_{signal}}{10^{(SNR_{db}/10)}} \tag{2.12}$$

Where $E_{signal} = \sum_n s_{in}(n)^2$. Therefore, defining $E_w = \sum_n w(n)^2$ where $w(n)$ is AWGN, the expression that obtains the AWGN $\tilde{w}(n)$ with the desired SNR_{db} is:

$$\tilde{w}(n) = \sqrt{\frac{E_{noise}}{E_w}} \cdot w(n) \tag{2.13}$$

2.5 Experiments and results analysis

As presented in the implementation section 2.4, there are a large number of parameters to vary for the system design. For this reason, an analysis based on a *ceteris paribus* of these parameters and elements that constitute the network will be performed in order to define parameter by parameter the options that improves its performance.

We will study in section 2.5.2 which features should be used in the input to obtain a better performance. In section 2.5.3 we will analyze the effect of increasing the number of filters as the network moves forward, while in section 2.5.4 a study of different filter length values is performed. Parameters such as network depth (section 2.5.5), length and step of the windows used on the input PCG segments of the network (section 2.5.6) will be analyzed. Finally, the performance of the network will be studied using different types of classes for the output of the network (section

2.5.7), culminating with a stability analysis through k-fold cross-validation (section 2.5.8). Additionally, the appendix presents an analysis of the implementation of class balancing based on the median and of two types of skipping connections that do not provide an improvement to the model, therefore they will be omitted in the experiments. In the section 2.5.1, the metrics used for each analysis to be performed will be described.

2.5.1 Performance metrics

In this work, the classical metrics will be used to evaluate the classification in a pattern recognition system (Powers, 2020): accuracy, precision, recall and F-score.

However, in this context of classification in multiple classes, a weighted macro average will be used to obtain a summary metric in the case of precision and recall (Pedregosa et al., 2011). From this, the performance metrics are defined as follow:

$$Accuracy = \frac{\sum_i T_{p_i}}{N_{samples}} \tag{2.14}$$

$$Precision = \frac{1}{\sum_i k_i} \sum_i k_i \frac{T_{p_i}}{T_{p_i} + F_{p_i}} \tag{2.15}$$

$$Recall = \frac{1}{\sum_i k_i} \sum_i k_i \frac{T_{p_i}}{T_{p_i} + F_{n_i}} \tag{2.16}$$

$$F_1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{2.17}$$

Where T_{p_i} correspond to the correct predictions for the class i, F_{p_i} and F_{n_i} to the false positives and negatives of the class i respectively, and k_i to the number of points labeled with the class i.

2.5.2 Features analysis

In this section, we will review different combinations of features in the input of the network defined in the previous section. To obtain these results, a windowing process of length $N_x = 1024$ and step $\tau_x = 64$ was used on the input signal. Moreover, the same parameters defined in section 2.4.4 are used for the encoder-decoder network. The results of this analysis are presented in table 2.2, and the details of the acronyms for each feature are presented in the table 2.3.

As can be seen, in general the results are quite homogeneous for different combinations of features. Regarding the validation and testing sets, the use of all the features detailed in section 2.4.2 returns a slightly better performance. In relation to the training set, the best performance is obtained using as input only the original

Table 2.2: Results of different combinations of features on the input of a CNN-based encoder-decoder architecture. For each of the metrics used, the combination of input features that obtains the best performance is highlighted in green. The first row corresponds to the base features combination, used in works such as (Renna et al., 2019; Springer et al., 2016).

CNN Encoder-Decoder ($N_x = 1024$, $\tau_x = 64$)																						
Features										Train				Validation				Test			
RS	HF	HT	HT'	DW	FE	ST 40	ST 60	MW	VF	Accuracy	Recall	Precision	F_1	Accuracy	Recall	Precision	F_1	Accuracy	Recall	Precision	F_1
	x	x		x	x					95.855	95.835	95.877	95.856	92.620	92.563	92.689	92.626	93.194	93.152	93.247	93.200
x										96.421	96.411	96.433	96.422	92.767	92.729	92.817	92.773	93.676	93.652	93.703	93.678
									x	95.134	95.100	95.171	95.136	92.004	91.915	92.103	92.009	92.687	92.622	92.766	92.694
			x						x	95.361	95.329	95.397	95.363	92.676	92.615	92.750	92.682	92.774	92.721	92.840	92.780
			x		x				x	95.272	95.240	95.304	95.272	92.809	92.749	92.878	92.813	92.695	92.643	92.755	92.699
			x		x	x			x	95.491	95.466	95.519	95.493	92.568	92.513	92.633	92.573	92.974	92.922	93.038	92.980
	x		x		x	x			x	95.761	95.739	95.783	95.761	92.898	92.852	92.954	92.903	93.182	93.131	93.245	93.188
	x		x		x	x	x		x	95.688	95.662	95.716	95.689	92.934	92.888	92.993	92.940	92.973	92.933	93.020	92.976
	x	x	x		x	x	x		x	95.786	95.766	95.808	95.787	92.484	92.425	92.555	92.490	93.141	93.100	93.193	93.147
	x	x	x	x	x	x	x		x	95.913	95.895	95.933	95.914	92.882	92.835	92.935	92.885	93.066	93.032	93.118	93.075
	x	x	x	x	x	x	x	x	x	95.953	95.935	95.973	95.954	92.845	92.781	92.915	92.848	93.288	93.246	93.339	93.292
x	x	x		x	x					95.935	95.919	95.952	95.936	92.993	92.948	93.051	92.999	93.200	93.156	93.254	93.205
x	x	x		x	x	x	x			96.172	96.153	96.191	96.172	92.989	92.921	93.066	92.994	93.506	93.475	93.547	93.511
x	x	x		x	x	x	x		x	95.902	95.886	95.919	95.903	92.867	92.824	92.922	92.873	93.411	93.379	93.451	93.415
x	x	x		x	x	x	x	x	x	95.858	95.842	95.875	95.858	92.884	92.833	92.953	92.893	93.490	93.460	93.523	93.491
x	x	x	x	x	x	x	x	x	x	96.072	96.056	96.088	96.072	93.070	93.041	93.105	93.073	93.745	93.720	93.778	93.749
x		x	x	x	x	x	x	x	x	95.705	95.683	95.728	95.705	92.953	92.905	93.009	92.957	93.333	93.303	93.372	93.337
		x	x	x	x	x	x	x	x	95.758	95.737	95.780	95.759	92.945	92.893	93.010	92.952	93.040	92.996	93.101	93.049
		x	x	x	x			x	x	95.960	95.942	95.979	95.960	92.950	92.896	93.012	92.954	93.504	93.470	93.551	93.511
		x		x	x			x	x	95.943	95.925	95.962	95.944	92.498	92.432	92.579	92.505	93.495	93.452	93.549	93.501
		x		x	x			x		95.949	95.932	95.967	95.949	92.678	92.619	92.745	92.682	93.460	93.419	93.511	93.465
			x	x	x			x		95.621	95.596	95.647	95.622	92.611	92.540	92.686	92.613	93.198	93.158	93.247	93.202
	x	x	x	x				x		95.853	95.830	95.877	95.854	92.503	92.432	92.596	92.514	92.820	92.772	92.878	92.825

Table 2.3: Detail of the abbreviations of the features presented in the table 2.2.

Feature	Abbreviation
Raw signal	RS
Homomorphic filters	HF
Classic Hilbert Transform	HT
Modified Hilbert Transform	HT'
Discrete Wavelet Transform	WT
Frequency band energy	FE
Spectral tracking	ST
Multi-scale Wavelet Product	MW
Variance Fractal Dimension	VF

signal. This result is remarkable since it reveals that the encoder-decoder architecture is robust against different combinations of descriptors at the input of the network, and that even just entering the raw signal would be enough to obtain a considerable performance in heart sounds detection. Due to the latter, it is possible to interpret that the network is capable of extracting, by itself and efficiently, the relevant properties of the PCG that allow characterizing each segment of interest.

However, the results of using all the features on the training set are quite close to the best performance obtained in this set when using only the raw signal at input. In view of the results, and from this point on, we will work with all the features presented in section 2.4.2 for the input of this network. With this, it is expected that the network will assign the importance that each feature should have in the final decision through the back-propagation process.

2.5.3 Number of filters analysis

So far we have used a fixed number of filters $n_c^{[l]}$ for each convolutional layer. For this reason, in this section we will study the increase in the number of filters as we go deeper into the network. This idea is based on the behavior of classical CNN architectures, where as one advances over the layers of the network, the dimension of the signal is decreased, but the number of filters is increased (Krizhevsky et al., 2012; Yann LeCun et al., 1998). To perform this experiment, 3 network designs were considered, varying the number of filters for a given constant N_c.

The first uses a constant number of filters N_c for the entire network, varying between $N_c = \{5, 10, 15, 20, 30, 50\}$. The second consists of a linear increase in the number of filters as each convolutional layer l is deepen into the network $(l \cdot N_c)$, using the same values of N_c as the first network. The third applies an exponential increase in the number of filters as the network deeps $((N_c)^l)$, varying between $N_c = \{3, 4\}$. The results of this experiment are presented in table 2.4.

As can be seen in all types of variations, as the number of filters increases, the metrics over the training set go up. This could indicate that the more filters this network has, the more capacity it will have to fit the training data. However, this causes an undesirable overfitting. Respecting to validation and testing sets, it is possible to notice that the results in general are stable for any type of network, independently of the number of filters to be used in each layer. Even so, networks with more filters tend to fail more in validation set.

According to the results, it is possible to conclude that the increase of the filters as one deeps over the network layers does not offer significant improvements, either in the linear or exponential configuration. Therefore, and in order to work with a more compact network that adequately adjusts to the problem, it is decided to work with the constant filter network with $N_c = 15$ (Id 3). From this point on, this constant number of filters will be used for the rest of the experiments.

Table 2.4: Results of the increase in the number of filters as one deeps over a CNNs-based encoder-decoder architecture. For each of the metrics used, the combination of input features that obtains the best performance is highlighted in green.

CNN Encoder-Decoder ($N_x = 1024$, $\tau_x = 64$)													
$n_c^{[l]}$	N_c	Trainable parameters	Train				Validation				Test			
			Accuracy	Recall	Precision	F_1	Accuracy	Recall	Precision	F_1	Accuracy	Recall	Precision	F_1
N_c	5	105318	94.635	94.574	94.698	94.636	93.081	92.967	93.209	93.088	93.360	93.278	93.451	93.364
	10	400633	95.598	95.577	95.620	95.599	92.365	92.295	92.451	92.373	93.268	93.228	93.324	93.276
	15	885948	96.197	96.182	96.214	96.198	93.208	93.175	93.250	93.212	93.433	93.410	93.465	93.437
	20	1561263	96.597	96.588	96.606	96.597	93.001	92.976	93.037	93.007	93.569	93.552	93.590	93.571
	30	3481893	97.188	97.183	97.193	97.188	92.933	92.898	92.977	92.937	93.402	93.378	93.431	93.405
	50	9603153	97.794	97.791	97.796	97.794	92.749	92.727	92.778	92.752	93.531	93.519	93.548	93.534
$l \cdot N_c$	5	840828	95.776	95.753	95.799	95.776	93.122	93.050	93.203	93.127	93.604	93.562	93.653	93.608
	10	3341653	96.632	96.624	96.640	96.632	93.097	93.066	93.140	93.103	93.536	93.507	93.573	93.540
	15	7502478	97.123	97.119	97.127	97.123	93.076	93.052	93.107	93.080	93.275	93.259	93.299	93.279
	20	13323303	97.441	97.438	97.445	97.441	92.866	92.844	92.896	92.870	93.179	93.166	93.200	93.183
	30	29944953	98.141	98.139	98.143	98.141	92.912	92.896	92.936	92.916	93.519	93.508	93.532	93.520
	50	83038253	98.478	98.477	98.479	98.478	92.819	92.807	92.836	92.822	93.303	93.297	93.312	93.305
$(N_c)^l$	3	8171100	96.193	96.179	96.207	96.193	92.694	92.645	92.750	92.698	93.190	93.159	93.230	93.195
	4	75918815	97.000	96.993	97.007	97.000	92.454	92.420	92.495	92.458	93.282	93.261	93.310	93.285

Table 2.5: Results of the analysis of the length of the filters as one deeps over a CNNs-based encoder-decoder architecture. For each of the metrics used, the combination of input features that obtains the best performance is highlighted in green.

CNN Encoder-Decoder ($N_x = 1024$, $\tau_x = 64$, $n_c^{[l]} = 15$)														
$H^{[l]}$	H_i	Trainable parameters	Train				Validation				Test			
			Accuracy	Recall	Precision	F_1	Accuracy	Recall	Precision	F_1	Accuracy	Recall	Precision	F_1
$\frac{H_i}{2^l}$	400	713448	96.109	96.095	96.125	96.110	93.032	92.993	93.082	93.038	93.130	93.098	93.172	93.135
	200	357198	96.252	96.237	96.268	96.252	92.840	92.797	92.895	92.846	93.447	93.413	93.492	93.453
	150	266448	96.190	96.170	96.211	96.191	91.701	91.627	91.794	91.711	92.904	92.856	92.969	92.912
	100	178398	96.201	96.174	96.229	96.202	92.226	92.164	92.303	92.233	92.753	92.703	92.822	92.763
	50	88998	96.612	96.569	96.660	96.614	91.488	91.357	91.649	91.502	91.886	91.781	92.017	91.899
H_i	400	1770948	96.000	95.985	96.018	96.001	92.703	92.661	92.757	92.709	93.094	93.060	93.138	93.099
	200	885948	96.221	96.206	96.236	96.221	93.132	93.084	93.188	93.136	93.558	93.532	93.593	93.563
	150	664698	96.262	96.248	96.277	96.263	93.252	93.217	93.296	93.256	93.534	93.500	93.578	93.539
	100	443448	96.475	96.466	96.484	96.475	93.028	92.995	93.073	93.034	93.622	93.598	93.659	93.628
	50	222198	96.583	96.573	96.593	96.583	93.159	93.130	93.198	93.164	93.372	93.345	93.407	93.376

2.5.4 Filter length analysis

In this section the filter length for each of the convolutional layers will be analyzed. For this experiment, two options for the behavior of the filters will be considered as one deeps over the layers of the network using a constant H_i presented in table 2.5.

The first option maintains the filter lengths H_i in every layer ($H^{[l]} = H_i$), varying for $H_i = \{50, 100, 150, 200, 400\}$. The second option decreases the length of the filters in half ($H^{[l]} = H_i/(2^l)$) each time a maxpooling layer is used (and not after each convolutional layer as experienced in the 2.5.3 section). This option also varies between the same values of H_i mentioned. The results of this experiment are shown in table 2.5.

As presented in the table 2.5, the option that best fits the training set are those that use filters with constant $H_i = 50$ and that decreases after each maxpooling layer. However, it is also this option that presents the worst results in the validation

Table 2.6: Results of the analysis of the number of convolutional layers (depth) used in CNNs-based encoder-decoder architecture. For each of the metrics used, the combination of input features that obtains the best performance is highlighted in green. The first row corresponds to the base architecture obtained from the previous analyses.

CNN Encoder-Decoder ($N_x = 1024$, $\tau_x = 64$, $n_c^{[l]} = 15$, $H^{[l]} = 150$)													
Encoder layers	**Trainable parameters**	**Train**				**Validation**				**Test**			
		Accuracy	**Recall**	**Precision**	F_1	**Accuracy**	**Recall**	**Precision**	F_1	**Accuracy**	**Recall**	**Precision**	F_1
[2,2,3,3]	664698	96.262	96.248	96.277	96.263	93.252	93.217	93.296	93.256	93.534	93.500	93.578	93.539
[2,3]	326748	96.569	96.557	96.582	96.569	92.813	92.770	92.868	92.819	93.211	93.179	93.250	93.214
[2,2,3]	461928	96.517	96.508	96.528	96.518	93.184	93.155	93.224	93.190	93.614	93.591	93.648	93.619
[2,3,3]	529518	96.536	96.529	96.544	96.537	93.013	92.973	93.062	93.017	93.363	93.340	93.391	93.365
[2,2,2,3]	597108	96.475	96.466	96.485	96.475	92.908	92.870	92.960	92.915	93.620	93.595	93.648	93.621
[2,3,3,3]	732288	96.321	96.308	96.334	96.321	92.886	92.843	92.939	92.891	93.578	93.552	93.613	93.582
[2,2,2,3,3]	799878	96.154	96.140	96.168	96.154	92.735	92.669	92.809	92.739	93.257	93.221	93.305	93.263
[2,2,3,3,3]	867468	95.699	95.677	95.722	95.700	93.063	93.028	93.101	93.064	93.318	93.287	93.355	93.321
[2,2,2,3,3,3]	1002648	95.728	95.706	95.751	95.729	93.524	93.473	93.583	93.528	93.677	93.643	93.718	93.681
[2,2,2,2,3,3]	968853	94.411	94.357	94.463	94.410	92.574	92.456	92.696	92.576	93.281	93.203	93.368	93.286
[2,2,3,3,3,3]	1070238	95.668	95.644	95.692	95.668	92.549	92.443	92.673	92.557	93.298	93.235	93.370	93.302

and testing sets, so it is concluded that this alternative overfit the network with the training data.

In general, for all the scenarios, it is also possible to notice that the network fits the training data quite well, with performances very close to the maximum in this set. However, for most cases, the option that offers the best results considering the validation and testing sets is the one that uses a constant length of the filters.

From this results, it is chosen to use the option with constant $H_i = 150$ since it offers the best results in the validation set. Furthermore, in relation to the test set, this option has a performance quite close to the option that reaches the maximum.

2.5.5 Network depth analysis

In this section we will analyze what is the number of layers that the network should contain. To do this, consider the diagram of figure 2.1. As can be seen in the encoder network, in the first 2 blocks and before each maxpooling layer, 2 convolutional layers are used with batch normalization and ReLU activation. In the same way, in the next 2 blocks and before the maxpooling layers, 3 of these composite layers are used. For the decoder network the same specifications of the layers are used but in reverse. In this analysis we will use the notation [2,2,3,3] to describe the encoder network just mentioned, where each of the numbers indicates the number of convolutional layers before each maxpooling layer. It should be noticed that the length of this sequence is directly related to the number of maxpoolings layers present in the network. The table 2.6 presents the different experiments performed together with their results.

As can be seen, in relation to the training set the results are quite homogeneous. Nevertheless, it is possible to notice that as the depth in the network increases, the fitting that the network makes on the data decreases. In contrast, when the network

Table 2.7: Results of the analysis of the training windows length N_x and the step τ_x for the inputs of the CNN-based encoder-decoder architecture. For each of the metrics used, the combination of input features that obtains the best performance is highlighted in green. The first row corresponds to the base architecture obtained from the previous analyses.

CNN Encoder-Decoder ($n_c^{[l]} = 15$, $H^{[l]} = 150$, $c^{[l]} = [2, 2, 3, 3]$)													
τ_x	N_x	Train				Validation				Test			
		Accuracy	Recall	Precision	F_1	Accuracy	Recall	Precision	F_1	Accuracy	Recall	Precision	F_1
64	1024	96.262	96.248	96.277	96.263	93.252	93.217	93.296	93.256	93.534	93.500	93.578	93.539
	256	91.273	90.641	91.911	91.272	89.445	88.530	90.426	89.468	90.109	89.343	90.936	90.132
	512	94.313	94.182	94.455	94.318	90.533	90.290	90.825	90.557	91.816	91.644	92.024	91.834
	2048	97.289	97.283	97.296	97.289	94.090	94.069	94.115	94.092	93.953	93.933	93.979	93.956
	4096	98.656	98.655	98.657	98.656	94.396	94.392	94.402	94.397	94.263	94.259	94.268	94.264
	8192	98.805	98.804	98.806	98.805	94.766	94.763	94.769	94.766	93.870	93.866	93.878	93.872
	16384	98.048	98.045	98.052	98.048	95.254	95.243	95.269	95.256	93.771	93.762	93.783	93.773
128	256	90.596	89.849	91.359	90.598	88.550	87.421	89.710	88.551	89.730	88.744	90.677	89.700
	512	93.521	93.319	93.734	93.526	91.620	91.410	91.867	91.638	92.063	91.904	92.267	92.085
	1024	95.673	95.648	95.698	95.673	92.668	92.607	92.749	92.678	93.166	93.128	93.212	93.170
	2048	96.957	96.953	96.961	96.957	93.900	93.886	93.918	93.902	93.860	93.842	93.883	93.862
	4096	97.798	97.795	97.801	97.798	94.558	94.550	94.569	94.559	94.493	94.485	94.503	94.494
	8192	98.623	98.621	98.624	98.623	95.029	95.026	95.034	95.030	93.917	93.909	93.930	93.919
	16384	98.885	98.884	98.886	98.885	95.567	95.563	95.570	95.566	94.050	94.046	94.058	94.052

is shallower, it better fits the training data, obtaining the best performance in the shorter network (Id 1).

On the other hand, in relation to the validation and testing sets, it is possible to see that the results are even more homogeneous than in the training set. The network that presents the best performance in these sets is the network with Id 8. However, one of the risks of using networks with so many maxpooling layers is the loss of signal resolution at the encoder output due to the reduction of points. Since the depth of the network does not give significantly better results compared to the base case (Id 0), the number of convolutional layers initially presented will be preserved.

2.5.6 Analysis of the length and step of the training windows

In this section we will study the effect of defining the length N_x and the step τ_x on the windowing process over the PCG to train the network. For this, networks will be trained for each combination of parameters $N_x = \{256, 512, 1024, 2048, 4096, 8192, 16384\}$ and $\tau_x = \{64, 128\}$. The results of the experiments are presented in table 2.7.

As can be seen from the results, as the size N_x of the window increases, the performance for all sets improves. This may be due to the fact that the system is trained with segments that have a broader context than with the length of the window used so far ($N_x = 1024$), allowing the system to characterize each PCG event in a better way.

Table 2.8: Results of the analysis of the number of labels to be used in the CNN-based encoder-decoder architecture. For each of the metrics used, the combination of input features that obtains the best performance is highlighted in green. The first row corresponds to the base architecture obtained from the previous analyses.

CNN Encoder-Decoder ($N_x = 16384$, $\tau_x = 128$, $n_c^{[l]} = 15$, $H^{[l]} = 150$, $c^{[l]} = [2, 2, 3, 3]$)												
Labels number	**Train**				**Validation**				**Test**			
	Accuracy	**Recall**	**Precision**	F_1	**Accuracy**	**Recall**	**Precision**	F_1	**Accuracy**	**Recall**	**Precision**	F_1
3	98.885	98.884	98.886	98.885	95.567	95.563	95.570	95.566	94.050	94.046	94.058	94.052
2	97.256	97.275	95.966	96.616	94.988	94.948	92.229	93.569	93.747	93.694	91.965	92.822
4	97.010	98.317	97.145	97.727	91.970	94.660	92.031	93.327	92.253	93.944	92.331	93.131

In relation to the step size τ_x of the windows, it is possible to notice that in general for lower values of this parameter, the performance is better in the training and validation sets. Meanwhile, in the test set there is no clear trend. However, the best performances for all sets are found at $\tau_x = 128$. This is an expected result since at a lower value of τ_x it is possible to have a greater number of windows to train the network, which is a form of data augmentation. Due to this, there is a regularizing effect on the network that causes performance to decrease for the network generalization that this implies.

In view of these results, and considering the computational cost of increasing the size of the training window at the network input, it was decided to use a length $N_x = 16384$ for the training window and a step of $\tau_x = 128$ points.

2.5.7 Number of labels analysis

So far it has been experimented using 3 classes to train the networks: S_1, S_2 and non-heart sound (S_0). However, in most works such as (Messner et al., 2018; Oliveira et al., 2019; Renna et al., 2019; S. E. Schmidt et al., 2010; Springer et al., 2016) 4 classes are used to train the network: S_1, systole, S_2 and diastole. Another variant less studied in the literature is the use of only 2 classes that indicate the detection of cardiac activity in the PCG: heart sound (S_1 or S_2) and non-heart sound (S_0). An example of these class definitions is presented in figure 2.14. In this section we will study which of these options allows better performance of the CNN with encoder-decoder architecture. The results are presented in table 2.8.

As can be seen, by train the network using 3 classes (S_1, S_2 and S_0) it is possible to obtain a better performance for the network. This result can have several interpretations. First of all, it is possible to notice that when using only 2 classes there is no improvement of the results in any of the sets, which suggests that the definition of the sounds S_1 and S_2 under a single label does not work correctly. This may be because the characteristics of the sounds S_1 and S_2 are different enough not to be generalized under the same label. Despite this, it is desirable that training with 3 classes present better results than with 2 classes, since with this latter system

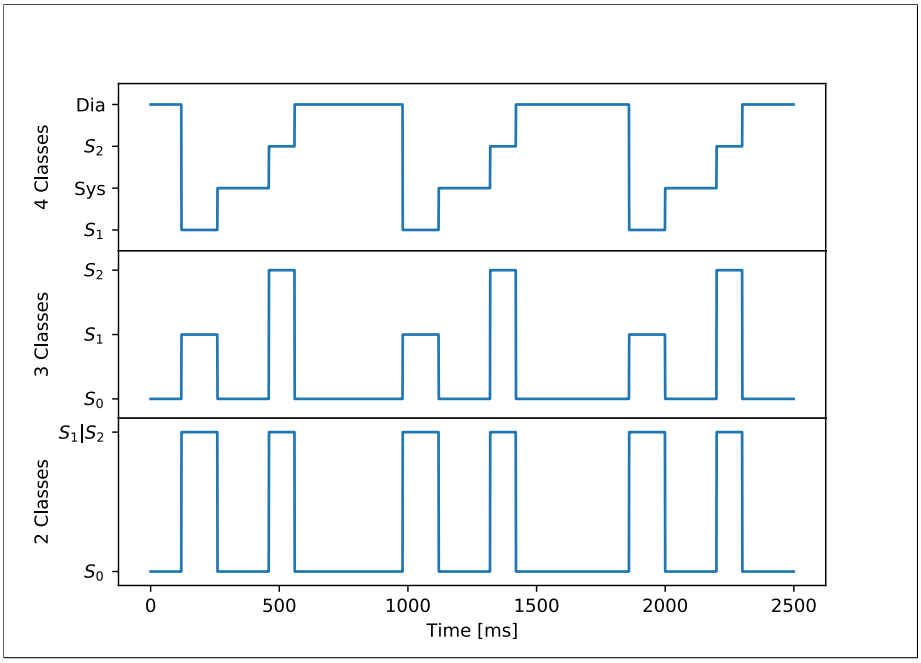

Figure 2.14: Variations in the number of classes used for the design of the encoder-decoder network.

it would not be possible to identify the systolic and diastolic intervals. This is not convenient in applications that require the study of each phase of the cardiac cycle to make a diagnosis because it will must be designed an additional system that classify each segment of the cardiac cycle.

Respect to the use of the 4 classes, there is no improvement in network performance either. This may be due to the fact that the systolic and diastolic intervals do not differ significantly in terms of their frequency characteristics, rather they resemble each other in the absence of heart sounds. Although by defining the 4 classes it is possible to easily obtain the detail of each phase of the PCG, by using 3 classes it is possible to do the same without loss of information. Indeed, the intervals defined between S_1 and S_2 would correspond to systole, while the intervals between S_2 and S_1 to diastole. Therefore, there is no need to define them explicitly to recognize all phases of a cardiac cycle.

Considering the points exposed in this analysis, it is decided to use 3 classes for the design of the system.

2.5.8 Network stability analysis

Finally, and in order to study how robust the implemented system is, k-fold cross-validation will be used dividing the database into $k = 10$ groups, where the records of each patient are only located in a single group. This makes it possible to ensure that

Table 2.9: Results of k-fold cross-validation with $k = 10$ for the CNN-based encoder-decoder architecture configured from the previous steps.

CNN Encoder-Decoder ($N_x = 16384$, $\tau_x = 128$, $n_c^{[l]} = 15$, $H^{[l]} = 150$, $c^{[l]} = [2, 2, 3, 3]$)								
k	**Train**				**Test**			
	Accuracy	**Recall**	**Precision**	F_1	**Accuracy**	**Recall**	**Precision**	F_1
1	98.632	98.631	98.634	98.632	93.546	93.535	93.557	93.546
2	98.645	98.643	98.647	98.645	94.630	94.625	94.635	94.630
3	98.252	98.248	98.255	98.252	91.468	91.433	91.508	91.471
4	98.704	98.703	98.706	98.704	91.643	91.607	91.679	91.643
5	98.536	98.534	98.539	98.536	92.547	92.492	92.623	92.558
6	98.724	98.722	98.726	98.724	92.016	91.946	92.101	92.023
7	98.869	98.867	98.871	98.869	91.981	91.901	92.044	91.972
8	98.708	98.707	98.711	98.709	94.490	94.485	94.497	94.491
9	98.733	98.731	98.735	98.733	91.814	91.782	91.855	91.818
10	98.632	98.631	98.634	98.632	93.546	93.535	93.557	93.546
μ	98.644	98.642	98.646	98.644	92.768	92.734	92.806	92.770
σ	0.163	0.163	0.162	0.163	1.190	1.209	1.169	1.189

the PCGs of any patient are not found simultaneously in the training and testing set. In addition, Gaussian white noise injection with $SNR = \{5$ dB, 1 dB, 0 dB, -1 dB$\}$ will be used for data augmentation and network regularization. Unlike the analyzes performed thus far, a validation set will not be used in this experiment. The table 2.9 presents the result of the k-fold cross-validation with $k = 10$ for the definitive encoder-decoder network.

From the presented results, it is possible to notice that for the different scenarios the final setting of the network on the training sets is quite stable. In summary, the accuracy is 98.644 $\pm$ 0.163%, the recall is 98.642 $\pm$ 0.163%, the precision is 98.646 $\pm$ 0.162% and the F-score is 98.644 $\pm$ 0.163%. Regarding the different test sets, it is possible to appreciate a little more variability in the results, obtaining an accuracy of 92.768 $\pm$ 1.190%, a recall of 92.734 $\pm$ 1.209%, a precision of 92.806 $\pm$ 1.1690% and a F-score of 92.770 $\pm$ 1.189%. The increase in variability in the test sets may be due to the fact that, in general, the training set does not vary considerably between the different iterations. Indeed, for each iteration only one tenth of this set is modified, so the network has the possibility of adjusting to a dataset that does not vary radically. On the other hand, it is to be expected that the test sets have a greater variability since they do not correspond to records or patients that the system knows in advance. However, although these results present less stability, they still offer a fairly high performance and a robustness of approximately $\pm 1\%$ for the different metrics.

This result is consistent with the results obtained in the previous sections, where

it has been seen that for different parameter variations, the performance of the network in general remains stable above 90% for the metrics of interest. A notable result is that the difference in network performance over the validation and testing sets remains relatively constant, varying on average by values of less than 1% for the different scenarios proposed. This could indicate that the network adequately generalize the characteristics of the heart sound, or that the heart sounds are signals that are sufficiently generalizable to be independent of the patient to be studied. In any case, it is recommended to divide the patients in different sets to have a clear notion of the behavior of the network in an applied context.

From this, it is possible to conclude that the CNN-based encoder-decoder architecture achieves to fit robustly on the auscultated signals, and to adequately generalize the relevant characteristics that allow determining the fundamental heart sounds.

2.6 Conclusions

One of the main problems in the analysis of respiratory and heart sounds is the mutual interference that they generate, which tend to mask and alter certain characteristics of interest and that could generate errors when making a diagnosis (Mondal, P. Bhattacharya, et al., 2013; Mubarak et al., 2018).

In this work, a study of a CNN based on an encoder-decoder architecture to solve the problem of heart sound segmentation was presented. The results allow us to conclude that it has a good performance, in addition to being considerably robust. Furthermore, qualitatively, it presents desirable properties since it allows the input signal to be classified point-wise, without having to reduce a segment to a single label (as networks with classical architectures do).

From the features analysis used in the network input, it is possible to appreciate that for different scenarios the results do not vary significantly. The use of features implemented in other types of heart sound segmentation methods such as variance fractal dimension, spectral tracking and the multi-scale product of Wavelets is proposed, making no significant improvement in its performance either. Even when using only the raw signal at the input of the network, the results do not vary considerably. This indicates that the network is powerful enough to extract the key features of the signal to classify each segment in the best possible way.

With respect to the parameters of the filters in CNNs, the best combination of number and length of the filters in each layer are $N_c = 15$ and $H^{[l]} = 150$ respectively. Meanwhile, the most balanced option for network depth is to use 2 blocks of 2 convolutional filters terminated with maxpooling followed by 2 blocks with 3 convolutional filters, also terminated in a maxpooling layer. The choice of this depth is based on the fact that the increase in the number of maxpooling layers does not offer improvements in the performance of the network, in contrast to the increase in

trainable parameters and the possible resolution problems that successive layers of maxpooling generates. Regarding the length N_x and step τ_x, the results indicated that the higher the N_x, the better the results are systematically obtained. Regarding step τ_x a conclusive pattern is not found. Therefore, a window length $N_x = 16384$ is chosen and step $\tau_x = 128$ is chosen for the training of the networks. Other techniques used in neural network training and design such as class balancing and skipping connections (which is a comparison with the work of (Renna et al., 2019)) did not offer a significant improvement to justify their use, therefore were discarded.

A parameter of interest in this work was the definition of the number of classes that the network is recognizing. In literature, typically 4 classes are used: S_1, systole, S_2 and diastole. However, in this work the network behavior was studied using 3 classes (S_1, S_2 and non-heart sound S_0) and 2 classes (heart sound or non-heart sound). The results indicated that using 3 classes the best performance is obtained, which is convenient since it is still possible to recognize the systolic and diastolic intervals knowing the positions of S_1 and S_2.

Finally, a stability analysis was performed from a k-fold cross-validation with $k = 10$ folds over the database used, ensuring that each patient is only in one fold. Because in general the results do not vary much, it is concluded that the network based on encoder-decoder is robust on the variation of parameters. An important result of this work is that, despite the fact that the validation set has patients from the training set, the performance of the network does not differ much compared to the test set whose patients are not in the training set. With this, it is concluded that this type of CNN is well suited for solving the heart sound segmentation problem.

Among the future works that could be evaluated is the use of a larger heart sound database such as that of (C. Liu et al., 2016), whose labels on the heart sound segments have been manually corrected by health professionals.

This kind of systems could be useful for the diagnosis of heart diseases since it would allow knowing the positions of the heart sounds and, therefore, the position and duration of the systolic and diastolic intervals. With this information, the presence of abnormal sounds such as heart murmurs in any of these intervals could be studied, or the duration of each interval could be studied, which could reveal the presence of some heart disease. Furthermore, they could provide relevant information for the detection of respiratory diseases, since it would allow the use of source separation techniques focused on the heart sound segments, keeping the respiratory sound of the auscultated signal intact. These contributions could be useful in manual diagnoses, and based on the results obtained from the Physionet challenge on 2016, in a performance improvement in design of automatic systems for the detection of cardiac pathologies.

Chapter 3

Source separation

Since the heart and lung are in very close areas of the body, it is inevitable that the recorded heart and lung sounds will interfere with each other in time and frequency (Noman Q. Al-Naggar and Al-Udyni, 2018; Canadas-Quesada et al., 2017; Kattepur et al., 2010; Lin and Hasting, 2013). For the heart sounds analysis, the aim is to reduce respiratory sound as much as possible, considering it as signal noise. In this case, the lung sounds can be attenuated by changing the position of the stethoscope (Ahlstrom et al., 2005), and even asking the patient to hold their breath so that the recorded sound would be purely cardiac. But in the case of the analysis of respiratory sounds, the heart sound introduces pseudo periodicities persistently and impossible to stop (Hossain and Zahra Moussavi, 2003; J. Hadjileontiadis and M. Panas, 1998; Lin and Hasting, 2013; Pourazad, Z. Moussavi, and Thomas, 2006).

For this reason, the separation between heart and lung sounds is of great interest to specialists in both areas (cardiologists and pulmonologists) since it will allow more precise diagnoses (Kattepur et al., 2010; A. K. Khan et al., 2010; Mondal, P. Banerjee, et al., 2017; Mondal, P. S. Bhattacharya, et al., 2011; Z. Wang et al., 2015). However, one of the main difficulties presented by this problem is that heart and lung sounds are not only overlapped in time, but also in frequency (Canadas-Quesada et al., 2017; J. Hadjileontiadis and M. Panas, 1998). Therefore, it is necessary to develop an algorithm that allows to recover each signal while maintaining its characteristic properties, avoiding the loss of information in the process. This chapter will focus on the recovery of respiratory sound from signals auscultated in the thorax with a strong cardiac component. This will allow computer science researchers to design better classifiers that automatically diagnose some type of respiratory disease (Kandaswamy et al., 2004; Reichert et al., 2008).

Non-Negative Matrix Factorization (NMF) is an analysis technique that, due to its desirable properties (see chapter D.3 available in appendix for more details), has taken strength in areas of source separation applied in text mining, chemical spectroscopy, image processing, bioinformatics, finance, and of course in audio signals (Févotte, Vincent, et al., 2018; Z. Y. Zhang, 2012). In this chapter, three decompo-

sition methods using NMF are proposed to recover the lung sound that, unlike the typical methods with NMF presented in the literature, use the heart sounds position in the auscultated signal. The first method performs the decomposition on each of the heart sound positions. The second method uses a masking step over all the heart sounds positions in the signal, which is the input to the decomposition process. The third method uses the information of the components obtained to reconstruct the lung sound based on the segments free of heart sound from the original signal. The main objective of these proposals is to not distort the segments in which the heart sounds are not found with the NMF decomposition.

In order to classify the components obtained, this work proposes 3 assignment criteria inspired by the work of (Canadas-Quesada et al., 2017). The first criterion is based on the spectral distribution of each component. The second is based on the spectral correlation between the components and the segments free from heart sound (which provide information on the lung sound). In relation to this criterion, in (Canadas-Quesada et al., 2017) the author uses an external database to characterize the heart sound. Our proposal has the advantage that information contained in the same signal is used, so it is not necessary to use an additional database. The third criterion is based on the correlation between the detected cardiac activity and the temporal characteristics of each component.

Furthermore, unlike (Canadas-Quesada et al., 2017), adaptive thresholds are proposed that take into account the nature of the data, using statistical descriptors like the mean for their calculation. Components assignment performance will be studied using each criterion individually.

In this chapter, one of the articles prepared in this research and sent for publication entitled "*Source separation for single channel thoracic cardio-respiratory sounds applying Non-negative Matrix Factorization (NMF) using the heart sound positions*" will be presented. In the same way as in the previous chapter, the introduction of this article will be omitted since it corresponds to information mentioned in the chapter 1.

The rest of this chapter is organized as follows. In the section 3.1 a review of the works implemented in the literature to separate these sounds is presented. Section 3.2 presents the mathematical foundation of NMF, used for source separation. In section 3.3 the database used in the experiments is detailed. In section 3.4 the base method and the three proposed ones methods are explained, and the assignment criteria of the components obtained is also explained. In section 3.5 the results are reported. Finally, section 3.6 presents the conclusion of this chapter.

It should be noted that in the chapter B in appendix it is possible to find some additional results. And in the chapter D it is possible to find more information about the NMF interpretation, properties and solution algorithms.

3.1 Related works

To date, different approaches have been made to address this problem. The first methods involved high-pass or band-pass filters, using cut-off frequencies close to 70-100 Hz to attenuate the heart sound and recover the respiratory sound (Vannuccini et al., 2000). Nevertheless, the elimination of certain low-frequency bands, especially below 200 Hz, causes the loss of relevant information of the respiratory sound that is spectrally overlapped with the heart sound, so it is not a desirable solution (Ahlstrom et al., 2005; Noam Gavriely et al., 1995; Hossain and Zahra Moussavi, 2003; J. Hadjileontiadis and M. Panas, 1998; Tsalaile et al., 2008).

A widely used method is the linear adaptive filters to reduce the presence of heart sounds. They are generally modeled as a problem of least mean squares (LMS) (Noman Qaid Al-Naggar, 2013; Iyer et al., 1986), normalized least mean squares (NLMS) (Noman Q. Al-Naggar and Al-Udyni, 2018) or recursive least squares (RLS) (Potdar et al., 2012) using an adaptive noise cancellation scheme (ANC). In (Yip and Y. T. Zhang, 2001) the preprocessing with Automatic Gain Control (AGC) of a Laplacian electrocardiogram (LECG) as a reference signal is proposed. In (Nersisson and Noel, 2017) an adaptive filter step estimation algorithm based on a Nelder-Mead optimization model is developed. In (Noman Qaid Al-Naggar, 2013) the input signal is used by applying a band-pass filter and adding Gaussian white noise to simulate the reference signal. However, adaptive filters do not completely remove heart sound due to their non-stationary nature which makes time alignment between the primary and reference signals difficult to apply (January Gnitecki and Z. M. Moussavi, 2007; Shah, Koch, et al., 2015; Shah and C. B. Papadias, 2013). Furthermore, it is not always possible to obtain an adequate reference signal that allows achieving separation (Mondal, P. Banerjee, et al., 2017; Z. Wang et al., 2015). In some cases, an electrocardiogram (ECG) (Hadjileontiadis, 1997; Yip and Y. T. Zhang, 2001) is used as the reference signal, which may not always be available. Also, if the reference signal is not well defined, it may decrease the separation performance.

In (Charleston and Azimi-Sadjadi, 1996) it is proposed to use a Reduced Order Kalman Filter (ROKF) to reduce the presence of heart sound. However, to develop this method 3 assumptions are used that are not necessarily correct (Pourazad, Z. Moussavi, and Thomas, 2006). Indeed, it assumes that the interaction between lung and heart sounds is additive, which is not necessarily correct since sounds can be mixed in even more complex ways due to the medium they are transmitted. It also assumes that the signals are mutually uncorrelated processes. But the increase of physical activity can cause both to increase proportionally, and by being transmitted through the same medium, the mutual interaction of both sounds can be affected. Although they are quite reasonable assumptions, they can make the method not adequate enough when separating sources (Pourazad, Z. Moussavi, and Thomas, 2006).

Wavelets denoising is also a widely used method of eliminating lung sounds from

heart sounds (Ali et al., 2017; Hossain and Zahra Moussavi, 2003; J. Hadjileontiadis and M. Panas, 1998; Messer et al., 2001). In (Messer et al., 2001) a study of the parameters in the denoising process is carried out, applying an averaging step that improves the Gaussian noise reduction. In (Várady, 2001) the authors try to remove ambient noise incorporating a microphone that records these sounds. In (Hadjileontiadis, 2005) an approximation with fractal dimension is used that allows conditioning even speech signals. In (Omari and Bereksi-Reguig, 2015) a criterion that allows selecting the best mother Wavelet is proposed. In (Mondal, Saxena, et al., 2018) an approach with Wavelet Packet Transform and denoising using info of the eigenvalues obtained by SVD is presented. Nevertheless, these methods do not work well in situations where the patient has murmurs (Omari and Bereksi-Reguig, 2015) or there is a large presence of unwanted noise (Mondal, Saxena, et al., 2018). Also, depending on the database there are mother Wavelets, thresholds and decomposition levels that are better adapted, generating an additional challenge in real scenarios (Shah, Koch, et al., 2015; Shah and C. B. Papadias, 2013).

Methods have also been implemented using time-frequency representations in order to recognize the segments where the heart sound is found to eliminate that segment completely and to reconstruct it based on the adjacent information, which should correspond to a pure lung sound. In (Ahlstrom et al., 2005) the Poincaré recurrence, Takens' theorem and a trajectory algorithm are used to achieve this. In (Pourazad, Z. Moussavi, and Thomas, 2006) a heart sound detection scheme is used based on the Continuous Wavelet Transform (CWT) with an adaptive threshold, band-pass filters to eliminate the segments of the spectrogram where it is detected the heart, and 2D interpolation algorithms to reconstruct the attenuated band. In (Flores-Tapia et al., 2007) a Multiscale Wavelet Product based on the Stationary Wavelet Transform (SWT) is used to find the heart sound, and a reconstruction using linear auto-regressive (AR) or moving average (MA) prediction models.

Another approach that was not so successful was the modeling of systems representing heart and lung sound. In (Hong Wang and L. Y. Wang, 2003) and later in (Hong Wang, L. Y. Wang, et al., 2004) it is intended to remove ambient noise. In the work of (Zheng et al., 2007) the heart and lung sound separation is also added, using 3 stethoscopes to measure these sounds. However, this kind of approach presents several difficulties. One of them is the identification of each system. An excitation is needed for each system in order to correctly model each block, which cannot always be done (Hong Wang and L. Y. Wang, 2003). Furthermore, if the systems have significant cardio-respiratory rhythm or ambient sound variations, a real-time estimation of the systems would be required, adding further difficulty to the method.

Empirical Mode Decomposition (EMD) is a fairly revised approach in which the signal is typically decomposed into oscillatory basis called Intrinsic Mode Functions (IMFs) that are derived from the signal itself. In (Mondal, P. S. Bhattacharya, et al., 2011) decomposition in IMFs is used to recognize heart and respiratory sounds, which are processed until the presence of heart sounds is eliminated. In (Mondal,

P. Banerjee, et al., 2017) a similar algorithm is proposed, but in the sum of IMFs that represents the lung sound, an FFT-based prediction algorithm is applied to reconstruct these gaps. In (Z. Wang et al., 2015) a separation method using Adaptive Fourier Decomposition (AFD), a different type of EMD, is presented. Nevertheless, the signal decomposition in IMFs does not have a mathematical foundation, which makes difficult to describe its physical meaning (Z. Wang et al., 2015).

Independent Component Analysis (ICA) is one of the Blind Source Separation (BSS) methods used to address this problem, and is a typical solution to the cocktail party problem. In (Pourazad, Z. Moussavi, Farahmand, et al., 2005) ICA is implemented using as input the spectrogram of the signal. In (Chien et al., 2006) and (Ayari et al., 2012) the signal in time domain is used as input. In (A. K. Khan et al., 2010), the frequency domain is used as input, together with other techniques such as Direction of Arrival (DOA) and Beamforming. In general, ICA does not provide satisfactory results. In (Pourazad, Z. Moussavi, Farahmand, et al., 2005) it is mentioned that spectrogram-based ICA can decrease heart sound, but not completely remove it. Furthermore, this method assumes that the heart and lung sounds are independent. This is not necessarily true since due to the medium through which it propagates (where reflections and delays are generated), the recorded sound is correlated and the mix can even have a convolutional nature (Pourazad, Z. Moussavi, Farahmand, et al., 2005). Finally, one of the technical limitations of this method is that it is necessary to have more than one observation simultaneously to implement it. However, most modern digital stethoscopes allow you to record an observation on a single channel (Shah and C. B. Papadias, 2013).

NMF is another type of BSS approach that uses a time-frequency representation of the cardio-respiratory signal. In(Lin and Hasting, 2013) the use of NMF is proposed using information from the Constant-Q Transform (CQT) of the signal for the construction of a binary mask. In (Shah and C. B. Papadias, 2013) a staged semi-supervised NMF problem is proposed, in which initially an estimated heart sound decomposition is obtained, then the cardio-respiratory sound and finally the cardio-respiratory sound with noise. In works such as (Canadas-Quesada et al., 2017; Shah and C. B. Papadias, 2013) clustering criteria is proposed to classify each obtained component. One of the main difficulties with NMF is that there is no a systematic method to determine the quantity of components that optimizes the separation process. However, one of its main advantages is that the components obtained by this method are not only suitable for solving the temporal and spectral overlap problem, but also that its results are physically interpretable as additive components of the original signal (Févotte, Vincent, et al., 2018; Z. Y. Zhang, 2012).

3.2 Theoretical background

3.2.1 Non-negative Matrix Factorization

NMF decomposes a non negative matrix $\mathbf{X} \in \mathbb{R}^{N \times M}$ into two non-negative matrices, $\mathbf{W} \in \mathbb{R}^{N \times K}$ and $\mathbf{H} \in \mathbb{R}^{K \times M}$, where K is the number of components in which the original signal will be separated and is called rank of the factorization (Févotte, Vincent, et al., 2018). Typically this range K is chosen such that $(N+M)K \leq NM$ (Canadas-Quesada et al., 2017; Févotte and Idier, 2011; Ganesh R. Naik, 2016; Lin and Hasting, 2013). This decomposition can be stated as (Lin and Hasting, 2013):

$$\mathbf{X} = \mathbf{WH} + \mathbf{E} \tag{3.1}$$

Where $\mathbf{E}$ is the error term between $\mathbf{X}$ and the estimated product $\mathbf{WH}$. In practice, to obtain the matrices $\mathbf{W}$, $\mathbf{H}$ the following optimization problem is defined: (Févotte, Vincent, et al., 2018):

$$\begin{aligned} \mathbf{W}, \mathbf{H} &= \min_{\mathbf{W},\mathbf{H}} D(\mathbf{X} \mid \mathbf{WH}), \\ &\text{s.t.} \mathbf{W}, \mathbf{H} \geq 0 \end{aligned} \tag{3.2}$$

Where the objective function $D(\mathbf{X}|\mathbf{WH})$ is called divergence, which is a measure of dissimilarity between $\mathbf{X}$ and $\mathbf{WH}$ (Z. Y. Zhang, 2012).[1] This is a separable function such that:

$$D(\mathbf{X}|\mathbf{WH}) = \sum_{n=1}^{N} \sum_{m=1}^{M} d([\mathbf{X}_{nm}] \mid [\mathbf{WH}_{nm}]) \tag{3.3}$$

Where $d(x|y)$ is a continuous scalar function over x and y, where $d(x|y) \geq 0\,,\ \forall x, y \geq 0$ and $d(x|y) = 0$ if and only if $x = y$ (Essid and Ozerov, 2014; Févotte, Vincent, et al., 2018). One of the most used divergence functions is the β-divergence, which is defined as (Essid and Ozerov, 2014; Févotte and Idier, 2011; Févotte, Vincent, et al., 2018):

$$d_\beta(x \mid y) \triangleq \begin{cases} \frac{1}{\beta(\beta-1)} \left(x^\beta + (\beta-1)y^\beta - \beta x y^{\beta-1}\right), & \beta \in \mathbb{R} \backslash \{0,1\} \\ \frac{x}{y} - \log\left(\frac{x}{y}\right) - 1 = d_{IS}(x \mid y), & \beta = 0 \\ x \log\left(\frac{x}{y}\right) - x + y = d_{KL}(x \mid y), & \beta = 1 \\ \frac{1}{2}||x-y||^2 = d_Q(x \mid y), & \beta = 2 \end{cases} \tag{3.4}$$

In general, for audio NMF applications is common to use a spectrogram magnitude ($\mathbf{X} = |\mathbf{S}(t,f)|$) from a signal $s_{in}(n)$, which by definition is non-negative (Bryan and D. Sun, 2013; Essid and Ozerov, 2014; Févotte, Vincent, et al., 2018). A typical solving algorithm for the NMF problem is the multiplicative update given by (Févotte, Bertin, et al., 2009; Févotte and Idier, 2011; Févotte, Vincent, et al., 2018):

$$\mathbf{H} \leftarrow \mathbf{H} \odot \frac{\mathbf{W}^T \left((\mathbf{WH})^{\odot[\beta-2]} \odot \mathbf{X}\right)}{\mathbf{W}^T (\mathbf{WH})^{\odot[\beta-1]}} \tag{3.5}$$

[1] The notation $\mathbf{A} \geq 0$ indicates that each entry of the matrix $a_{ij} \geq 0, \forall i, j$

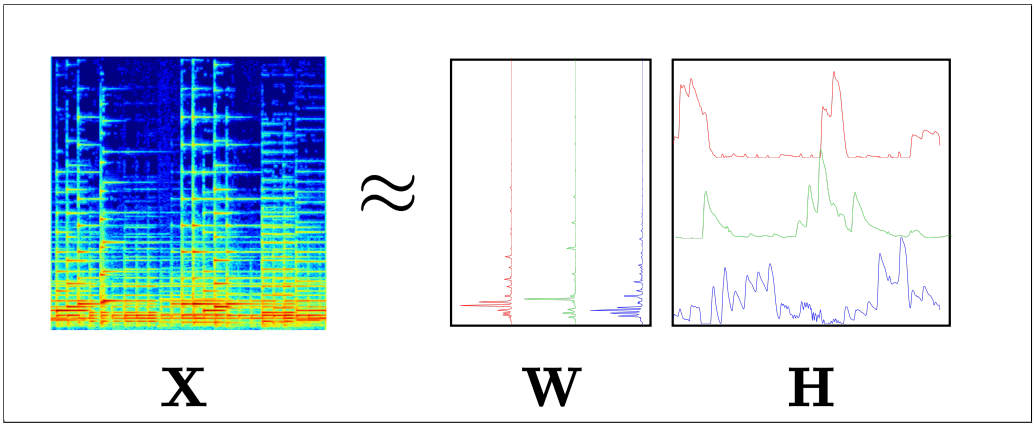

Figure 3.1: NMF decomposition example. In this case, a 6-seconds segment of the song Feeling Good (version by British band Muse) played on piano with $K = 3$ was decomposed. As can be seen, in each column of $\mathbf{W}$ we have the spectral pattern of each component, while in each row of $\mathbf{H}$ we have the activation patterns in time. Each component has a different color. Inspired by the graphics of (Févotte, Vincent, et al., 2018).

$$\mathbf{W} \leftarrow \mathbf{W} \odot \frac{\left((\mathbf{WH})^{\odot[\beta-2]} \odot \mathbf{X}\right)\mathbf{H}^T}{(\mathbf{WH})^{\odot[\beta-1]}\mathbf{H}^T} \tag{3.6}$$

Where $\odot$ denotes element-wise multiplication and $\frac{A}{B}$ element-wise division. On the one hand, the matrix $\mathbf{W} = [w_1, w_2, \ldots, w_K]$ can be interpreted as a dictionary of recurring patterns, represented by each column. On the other hand, $\mathbf{H} = [h_1, h_2, \ldots, h_K]^T$ can be understood as a matrix containing the activation or coding coefficients of the bases of $\mathbf{W}$ through the time, represented by each row (Févotte, Vincent, et al., 2018; Ganesh R. Naik, 2016).

When matrix $\mathbf{X}$ is an spectrogram magnitude, each column of $\mathbf{W}$ is the magnitude of the characteristic frequency spectrum of each component, while each row of $\mathbf{H}$ will indicate the weight of each spectrum recognized in the matrix $\mathbf{W}$ through time. An example of this can be seen in the figure 3.1 where an audio record is decomposed into $K = 3$ components.

From this, the definition of a component $\mathbf{X}_i \in \mathbb{R}^{N \times M}$ from the NMF decomposition of a spectrogram magnitude $\mathbf{X}$ is given by:

$$\mathbf{X}_i = w_i h_i^T \tag{3.7}$$

And:

$$\mathbf{WH} = \sum_{i=1}^{K} \mathbf{X}_i \tag{3.8}$$

However, in general this optimization problem does not reach the global minimum, so $\mathbf{E} \neq 0$ and $\mathbf{X} \approx \mathbf{WH}$. Therefore, the sum of the components does not return the original matrix. To correct this problem, a useful solution is the construction of masks that allow filtering the original matrix from the information present in each component.

The Wiener filter is a filter that allows to create a mask $\mathbf{M}_i$ from the relative information provided by the component $\mathbf{X}_i$ over the total of components, and is definied as (Bryan and D. Sun, 2013; Canadas-Quesada et al., 2017; Essid and Ozerov, 2014; Févotte, Vincent, et al., 2018):

$$\mathbf{M}_i = \frac{\mathbf{X}_i}{\mathbf{WH}} = \frac{w_i h_i^T}{\sum_{i=1}^{K} w_i h_i^T} \tag{3.9}$$

Where the division between matrices is element-wise. As can be seen in (3.9) this mask behaves like a template that indicates the weight in each entry (n, m) of the matrix to be masked. Indeed, the Wiener filter distributes the energy proportionally over each component, where each proportion indicates the association between the amplitude of the component $\mathbf{X}_i$ given by (3.7) and the amplitude of the multiplication $\mathbf{WH}$.

Note that $\sum_k \mathbf{M}_i = 1 \in \mathbb{R}^{N \times M}$, where each mask $\mathbf{M}_i$ contains weighted information of the i-th component. Therefore, the masks could be used directly on the matrix $\mathbf{X}$ ensuring that the sum of the components is also $\mathbf{X}$. Once the mask $\mathbf{M}_i$ is obtained, it is possible to define the i-th masked component of $\mathbf{X}$ as (Bryan and D. Sun, 2013; Canadas-Quesada et al., 2017; Essid and Ozerov, 2014; Shah, Koch, et al., 2015; Shah and C. B. Papadias, 2013):

$$\mathbf{S}_i = \mathbf{M}_i \odot \mathbf{X} \tag{3.10}$$

Finally, through the process it is possible to ensure that the sum of masked components $\mathbf{S}_i$ gives the original matrix, that is:

$$\mathbf{X} = \sum_{i=1}^{K} \mathbf{S}_i \tag{3.11}$$

An example of this masks can be appreciated in the figure 3.2, where a decomposition with $K = 3$ components using Wiener filter is done.

Once applied the masks over the matrix $\mathbf{X}$, and considering that it represents the spectrogram magnitude of a signal, it is possible to perform the inverse transformation of the spectrograms of each component $\mathbf{S}_i(t, f)$ to obtain the temporal representation $s_i(n)$. Using the phase of the original spectrogram, it is possible to obtain the STFT for each component as (Bryan and D. Sun, 2013; Essid and Ozerov, 2014):

$$STFT_i = \mathbf{S}_i \odot e^{j\angle \mathbf{S}} \tag{3.12}$$

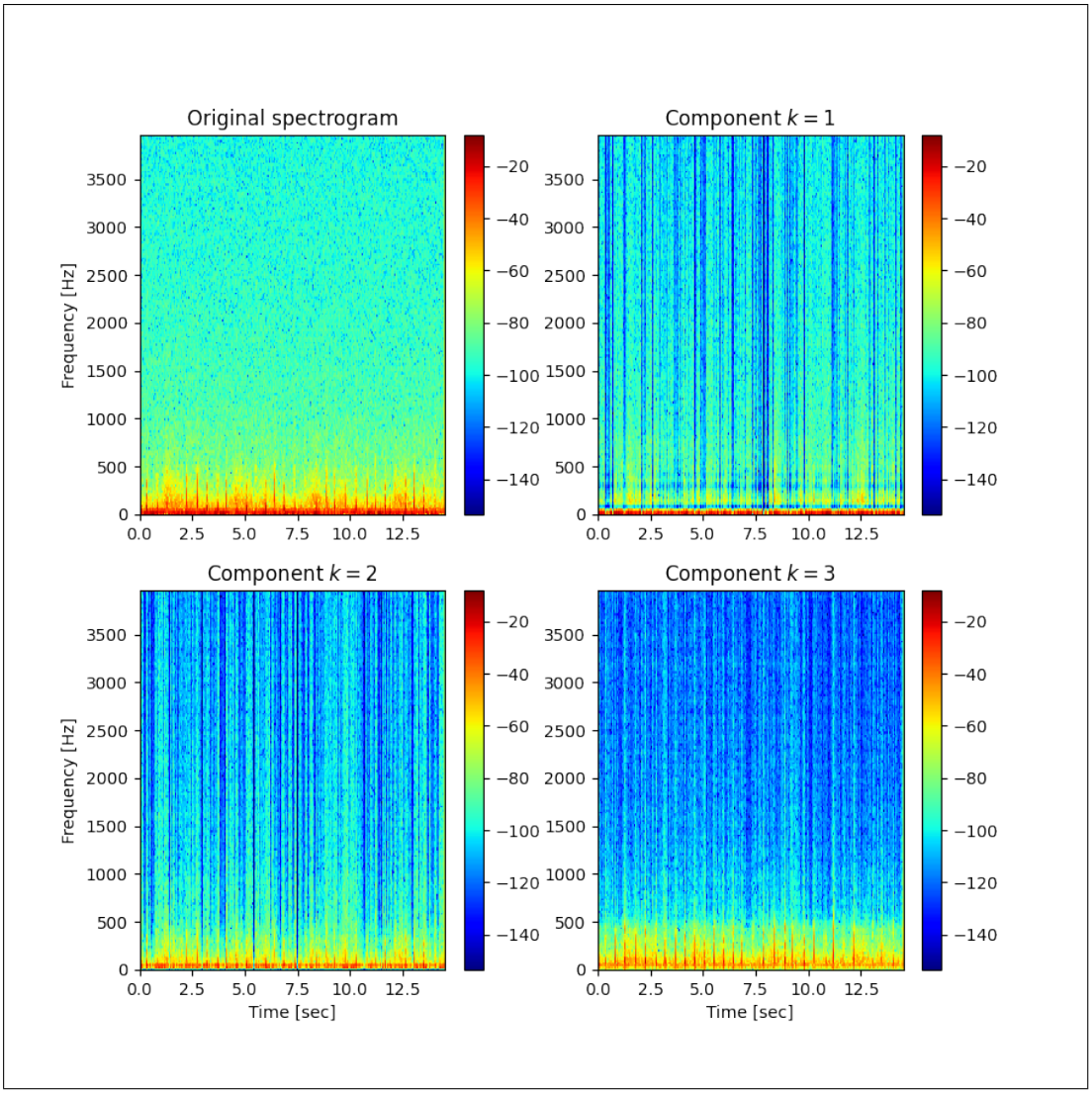

Figure 3.2: Results of the NMF decomposition (in dB) using Wiener filter for $K = 3$ components. As can be seen, each component has a proportion of the original signal. The sum of the 3 components results in the original spectrogram.

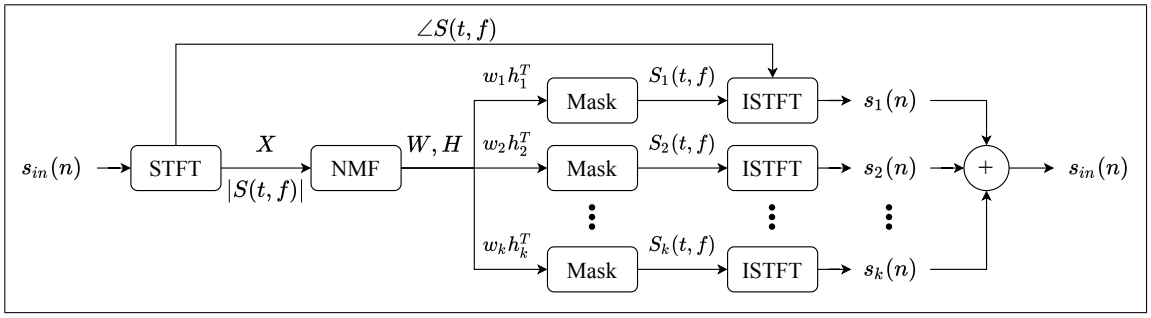

Figure 3.3: General diagram of the NMF decomposition process for source separation. This figure summarizes each step of the process, from the input of the interest signal $s_{in}(n)$ in time domain to the recovery of each component $s_i(n)$ in the same domain.

Where $\angle\mathbf{S}$ is the phase of the original spectrogram and j corresponds to the imaginary unit. Finally, calculating the inverse STFT (ISTFT) and taking the real values of this transform to recover the K components $s_i(n)$ that reconstruct the original signal:

$$s_i(n) = \Re(ISTFT(STFT_i)) \tag{3.13}$$

Satisfying that:

$$s_{in}(n) = \sum_{i=1}^{K} s_i(n) \tag{3.14}$$

In which the $\Re(\cdot)$ operator gets the real part of its argument. Figure 3.3 shows a diagram that summarizes the NMF decomposition process.

One of the main difficulties that NMF presents is that the solution is not unique. Indeed, if we can express a matrix $\mathbf{X}$ as the multiplication of two matrices $\mathbf{W}^*$ and $\mathbf{H}^*$, if there is an invertible matrix $\mathbf{Q}$, it is also possible to express:

$$\mathbf{X} = \mathbf{W}^*\mathbf{H}^* = \mathbf{W}^*\mathbf{Q}^{-1}\mathbf{Q}\mathbf{H}^* \tag{3.15}$$

Which are still valid solutions (Essid and Ozerov, 2014; Ganesh R. Naik, 2016). Nevertheless, among the main advantages that NMF have over other similar methods (like PCA (Ganesh R. Naik, 2016; Z. Y. Zhang, 2012)) is the ease of interpreting its results. Because subtractive combinations are prohibited, each component is a constitutive part of the original result (Févotte, Vincent, et al., 2018). This is due to the fact that, geometrically, the base vectors generate a cone that contains the original data in the positive ortant, so the solutions of the decomposition are bounded (Ganesh R. Naik, 2016). Therefore, each component can be interpreted as a portion or proportion of the original matrix. These properties will be useful in the heart and lung separation methods.

3.3 Databases

In this work, two databases were used. The first is a respiratory sounds database presented at International Conference on Biomedical Health Informatics (ICBHI) at 2017 (Rocha et al., 2019). This dataset was created by two research teams in Greece and Portugal using different electronic stethoscopes (*3M Littmann Classic II SE Stethoscope*, *3M Litmmann 3200 Electronic Stethoscope* and *WelchAllyn Meditron Master Elite Electronic Stethoscope*) and microphones (*AKG C417L Microphone*). It contains 920 recordings of a total of 126 patients, varying between 10-90 seconds and sampled at different rates (44100, 10000 and 4000 Hz). Each of these sounds labels the patient ID number, the recording index for each patient, the location of the instrument capture (variable between trachea and anterior/posterior left/right area of the chest and left/right lateral side of the chest) and the sound acquisition mode (single channel or multiple channels). Furthermore, for each patients in this database its diagnosis is indicated.

The second is a heart sounds database presented in the challenge posed by (Bentley et al., n.d.) during 2011, whose objective was the segmentation and classification of these sounds based on diseases. This database contains two datasets (A and B) which contain normal heart sounds, with murmurs, with artifacts, and with the presence of extra heart sounds (such as S3 or S4). In dataset A, 176 heart sounds sampled at 44100 Hz are presented, of which 21 have labels for the position where the fundamental heart sounds are located and all correspond to normal patients recordings. Meanwhile, dataset B contains 676 heart sounds sampled at 4000 Hz, of which none are labeled.

From each databases, a set of 12 cardio-respiratory sounds is generated from the sum of heart and lung sounds, making sure that both signals have the same energy. For the selection of the lung signals of the base (Rocha et al., 2019), respiratory sounds were auscultated from the thorax, and they have a very weakened inherent heart sound component, both auditory and graphically (based on their temporal and spectral information). For heart sounds, PCGs that have a low baseline noise level are used. The position of each fundamental sound can be obtained through heart sound detection algorithms such as (Renna et al., 2019; Springer et al., 2016), or by manual recognition and labeling of the signal. In this work, a convolutional neural network based on an encoder-decoder architecture is used, which indicates the silence (S_0) and heart sound segments (S_1 and S_2).

Since the heart and lung sounds in the different databases do not have the same sampling rate in most cases, all signals will be resampled at $f_s = 11025$ Hz. Additionally, and in order to reduce the presence of noise in segments free of heart sound, sound attenuation is performed by a factor $\kappa = 0.05$ on the S_0 segments. For this, the binary signal that indicates the heart sounds position is used, which is convolved with a 100-point Hamming window to smooth the transition between states. This smoothed signal is multiplied element-wise on the recording of the heart sound, ob-

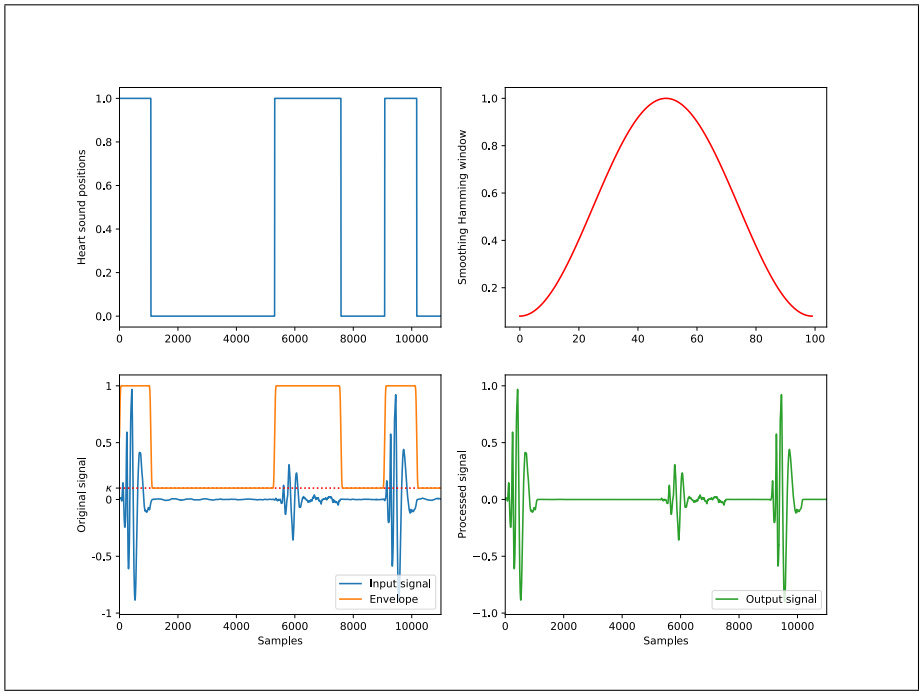

Figure 3.4: Noise reduction process in segments free of heart sound.

taining a signal whose segments S_0 are attenuated which will be added to the lung sound. The illustration of this process is presented in figure 3.4.

With this synthetic database of 12 cardio-respiratory sounds, the performance of source separation will be evaluated, since both the lung and the heart signal are previously known. Therefore, direct comparisons can be made between the obtained signals and the original signals.

3.4 Implementation

In this paper, the heart and lung sound separation problem is divided into two main modules: preprocessing and source separation strategy. In relation to the latter, different combinations will be made between the decomposition methodology with NMF and the assignment of the components in the heart cluster K_{heart} or the lung cluster K_{lung} for NMF components (see figure 3.5). From this, the output lung sound $s_{lung}(n)$ and heart sound $s_{heart}(n)$ will be defined following synthesis process:

$$s_{lung}(n) = \sum_{k \in K_{lung}} s_k(n) \tag{3.16}$$

$$s_{heart}(n) = \sum_{k \in K_{heart}} s_k(n) \tag{3.17}$$

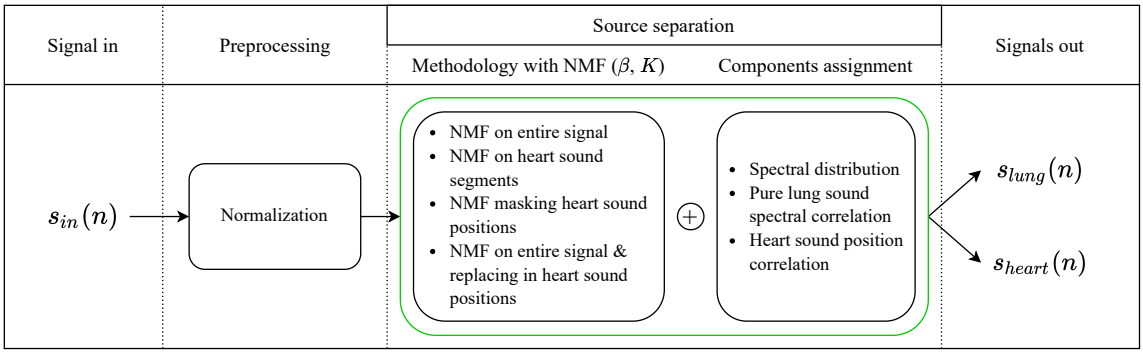

Figure 3.5: General diagram of the source separation.

In the next sections the detail of each module will be explained.

3.4.1 Preprocessing

Different factors such as the capture position and gain of the stethoscope, together with the age, sex and physiology of the patients, generate variability between the samples that compose the database. To reduce these effects the input signal will be normalized using:

$$s_{norm}(n) = \frac{s(n)}{\max|s(n)|}, \quad \forall n \tag{3.18}$$

3.4.2 Decomposition methods using NMF

In this section, four source separation methods that incorporate the NMF decomposition process are presented.

3.4.2.1 NMF on entire signal

In this method, an NMF decomposition is used on the complete signal, assigning the components to K_{resp} or K_{heart}, and finally synthesizing the sound using (3.16) and (3.17) as illustrated in figure 3.6. This method will be the base scenario from which the comparisons will be made to evaluate if the methods proposed in the sections 3.4.2.2, 3.4.2.3 and 3.4.2.4 improve the results.

Given that the method just presented performs NMF decomposition on the entire signal, it is possible that it generates distortions in segments in which the heart sound is not present, disturbing the segments of pure lung sound. To avoid this situation, and taking advantage that the heart sound is limited to specific areas of the signal, strategies will be used to preserve the properties of the segments that contain only lung sound.

In the next methods, it will be necessary to know previously the heart sound position. An example of the labeled signal can be seen at the upper right plot on

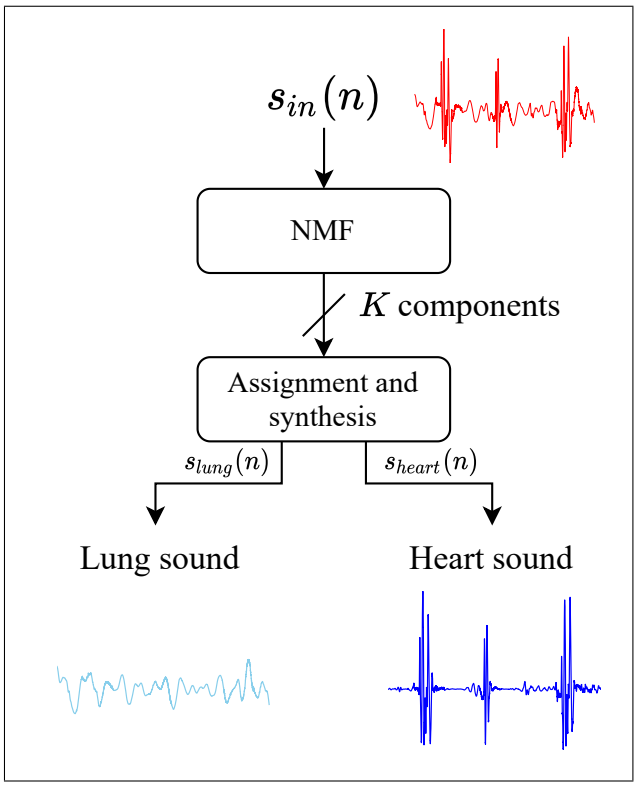

Figure 3.6: Diagram of the base method of NMF decomposition. In this implementation, the whole signal is decomposed into K components, which are grouped into clusters of heart and lung sounds.

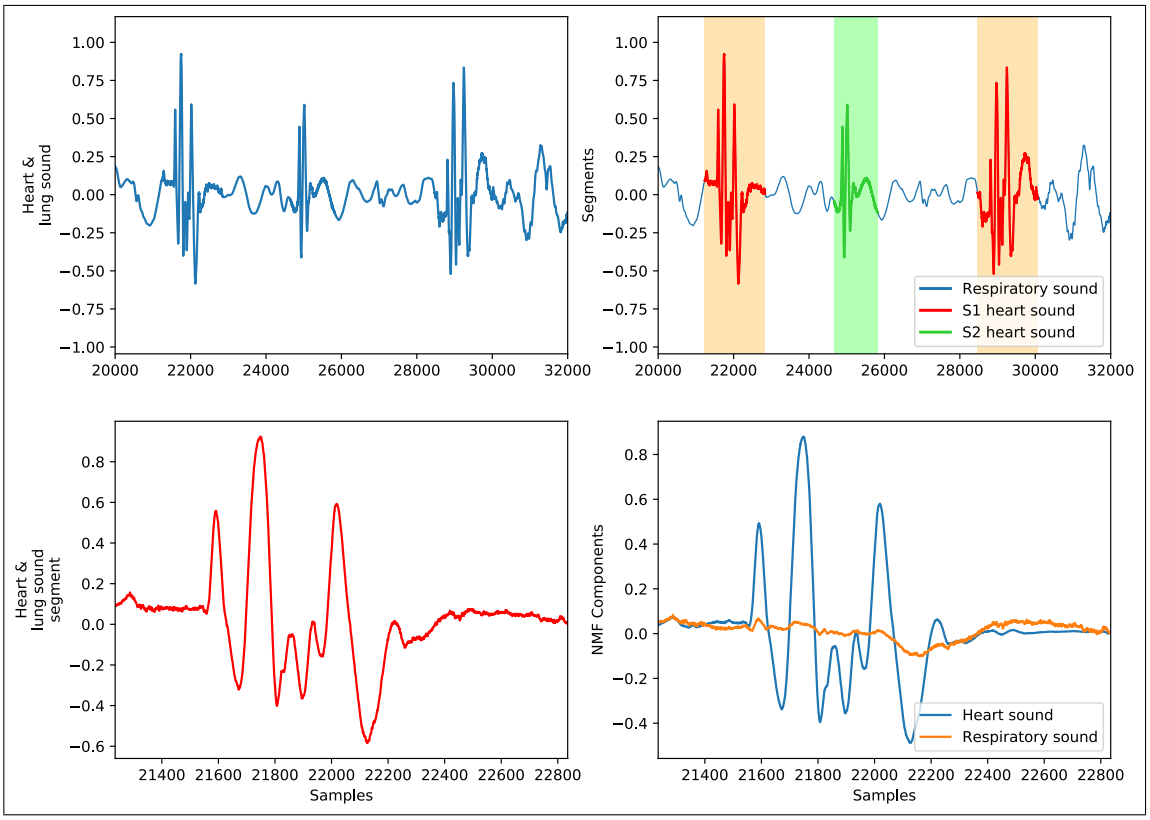

Figure 3.7: Heart sound segmentation graphs for NMF on heart sound segments method (f_s = 11025 Hz). Once the segments have been recognized (upper right graph), for each of these (lower left graph) the NMF decomposition is performed, obtaining the components that represent the heart and the lung sound (lower right graph).

figure 3.7, which highlights the areas where the heart sound is present. It will be assumed that the rest of the signal contains only lung sound.

3.4.2.2 NMF on heart sound segments

In this method, NMF decomposition is performed on each of the heart sound segments independently, as shown in figure 3.8. To decompose, it is necessary to extract the segment of interest from the rest of the signal, also obtaining a signal without this segment. Once the NMF decomposition has been carried out and the K components have been obtained, $s_{lung}(n)$ and $s_{heart}(n)$ will be classified and synthesized in the same way as in the equations (3.16) and (3.17). Then, $s_{lung}(n)$ will be inserted into the signal that was left without the segment, while $s_{heart}(n)$ will be inserted into an array of zeros at the corresponding time positions.

Inserting these segments can generate clicks caused by the discontinuities between the inserted segment and the original signal, a phenomenon displayed at the first row of figure 3.9. To correct this problem, a fade between the two segments is used to achieve a smooth transition. We will implement a fading with N_{fade} points at the edges of each segment, using a tail of a raised cosine with $\beta_{rc} = 1$ and $T = 1$ as a fading function, as can be seen in the plots of the second row in figure 3.9. The modified raised cosine with $T = 1$ used in this work (Couch, 1983; Proakis, 2001) is given by:

$$h(n) = \begin{cases} 1, & |n| \leq \frac{1-\beta_{rc}}{2} \\ \frac{1}{2}\left[1 + \cos\left(\frac{\pi}{\beta_{rc}}\left[|n| - \frac{1-\beta_{rc}}{2}\right]\right)\right], & \frac{1-\beta_{rc}}{2} < |n| \leq \frac{1+\beta_{rc}}{2} \\ 0, & \text{otherwise} \end{cases} \tag{3.19}$$

It is possible to see the result of the connection between the two segments in the graphs of the third row in figure 3.9, where a continuous transition between both signals is achieved.

3.4.2.3 NMF masking heart sound positions

In this method, the heart sounds positions are used to mask the input signal $s_{in}(n)$, as shown in figure 3.10. On the one hand, the masked respiratory sound is obtained, which corresponds to a vector containing all the sound segments where no heart sound is detected. This vector has zeros in the segments where heart sound is detected. On the other hand, the masked cardio-respiratory sound is obtained, which corresponds to a vector containing all the segments in which the heart sound is overlapped with the lung sound. This vector has zeros in the segments where no heart sound is detected. This process can be seen in detail in figure 3.11.

Then, the masked signal with the heart sound segments is used as input to the NMF decomposition process, from which K components will be obtained, classified

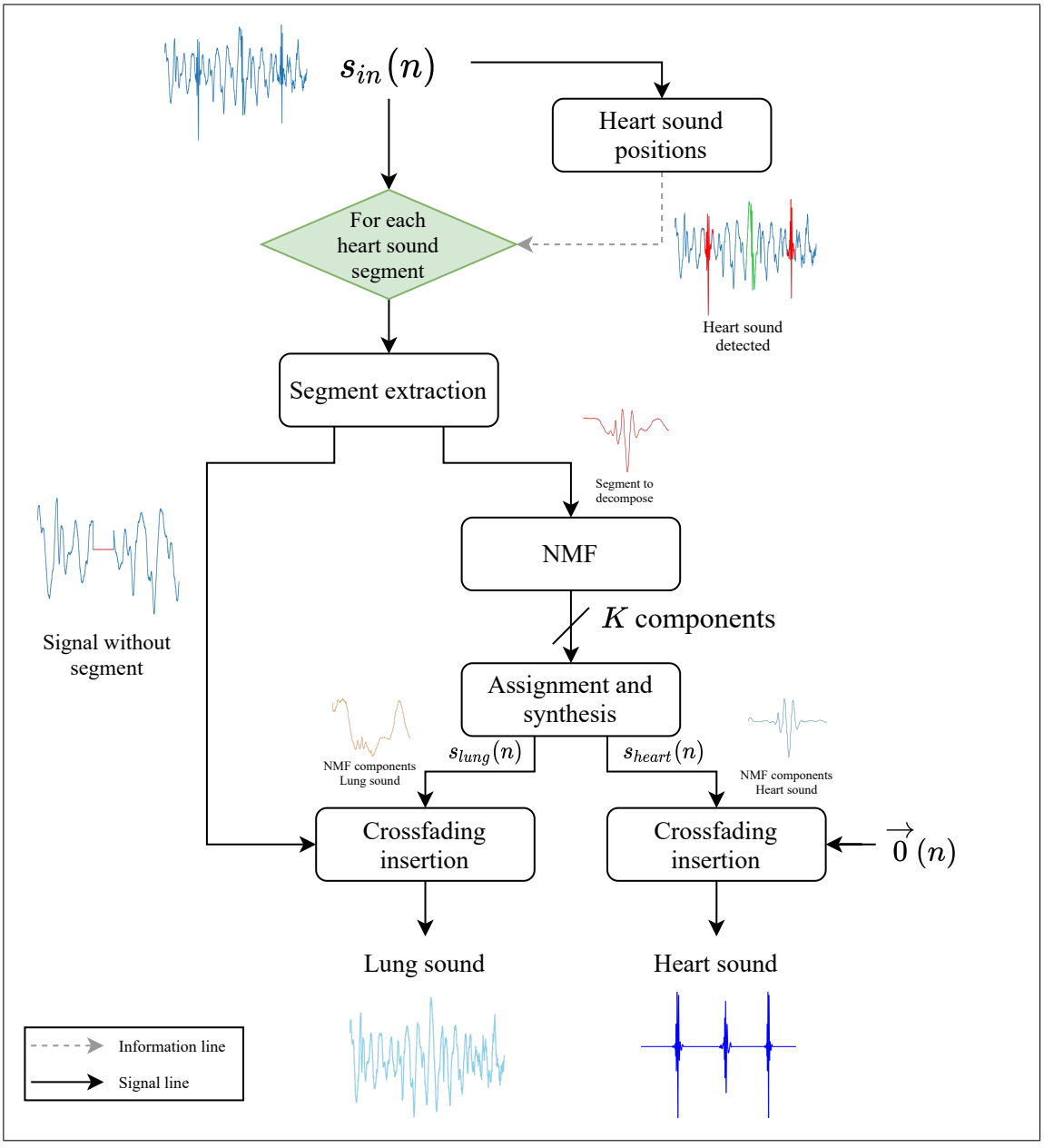

Figure 3.8: Diagram of the NMF decomposition method on heart sound segments. Once the lung sounds of a segment are obtained, they are mixed with the adjacent pure lung sounds defined in the heart sound detection process. Meanwhile, the heart sound is concatenated to a zero vector at its corresponding time position.

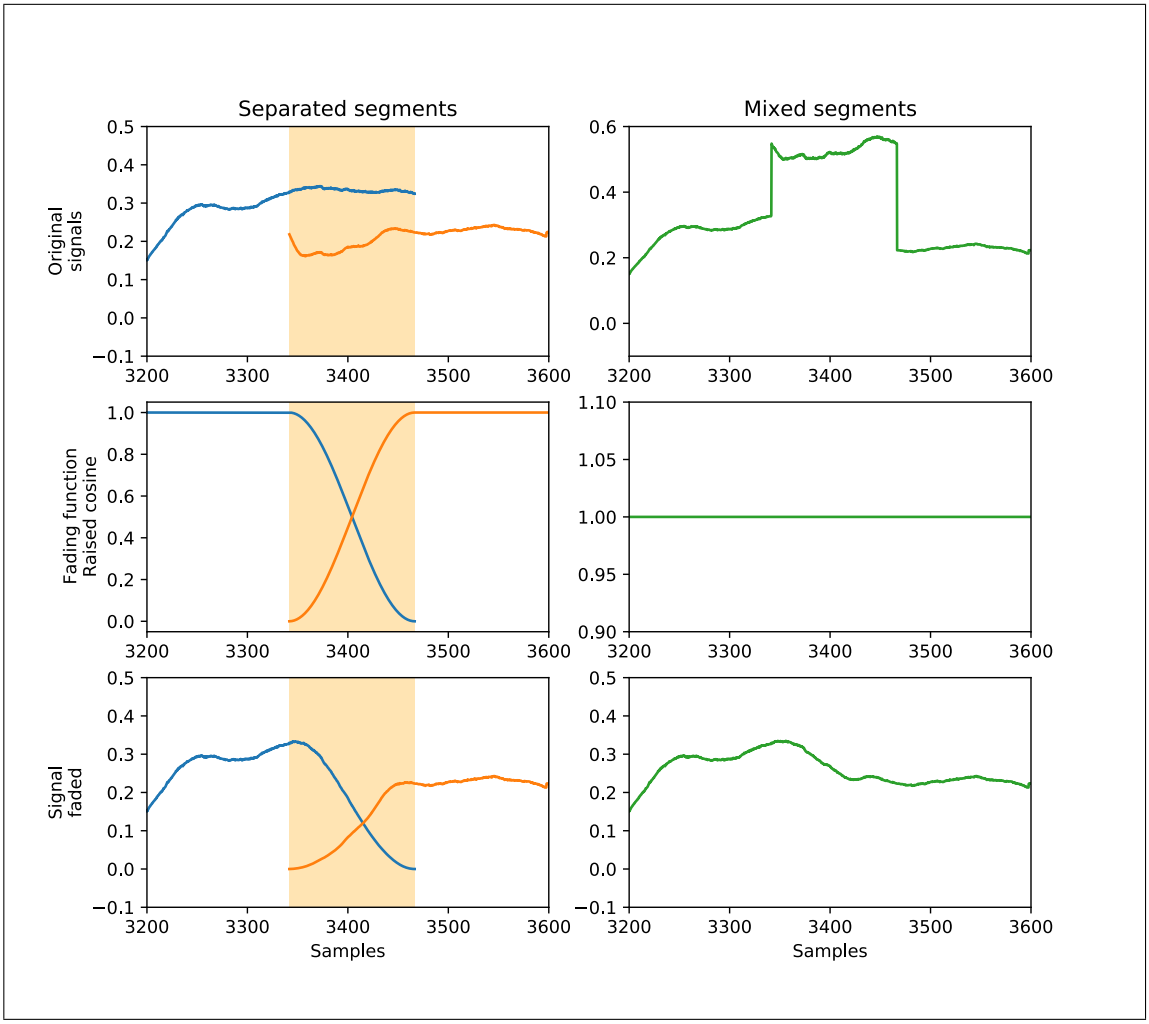

Figure 3.9: Transition zones between different signal segments at $f_s = 11025$ Hz (highlighted in orange). It is important that the transition would be smooth (as continuous as possible) in order to avoid artifacts within the estimated signal such as clicks (first row). To avoid this problem, the tails of a raised cosine window are used, whose sum is constant throughout n (second row). By applying this window it is possible to achieve a transition that does not introduce clicks (third row).

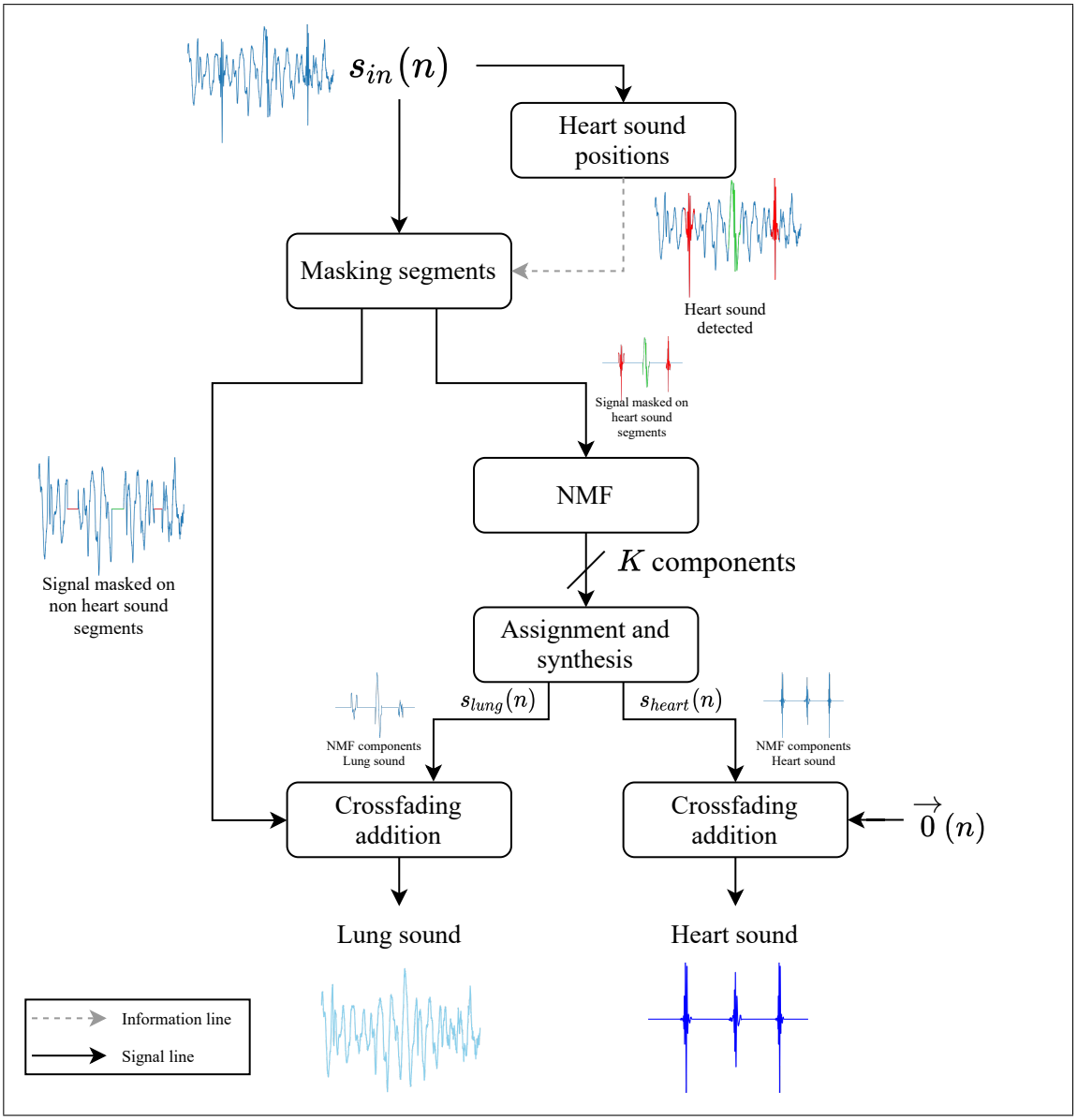

Figure 3.10: Diagram of the NMF masking heart sound positions method. In this case, it is proposed to mask the segments where the temporal overlap between heart and lung sounds occurs, performing the NMF decomposition to this signal. The masked lung sound will be concatenated with the components that are recognized as lung sound in the NMF decomposition. Meanwhile the heart sound will be the result of the components recognized as such in the NMF decomposition.

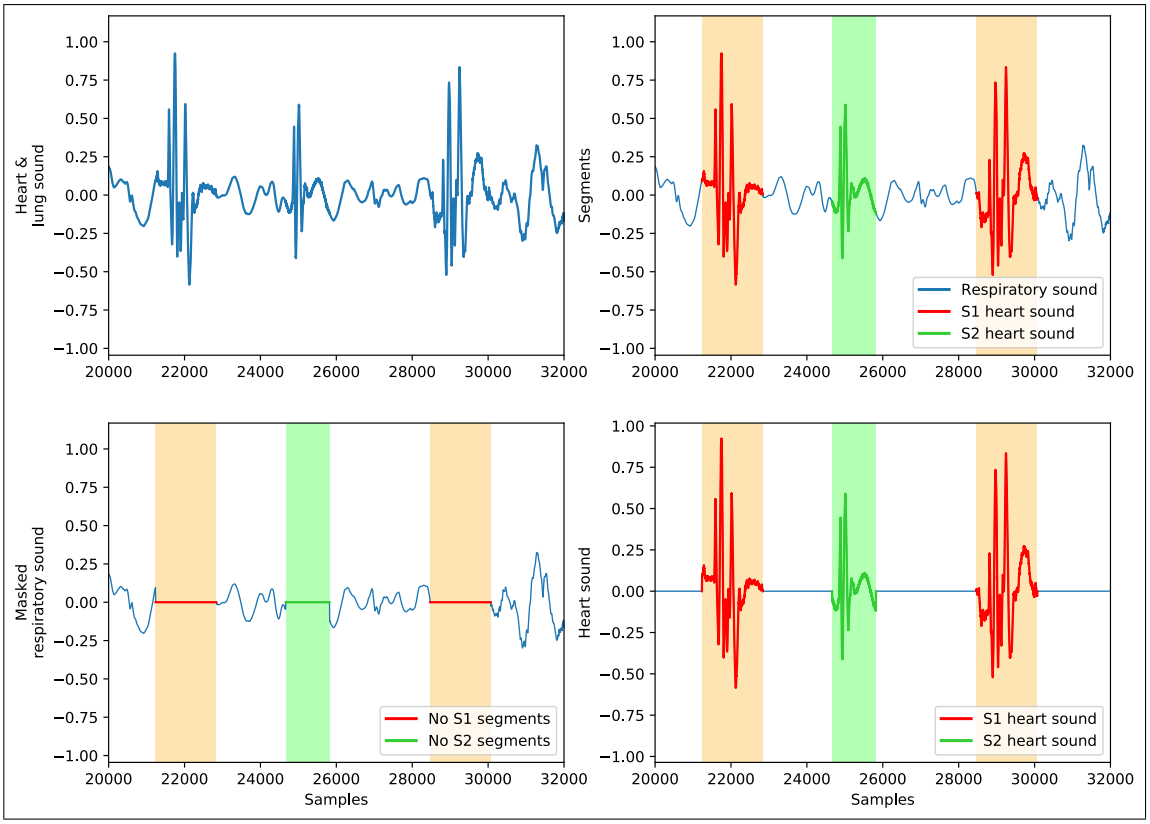

Figure 3.11: Heart sound segmentation plots for the NMF masking heart sound positions method. Once the heart sound positions are recognized (upper right graph), 2 signals are generated: one signal in which the heart sounds have been extracted, which will represent the pure lung sound segments (lower left graph); and one signal that will consist solely of these segments, connected with an array of zeros and that will represent the cardio-respiratory sound to be separated (lower right graph).

and synthesized using the equations (3.16) and (3.17) to get the sounds $s_{lung}(n)$ and $s_{heart}(n)$. Finally, $s_{lung}(n)$ is summed with the masked lung signal, making sure that the edges of each segment are faded following a process similar to the mentioned in section 3.4.2.2. The same process will be performed for $s_{heart}(n)$, but adding it to an array of zeros.

3.4.2.4 NMF on entire signal & replacing in heart sound positions

In this method, like the base method, a NMF decomposition is performed on the entire signal as presented in figure 3.12. However, it is proposed to use the pure lung sound segments of the input signal $s_{in}(n)$, in conjunction with the $s_{lung}(n)$ segments where the heart sounds were originally located (both highlighted in red in the figure 3.12). To obtain the output lung sound, these segments of interest are added making sure that the edges of each segment are faded. On the other hand, heart sound is simply defined as the output of the component assignment and synthesis process performed after decomposition. This method can be understood as a direct extension of the NMF on entire signal method.

3.4.3 Components assignment

Once the K has been obtained from the NMF decomposition, the main challenge is its classification in heart or lung sound. For this, it is necessary to define criteria that allow to assign each component based on its intrinsic characteristics. Next, we propose 3 assignment criteria inspired by (Canadas-Quesada et al., 2017).

3.4.3.1 Spectral distribution

This criterion exploits the frequency bands where both sounds predominate. Heart sounds predominate in the frequency bands between 30-120 Hz (S_1) and 70-150 Hz (S_2) (Rudnitskii, 2014), while lung sounds are concentrated in the 20-100 Hz frequency band (Pasterkamp et al., 1997). Therefore, heart sounds tend to be concentrated in a slightly higher frequency band than lung sounds.

To design an indicator of this notion, we propose a metric that we will call p% frequency energy, defined as:

$$C_1(i) = \begin{cases} \min\ f \\ \text{s.t.} \quad \sum_{n=0}^{f} |w_i(n)|^2 \geq p\% \cdot \sum_{n=0}^{N} |w_i(n)|^2 \\ 0 \leq f \leq N\ ,\ f \in \mathbb{Z} \end{cases} \tag{3.20}$$

Where w_i is the i-th spectral pattern obtained from the matrix $\mathbf{W}$, N corresponds to the row dimension of the matrix $\mathbf{W}$ (associated with the Nyquist frequency) and

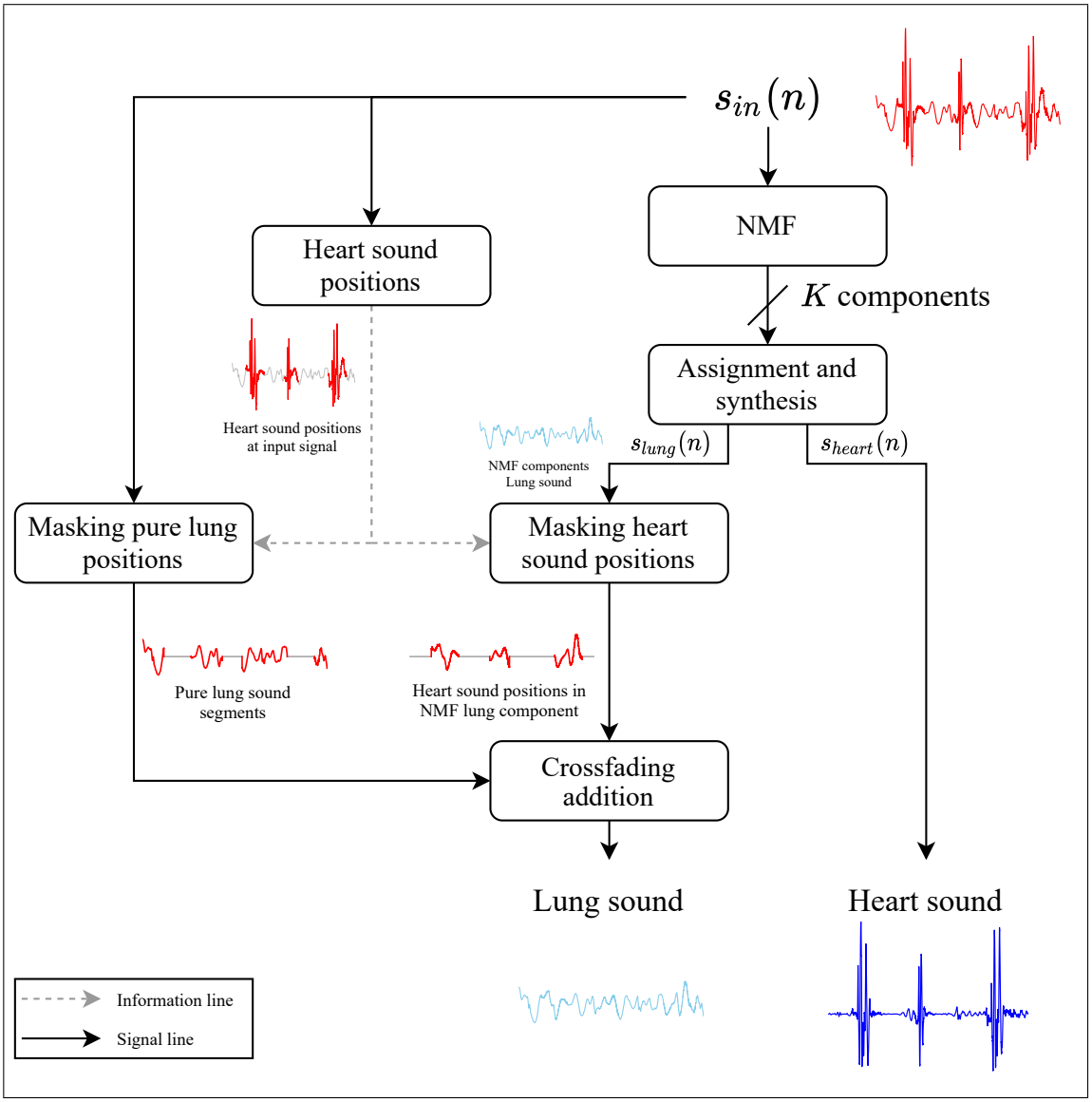

Figure 3.12: Diagram of the NMF on entire signal & replacing in heart sound positions method replacing segments. In this case, it is proposed to mask the segments free of heart sound, and to use the heart sound positions in the lung sound obtained by NMF. From the latter, the lung sound segments obtained are inserted into the masked lung signal.

$p\%$ corresponds to the energy percentage of interest to be compared. Intuitively, in (3.20) we seek to find the frequency bin f such that the accumulated energy up to this frequency is $p\%$ of the total energy. Unlike (Canadas-Quesada et al., 2017), where the author uses a fixed threshold to classify, in this work an adaptive threshold is proposed based on the shape of the data. This threshold is defined as the mean of the $C_1(i)$:

$$U_e = \frac{1}{K}\sum_{i=1}^{K} C_1(i) \tag{3.21}$$

Finally, if the value of the descriptor $C_1(i)$ is greater than the threshold U_e, then the i-th component will correspond to heart sound. Otherwise it will correspond to lung sound. This is summarized in the following expression:

$$\begin{cases} \text{If } C_1(i) \geq U_e, & w_i \in \text{Heart sound} \\ \text{Else}, & w_i \in \text{Lung sound} \end{cases} \tag{3.22}$$

3.4.3.2 Pure lung sound spectral correlation

This criterion focuses on the study of the spectral patterns obtained in the matrix $\mathbf{W}$ using the segments with only lung sound (signal segments in blue in figure 3.11). One of the main advantages of performing this method is that information from the same signal is used, unlike (Canadas-Quesada et al., 2017) where an external database of pure heart sounds is used as a reference to cluster the components with similar characteristics.

From this, it is possible to construct a dictionary $\mathbf{W}_{dic} \in \mathbb{R}^{N \times K_{dic}}$ whose columns will correspond to the PSD of the segments with pure lung sound, where N corresponds to the row dimension of the matrix $\mathbf{W}$ and K_{dic} corresponds to the number of segments free of heart sound in the input signal. Because the segments generally have different lengths, it is necessary to define a uniform fixed length in order to be able to compare them with the spectral patterns of the $\mathbf{W}$ matrix. To achieve this, the signal's periodogram is calculated using Welch's method with a window of length $2N$ and 75% overlap.[2] Then, the spectral correlation for the i-th NMF basis w_i and the j-th dictionary basis w_{dic_j} is defined as the Pearson correlation coefficient between both signals:

$$C(i,j) = \frac{1}{N-1}\sum_{n=1}^{N}\left(\frac{(w_i(n)-\mu_{w_i})(w_{dic_j}(n)-\mu_{w_{dic_j}})}{\sigma_{w_i}\sigma_{w_{dic_j}}}\right) \tag{3.23}$$

Where μ_s is the mean of the signal $s(n)$ and σ_s its standard deviation. Once the correlations between the i-th basis w_i and all the bases of $\mathbf{W}_{dec}$ have been calculated,

[2] A window length $2N$ is chosen since only half the spectrum is desired (between 0 and the Nyquist frequency).

the one with the highest correlation of all is chosen. Therefore, the representation parameter of the w_i component is defined as:

$$C_2(i) = \max_j C(i,j) \tag{3.24}$$

Finally, the following criteria are applied to perform the clustering.

$$\begin{cases} \text{If } C_2(i) < U_f, & w_i \in \text{Heart sound} \\ \text{Else,} & w_i \in \text{Lung sound} \end{cases} \tag{3.25}$$

Where the threshold is defined as the mean of the coefficients $C_2(i)$ obtained:

$$U_f = \frac{1}{K} \sum_{i=1}^{K} C_2(i) \tag{3.26}$$

3.4.3.3 Heart sound position correlation

Given that the heart sound has a characteristic temporal pattern, this information is used as a criterion that allows classifying a component by comparing it with the i-th temporal pattern h_i^T from matrix $\mathbf{H}$. To achieve this, $P(n)$ is defined as a sequence of binary pulses, being 0 in the segments where no heart sound is detected and 1 in which it is detected. This result can be seen in the left frames in figure 3.13. Note that the i-th temporal pattern $h_i^T \in \mathbb{R}^{M\times 1}$ and that $P(n) \in \mathbb{R}^{N_{s_{in}}\times 1}$, where $N_{s_{in}}$ corresponds to the length of the original signal $s_{in}(n)$ and M corresponds to the column dimension of the spectrogram. In general it is true that $M < N_{s_{in}}$. Therefore, in order for it to be compared with the time factor h_i^T it is necessary to make sure that both signals have the same lengths. To do this, h_i^T is interpolated, obtaining the signal $h_{i_{ext}}^T \in \mathbb{R}^{N_{s_{in}}\times 1}$ with the desired length.

Then, an estimate of the cardiac activations of the component $h_{i_{ext}}^T$ is calculated. For this, in (Canadas-Quesada et al., 2017) it is proposed to use the mean of the vector $h_{i_{ext}}^T$ as threshold, defining as 0 the elements below this threshold and 1 for the elements above this value. Then, the cardiac activations of the i-th component are defined as:

$$P_i(n) = \begin{cases} 1, & \text{si } h_{i_{ext}}^T(n) \geq \sum_{t=1}^{N_{s_{in}}} \frac{h_{i_{ext}}^T(t)}{N_{s_{in}}} \\ 0, & \text{si } h_{i_{ext}}^T(n) < \sum_{t=1}^{N_{s_{in}}} \frac{h_{i_{ext}}^T(t)}{N_{s_{in}}} \end{cases} \tag{3.27}$$

From the cardiac activation signals $P(n)$ and $P_i(n)$, and unlike (Canadas-Quesada et al., 2017), the following classification metric is defined as the percentage of coinciding points between the two signals:

$$C_3(i) = \frac{\#\{P(n) = P_i(n)\}}{N_{s_{in}}} \tag{3.28}$$

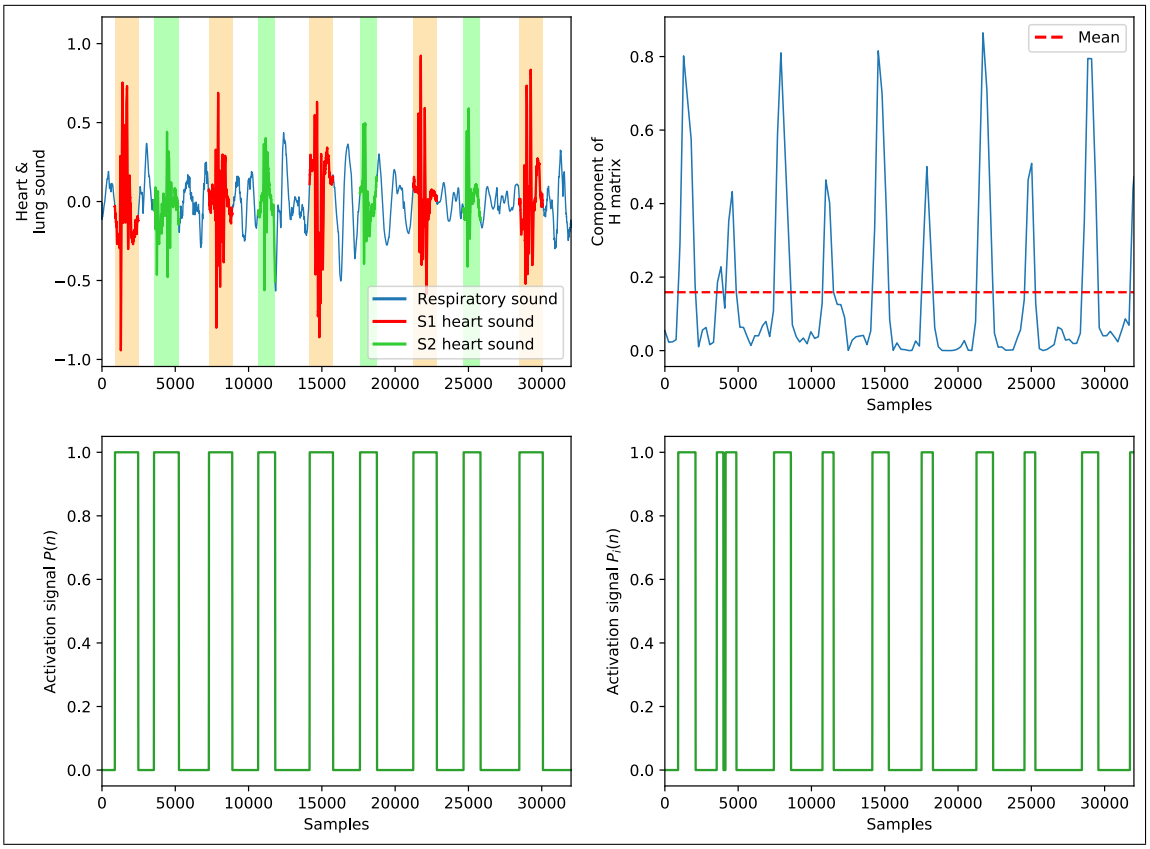

Figure 3.13: Graphs of cardiac activations. Once the heart sound positions are recognized (upper left graph), the binary signal of cardiac activations $P(n)$ (lower left graph) is constructed. In the same way, using the information from the i-th row of the $\mathbf{H}$ matrix, h_i^T, an estimated binary sequence of cardiac activations $P_i(n)$ is constructed comparing each point with the mean of the signal (upper and lower right graph respectively).

Finally, and as in the previous cases, the clustering criterion is based on the comparison with a threshold value U_t, defined by:

$$\begin{cases} \text{If } C_3(i) \geq U_t, & h_i^T \in \text{Heart sound} \\ \text{Else,} & h_i^T \in \text{Lung sound} \end{cases} \tag{3.29}$$

Where again the threshold is defined based on the mean of the results:

$$U_t = \frac{1}{K} \sum_{i=1}^{K} C_3(i) \tag{3.30}$$

An exceptional case of this criterion arises for the method presented in the section 3.4.2.2 where the decomposition is performed only in the heart sound segments. For this reason, it is not possible to define $P(n)$ in the same way as in the other cases.

As a result of this, the temporal pattern $P(n)$ is defined as a binary sequence adjusted to the boundaries of the segment to be decomposed, as shown in figure 3.14. In this case, the binary sequence is composed by a vector which has N_{fade} points with zero values at both edges of the heart sound segment, while the segment corresponding to heart sound is defined with ones (lower left plot). These slack points allow the segment to be subsequently mixed with the rest of the signal. It should be noted that what is decomposed by NMF corresponds to this entire signal, including the slack points (upper left graph in figure 3.14). As in the other cases, (3.27) is used to define the cardiac activation of the i-th component.

3.4.4 System parameters

To obtain results, the spectrogram of the input signal will be calculated using a Hann window of length $N_{wind} = 2048$ and 90% overlap, based on a analysis available in the appendix.

In relation to NMF decomposition, the variation of the objective function based on the β-divergence, using values of $\beta = 1$ (Kullback-Leibler divergence) and $\beta = 2$ (quadratic function). The number of K components in which the signal will be decomposed will vary between $K = \{2, 3, 4, 5, 7, 10, 15, 20, 30, 50\}$. In addition, a Wiener filter will be used to build the mask.

For the strategies presented in sections 3.4.2.2, 3.4.2.3 and 3.4.2.4, a number of $N_{fade} = 100$ fading points will be used for the mixing between segments (see figure 3.9).

Regarding the clustering criteria, as shown in figure 3.5, the impact of each clustering criterion will be studied.

Finally, to perform the NMF decomposition of the signal $s_{in}(n)$, the Python `scikit-learn` library (Pedregosa et al., 2011) is used with a multiplicative update solving method of a maximum of 500 iterations and a tolerance range of 10^{-4}.

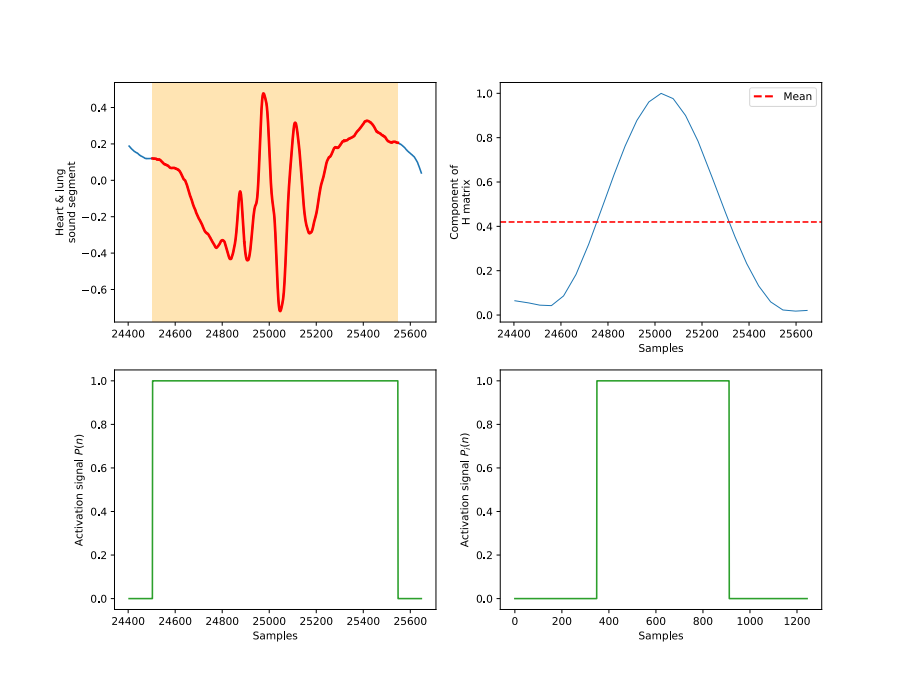

Figure 3.14: Cardiac activation in specific heart sound segment.

3.5 Results analysis

Due to the diversity of parameters and options that exist for each one, the analysis is divided into the study of assignment criteria (section 3.5.1) and NMF decomposition parameters (section 3.5.2).

Since the interest of this work is to recover the lung sound from the input signal, different metrics will be used to evaluate the performance of the decomposition, comparing the original lung sound with the sound obtained by the proposed methods. As a performance metric, the temporal correlation (Mondal, P. Banerjee, et al., 2017; Mondal, Saxena, et al., 2018) and the correlation between the Power Spectral Density (PSD) of the original signal with its respective recovered signal will be calculated. For this, the Pearson correlation coefficient will be used, defined as:

$$\rho = \frac{E\left[(s(n) - \mu_s)(\hat{s}(n) - \mu_{\hat{s}})\right]}{\sigma_s \sigma_{\hat{s}}} \tag{3.31}$$

Where $s(n)$ corresponds to the original lung sound and $\hat{s}(n)$ the obtained lung sound by NMF, either in time or their PSDs. This metric will allow quantifying the similarity between the original signal and the obtained one independent of the scale of both signals. The higher this indicator, the signals will have a more similar shape, and therefore the separation will perform better.

Furthermore, the Mean Square Error (MSE) will be calculated, defined as (Mondal, Saxena, et al., 2018) (Noman Q. Al-Naggar and Al-Udyni, 2018; Shah and C.

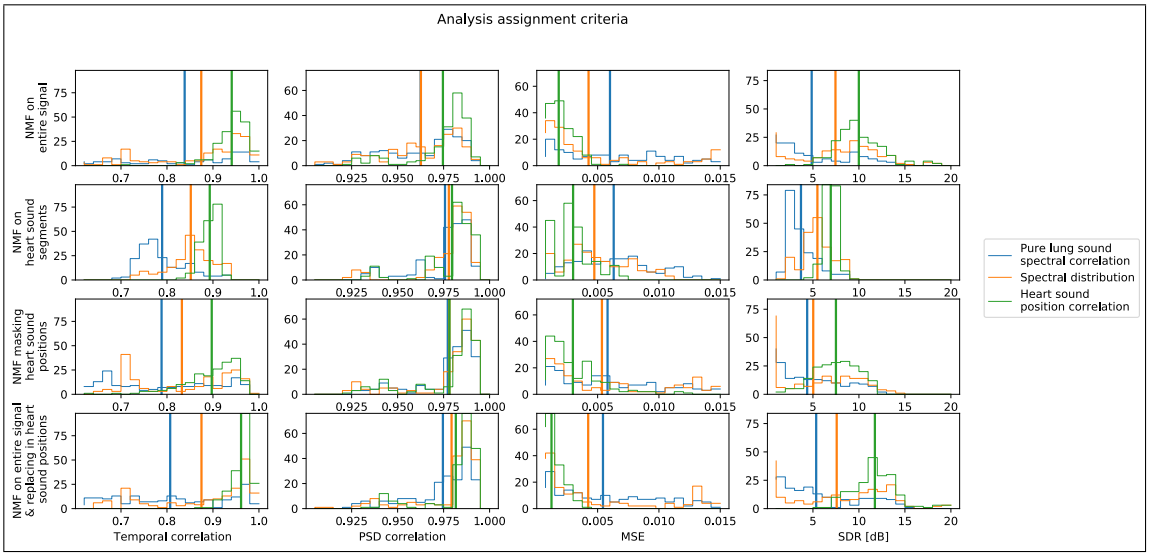

Figure 3.15: Histogram of results for assignment criteria using the temporal correlation, PSD correlation, MSE and SDR metrics for each type of NMF decomposition. The vertical lines indicates the mean of each histogram. It should be noted that this histogram incorporates only the simulations that use $N_{wind} = 2048$ and 90% overlap.

Papadias, 2013):

$$MSE = \frac{\|s(n) - \hat{s}(n)\|^2}{N_{signal}} \tag{3.32}$$

Where N_{signal} is the number of points that the signal has, $s(n)$ and $\hat{s}(n)$ are the original and obtained lung signal respectively in time domain. The MSE allows quantifying the raw error obtained by reconstruction. The lower the MSE, the better the separation performance.

Finally, the Signal to Distortion Ratio (SDR) will be used, and it is defined as (Canadas-Quesada et al., 2017; Mondal, P. Banerjee, et al., 2017; Vincent et al., 2006):

$$SDR = 10 \cdot \log_{10} \left(\frac{\|s(n)\|^2}{\|s(n) - \hat{s}(n)\|^2} \right) \tag{3.33}$$

Using the same nomenclature as previously. This value is expressed in decibels (dB) and the higher the SDR value, the better the reconstruction.

3.5.1 Selection of assignment criteria

This section will analyze the impact that each of the assignment criteria presented in section 3.4.3 have on the results. In figure 3.15 a histogram of the results obtained using each assignment criteria is presented for every NMF decomposition method under the different performance metrics. Each histogram is obtained from the variation

of the parameters K and β of the NMF decomposition, presented in the section 3.4.4. It should be noticed that in each histogram the assignment criteria are presented in different colors.

As can be seen, and independent of the decomposition method and the metric, the option that offers the worst performance is the pure lung sound spectral correlation. Indeed, this assignment method concentrates a large number of results in low ranges of temporal correlation and SDR, and high MSE error values. It even has a lower PSD correlation compared to the other criteria, especially in methodologies that use NMF on the entire signal (3.4.2.1 and 3.4.2.4).

Similarly, the spectral distribution criterion tends to give lower temporal correlations in all decomposition methods, except for the NMF on heart sound segments where this effect is not so significant. In relation to its SDR, in general it tends to spread over the entire range of dB, so it is not a reliable classification criterion. A similar effect occurs regarding the MSE of this criterion.

However, the heart sound position correlation criterion is systematically and robustly better for all the scenarios presented since it concentrates its temporal and PSD correlation close to one, its SDR in high values, while it concentrates its MSE in values close to zero. This can be confirmed by comparing the means of the histograms (represented by the vertical lines in each plot), where for each decomposition method and for each metric this criterion is the best. It should be noted that even performing this analysis on different window sizes and overlaps to obtain the spectrogram, the aforementioned properties of this criterion are maintained (see figure B.3 available in the appendix).

Therefore, the heart sound position correlation criterion for the assignment of the components will be preferred over the rest.

3.5.2 NMF parameters

In this section, we will analyze the impact that the β used in the divergence function and the number of components K have on the results, using the options presented in section 3.4.4.

3.5.2.1 β-divergence

When reviewing the histogram of the results obtained for $\beta = \{1, 2\}$, using a window of size $N_{wind} = 2048$ and 90% overlap for the calculation of the spectrogram (see figure 3.16), it is possible to notice that in general the results improve when $\beta = 2$, with the exception of the NMF on heart sound segments method. This feature is maintained even using different N_{wind} sizes and overlap (detail available in figure B.6 in the appendix). However, a curious result occurs when the heart sound positions correlation is used as the assignment criterion, since the decision of β becomes irrelevant in this case, presenting quite similar results (detail available in figure B.6 in

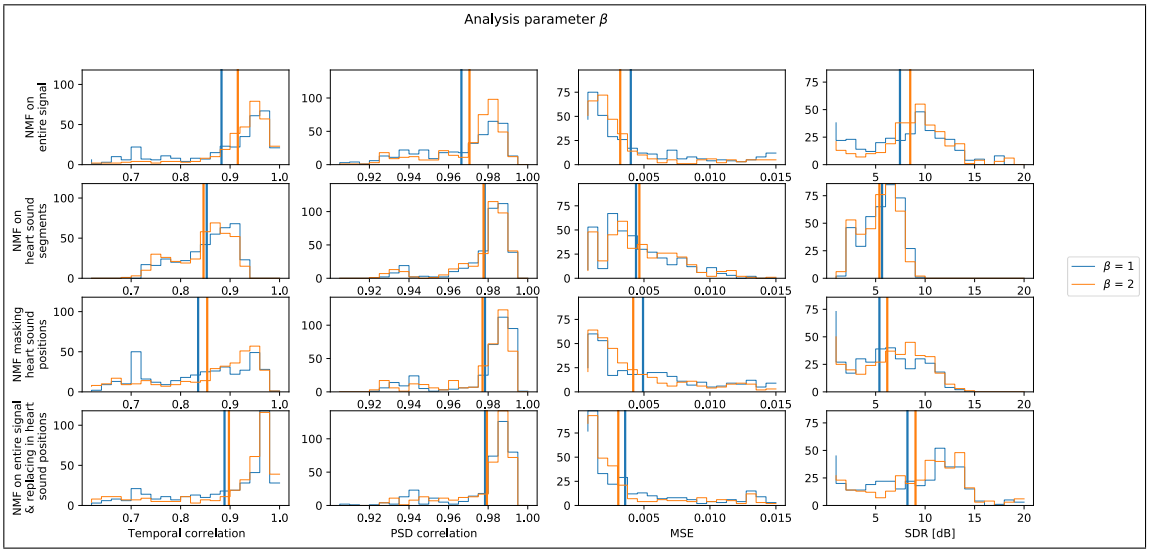

Figure 3.16: Histogram of results for β using the temporal correlation, PSD correlation, MSE and SDR metrics for each type of NMF decomposition. The vertical lines indicates the mean of each histogram. It should be noted that this histogram incorporates only the simulations that use $N_{wind} = 2048$ and 90%.

the appendix).

Therefore, the use of $\beta = 2$ will be preferred for this implementation.

3.5.2.2 Number of components K

Regarding this parameter, it is possible to note that the smaller the number of K components, the better results are obtained (see figure 3.17). Indeed, it is possible to note that for the different methodologies and for each of the metrics the option $K = 2$ presents a significant improvement compared to the other options. This result is also maintained when using different N_{wind} sizes and overlaps (detail available in figure B.5 in the appendix). However, and like the parameter β, if the heart sound position correlation criterion is used to assign the different components, there is no clear difference between the different values of K (detail available in figure B.7 in the appendix).

For this reason, it is preferable to use $K = 2$ at the time of decomposition since it systematically generates better results under different parameter variations.

3.5.3 Methodology with NMF analysis

By defining the β-divergence function to $\beta = 2$ and $K = 2$ as the number of components, the results presented in table 3.1 are obtained. As can be seen, the alternative that gives the best results is the NMF on entire signal & replacing in heart sound

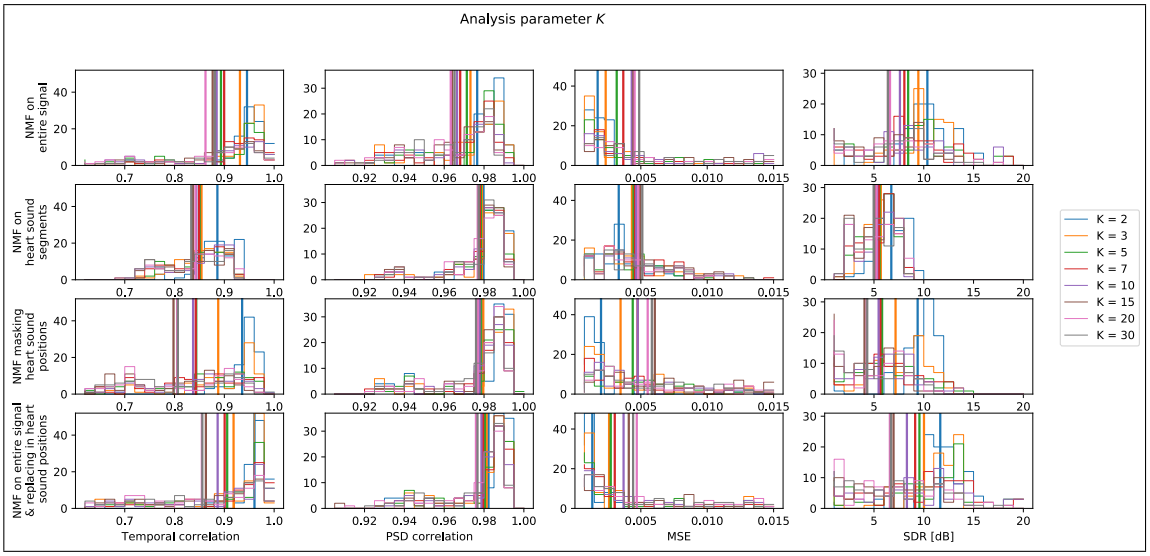

Figure 3.17: Histogram of results for the number of components K using the temporal correlation, PSD correlation, MSE and SDR metrics for each type of NMF decomposition. The vertical lines indicates the mean of each histogram. It should be noted that this histogram incorporates only the simulations that use $N_{wind} = 2048$ and 90% overlap.

Table 3.1: Results of the NMF methods comparing the original lung signal and the lung signal obtained for using $N_{wind} = 2048$, 90% overlap, $\beta = 2$, $K = 2$, and heart sound position correlation criterion. The best method for each metric is highlighted in yellow.

Metric\Method	**NMF on entire signal**	**NMF on heart sound segments**	**NMF masking heart sound positions**	**NMF on entire signal & replacing in heart sound positions**
Temporal correlation	0.9481 ± 0.024	0.8834 ± 0.0313	0.8648 ± 0.1396	0.965 ± 0.0143
PSD correlation	0.9777 ± 0.0173	0.9777 ± 0.0163	0.9759 ± 0.0205	0.9828 ± 0.0161
MSE	0.0016 ± 0.0008	0.0035 ± 0.0019	0.0042 ± 0.0053	0.0011 ± 0.0006
SDR (dB)	10.1806 ± 1.9769	6.6107 ± 1.3386	7.4382 ± 4.1117	11.8606 ± 1.5946

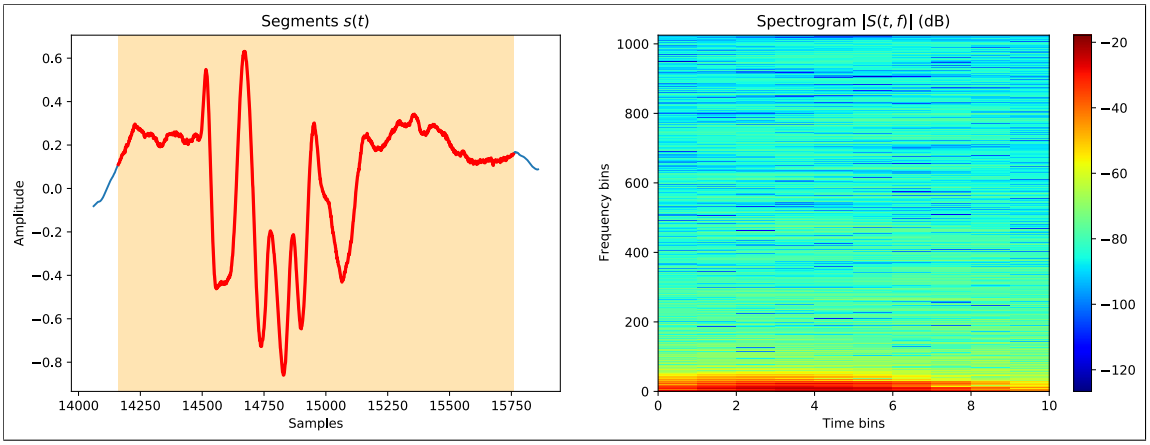

Figure 3.18: Example of a spectrogram of a heart sound segment with $N_{wind} = 2048$ and 90% overlap.

positions, since for all the metrics it has a better performance than the rest of the methods. This result is expected since, unlike the base method, it does not modify the pure lung sound segments, which is a desirable behavior in terms of lung sound reconstruction.

In addition, by decomposing on the entire signal (3.4.2.1 and 3.4.2.4), a better characterization of the lung sound is performed than in NMF methods that decompose only the heart sound segments (3.4.2.2 and 3.4.2.3). As presented in the table 3.1, these last two methods show the worst results under all the metrics, even worse than the baseline method. This may be due because these decompositions consider information only from the segments where the heart sound is present. In addition, nothing ensures that in the segments where the heart sounds occur there is the presence of respiratory sounds, since several beats can occur when the patient is neither inhaling nor exhaling. Therefore, in this type of segment, components that do not present the characteristic properties of respiration could be classified as lung sounds. As a result of this, the characterization of the lung sound in the NMF decomposition would be poorer.

In particular, in the case of the NMF on heart sound segments method, there may be an additional conflict due to the restriction of K required for NMF factorization, as mentioned in section 3.2.1. Indeed, by performing the decomposition only on the segments where the heart sound is located, the temporal dimension of the spectrogram contains fewer points than in the case of the complete signal. Because of this, there are not enough points in the components of the $\mathbf{H}$ matrix to adequately represent the temporal pattern of the components. The figure 3.18 shows a heart sound segment containing approximately 1500 points at a sampling rate of $f_s = 11025$ Hz, considering $N_{fade} = 100$ fade points. Therefore, when calculating the spectrogram using a window length of $N_{wind} = 2048$ with 90% overlap, the time dimension in the

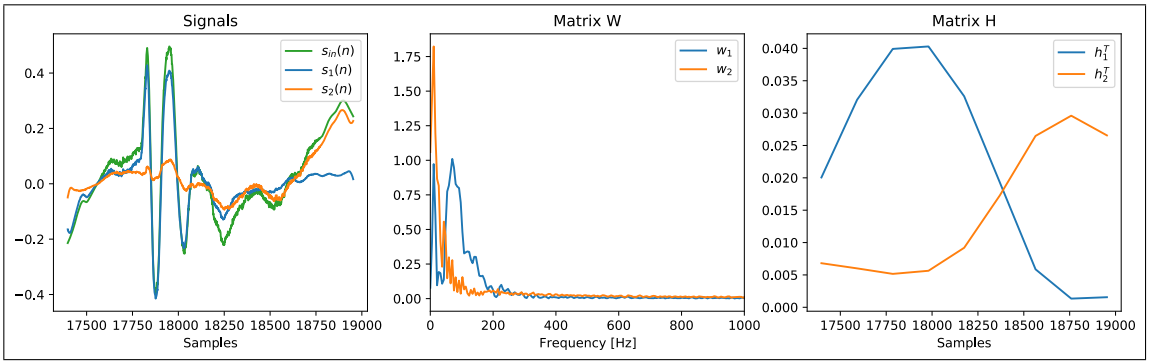

Figure 3.19: NMF decomposition example on a heart sound segment with $K = 2$. For this decomposition a spectrogram with $N_{wind} = 2048$ and 90% overlap was used. In the left plot the original signal and the reconstruction of both components is presented, where the component #1 correspond to heart sound and the component #2 to lung sound. The intermediate plot shows the spectral patterns w_i for each component. And in the right plot the temporal patterns h_i^T are shown.

spectrogram is described by only 10 bins, thus the constraint $(N + M)K \leq NM$ is not satisfied in some cases.

Another apparent cause is that this decomposition considers only the information contained in each segment individually without considering the other segments, including even less information than the NMF masking heart sound positions. This causes that there is no clear definition of the lung sound properties, and therefore, the definition of its spectral pattern in the W matrix. In practice, this means that the spectral pattern recognized in that segment tends to characterize the lung sound as basal noise, assigning to what is classified as heart sound almost all of the information and energy of the segment. The two facts mentioned can be seen in the result of figure 3.19, where the reconstructed components are presented together with their spectral and temporal patterns. Notice from the results of the **H** matrix (right plot) the low resolution of the temporal patterns. Furthermore, it is possible to see in the temporal representation of the components (left plot) that the component corresponding to the heart sound (component #1) has almost all the energy of the segment. While the component corresponding to the lung sound (component #2) oscillates much closer to zero.

Although the NMF method on the entire signal & replacing in heart sound positions achieves better results than the base method, it is dependent on the correct detection of the heart sounds in the auscultated signal. Indeed, if they are not detected correctly, the heart sounds would remain in the output lung sound, which is not desirable. Therefore, for the correct performance of this method, it is necessary that the heart sound detection step would be robust.

3.5.4 Best results

From the analysis of the experiments performed, it is possible to conclude that the combination of parameters that generates an optimal separation is the NMF on entire signal & replacing heart sound position method with $\beta = 2$ and $K = 2$. The graphical results of this separation are presented in the figures 3.20, 3.21, 3.22 y 3.23. One of the audio files from the created database was used to illustrate the properties of the decomposition.

The figure 3.20 shows the original cardiorespiratory signal to be decomposed, and the components obtained from the decomposition. The estimated lung sound in the segments where the heart sound occurs (highlighted in purple) has a similar shape to the areas of pure lung sound (not highlighted). In turn, the estimated heart sound presents characteristics of its impulsive nature, which is why it is possible to appreciate the good performance of the separation. This can also be seen in the spectrogram in figure 3.23, where the low-frequency peaks present in the original cardiorespiratory sound spectrogram are eliminated through this method.

Figures 3.21 and 3.22 show the comparisons between the original respiratory sound and the obtained respiratory sound. It is possible to appreciate that in the segments of pure lung sound the error between both signals is zero since they are not modified. However, in the segments where the decomposition is performed it is possible to appreciate low approximation errors. In general terms, the decomposition achieves to maintain the properties of the lung sound. Furthermore, spectrally this can be seen in the PSDs presented in figure 3.22, which are quite similar.

3.6 Conclusions and future work

In this work, three new methods are proposed for source separation of this signals. The first method consists in the decomposition only on the segments where heart sound is detected. The second method consists of masking the signal, leaving only the segments where the heart sound is found and zeroing the rest. The third method uses the components classified as lung sound with an NMF on entire signal method, to obtain the segments where the heart sound was originally located to mix it with the original signal. These methods were compared to a method which performs NMF decomposition on the entire raw signal.

In addition, three strategies to assign components to the sources are proposed, which use information from the same input signal to make the decision. The first uses the spectrum energy distribution, the second the PSD of the pure lung sound segments and the last uses heart sound positions detected in the signal.

Based on the results, it is possible to conclude that the NMF on entire signal & replacing the heart sound positions offers better results for almost all scenarios. Indeed, because there are areas that remain unaltered, it is possible to better recover the properties of lung sound. However, it is dependent on the correct detection

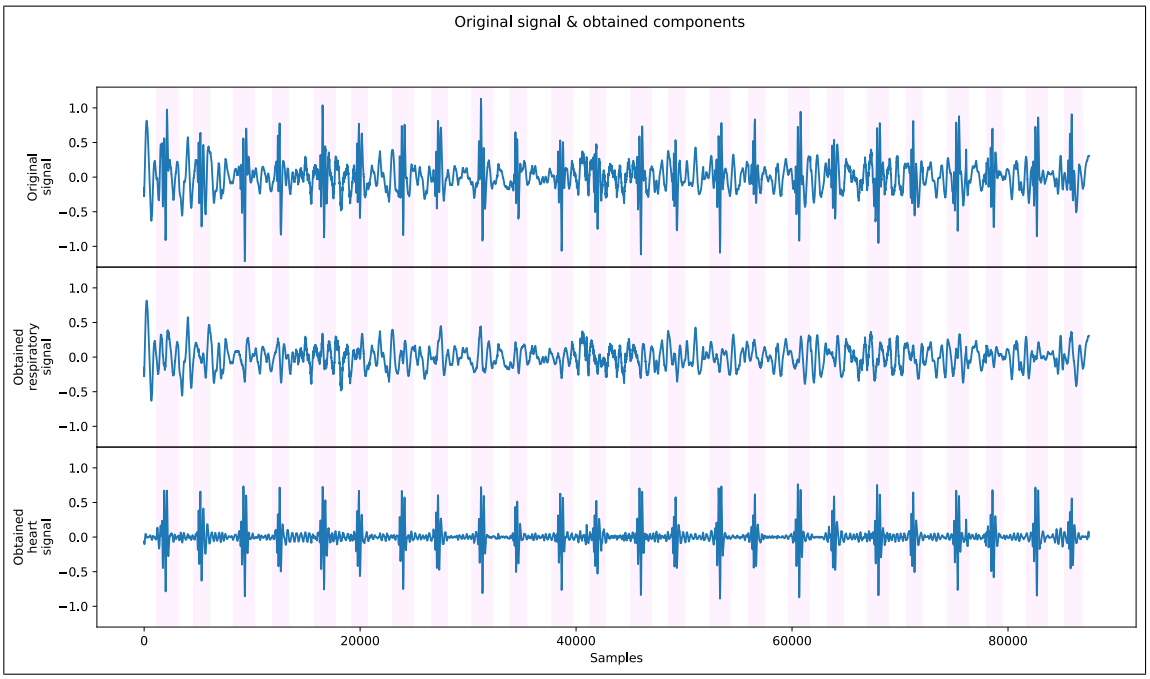

Figure 3.20: Separation of the original cardio-respiratory signal into its lung and heart sound components using NMF replacing segments with $\beta = 2$, $K = 2$, $N_{wind} = 2048$, 90% overlap and heart sound position criterion. The first graph shows the original cardio-respiratory signal, in which the heart sound segment is highlighted by the purple band. The second and third graphs show the lung and heart sounds respectively. This signal represents a portion of the 12^{th} audio file in the database.

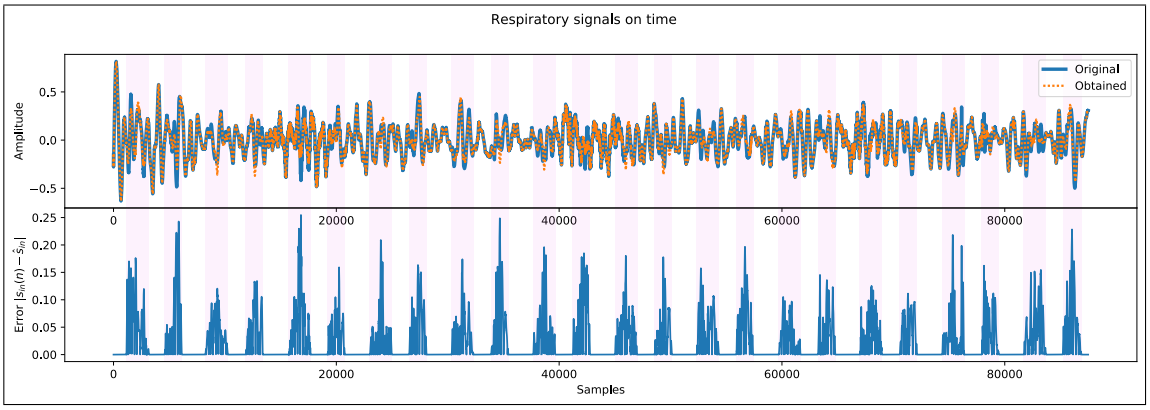

Figure 3.21: Comparison between original lung sound and obtained by NMF replacing segments with $\beta = 2$, $K = 2$, $N_{wind} = 2048$, 90% overlap and heart sound position criterion. The upper graph shows the original lung signal (blue line) and the obtained by decomposition (orange dotted line), highlighting in violet the heart sound segments. The lower graph shows the error between both signals. This signal represents a portion of the 12^{th} audio file in the database.

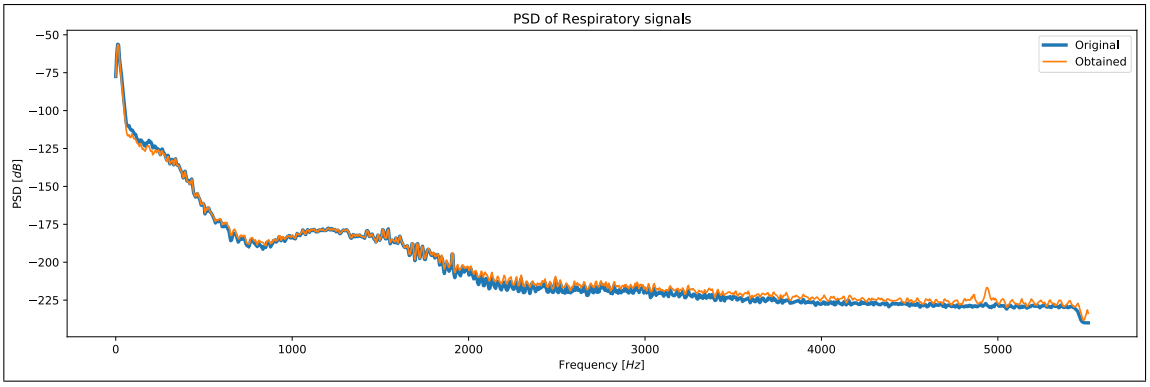

Figure 3.22: Comparison between the PSDs of the original lung sound and the obtained by NMF replacing segments with $\beta = 2$, $K = 2$, $N_{wind} = 2048$, 90% overlap and heart sound position criterion. The blue signal corresponds to original signal PSD, while the orange corresponds to PSD of the signal obtained by decomposition. For the PSD calculation, the Welch method was used with the same parameters used for decomposition. This signal represents a portion of the 12^{th} audio file in the database.

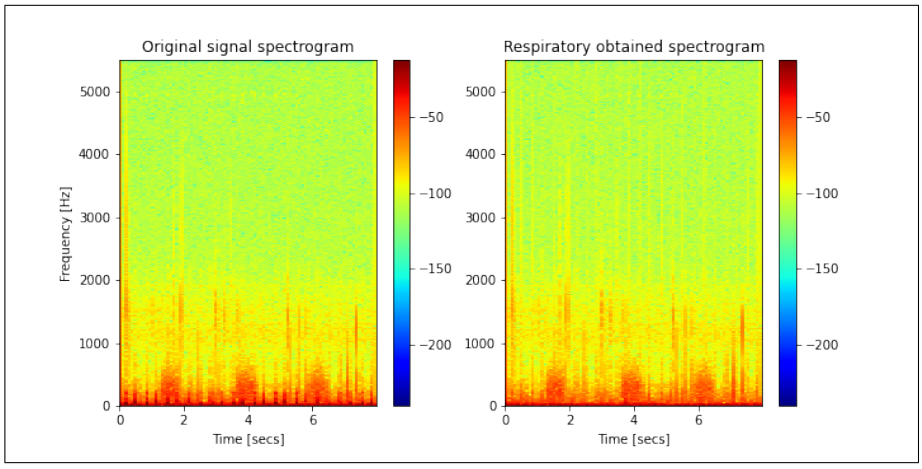

Figure 3.23: Comparison of the spectrograms between the original cardio-respiratory sound and the lung signal obtained by NMF replacing segments with $\beta = 2$, $K = 2$, $N_{wind} = 2048$, 90% overlap and heart sound position criterion. The spectrograms were obtained using the same parameters as for the decomposition. This signal represents a portion of the 12^{th} audio file in the database.

of heart sounds in the input signal, which could be an additional difficulty. In contrast, the NMF separation on heart sound segments has a lower performance than even the base case. This may be due that when performing the decomposition of a spectrogram obtained from such a short segment, errors may exist due to the fact that the restrictions of the range of the NMF decomposition are not accomplished and the low temporal extension of the spectrogram. Additionally, it could be because the decomposition only considers the information from that segment, so it could not have correctly detected the patterns with the properties of each interest signal. In the case of NMF masking heart sound positions, something similar happens, but using a greater number of segments simultaneously.

By analyzing the results, it is concluded that the criterion that considerably improves the performance of the separation is the heart sound position correlation. Using this criterion with $N_{wind} = 2048$ and 90% overlap to obtain the spectrogram, there is no clear trend for the number of components K and the β-divergence parameters in the objective function. However, when performing the analysis on all the simulations without restricting the assignment criteria or spectrogram parameters, it is possible to notice that $\beta = 2$ gives better results, while $K = 2$ is systematically better in terms of number of components.

One of the main difficulties of NMF decomposition is the wide variety of parameters that can be used. This in turn raises the idea of moving towards the realization of a system that allows automating the process of selecting parameters for the NMF decomposition. Either, the design of a robust source separation system that allows recovering heart and respiratory sound with a similar or better performance, but with the advantage of handling a smaller set of parameters.

This type of preprocessing could facilitate the extraction of relevant features used in medical diagnostics, and even improve the performance of machine (or deep) learning-based classifiers. The latter could generate great advances in the field of telemedicine, opening the opportunity to develop technology that considers topics such as the automatic diagnosis of respiratory and heart diseases.

Chapter 4

General conclusions and future work

4.1 Conclusions

This work aimed to design a preprocessing system for the signal auscultated in the patient's chest to be used in the diagnosis of respiratory diseases. For this, the use of two consecutive blocks is proposed: a heart sound detection block based on a CNN, and a source separation algorithm that returns lung and heart sounds.

In the heart sound detection stage, a CNN based on an encoder-decoder architecture is implemented for the detection of fundamental heart sounds, using various features at the input. A detailed analysis of the parameters that defines each feature and those that define the network architecture is performed. When simulating different combinations of parameters, the results of the network remain in a similar range for different metrics. A relevant result is that regardless of the input features used, this system output remains sufficiently robust. Even using the raw signal as input to the network, it is possible to obtain high performance. To demonstrate this fact, a k-fold cross-validation was performed with $k = 10$ folds, with which the robustness of the results was verified. At this same stage, a CNN based on classical architectures was also tested, obtaining slightly lower results (see chapter A in appendix). One of the main advantages of the encoder-decoder architecture is that it allows a point-wise classification of the signals, so that the output of the network will have the same number of points as the input signal. It is possible to conclude from this that CNNs based on encoder-decoder architecture is a powerful tool to tackle this type of problem.

In addition to being useful in source separation strategies focused on the heart sound segments (as presented in this work), this detection system could be useful for the diagnosis of heart diseases since it allow knowing the heart sounds positions, and hence the position and duration of the systolic and diastolic intervals. With this information, the presence of abnormal heart sounds such as heart murmurs, S_3 and

S_4 could be studied in any of these intervals, or even the durations of each interval, which could provide valuable information for an eventual diagnosis of the state of the heart. Therefore, this system alone could be a good starting point for a project based on automatic heart disease detection.

For the source separation stage, three decomposition methods that use NMF and the heart sound positions as reference information were proposed. These methods were compared with a baseline scenario in which an NMF decomposition on the entire signal was performed. In addition, three assignment criteria for the K components obtained from the decomposition are proposed. The method that obtained the best results was the one that performed an NMF decomposition on the entire signal and replaced the segments of the heart sound positions with the lung signal obtained. This method mixes the good properties of applying NMF over the entire signal when recognizing spectral patterns in the **W** matrix, while achieving the aim of not modifying the pure lung sound segments. However, it is dependent on the correct detection of heart sounds to operate properly. From these results, it is also possible to conclude that NMF is an adequate method to solve this type of source separation problems.

Another conclusion that can be obtained from this algorithm is that, in the case of using methods that consider only the information of the heart sounds (such as the NMF method on heart sound position and the NMF masking heart sound position), the results decrease considerably. This is because the information of the pure lung sound segments is not incorporated into the NMF algorithm, therefore, it is not possible to sketch a clearly recognizable pattern for the lung sound in the **W** and **H** matrices. This explains why even the base method shows better results than these methods in all metrics.

It is expected that this preprocessing system will provide better results in the diagnosis of respiratory diseases since the characteristics of the lung signal can be extracted without the interference of heart sounds. This is desirable, for example, in cases when use spectral features to classify a complete segment of the signal. In the ICBHI 2017 database presented for the source separation system, labels indicating the presence of wheezes and crackles on each respiratory cycle (both, inhalation and exhalation) are presented. Therefore, the presence of a sound such as cardiac that spectrally overlaps with the lung sound can introduce false correlations when performing the classification using features related to the signal spectrum, which is not desirable.

4.2 Future work

From this research, some lines of work emerge that could be carried out to improve the performance of this system.

With regard to the heart sound detection, the training of a CNN with an encoder-

decoder architecture on a database whose labels are manually corrected by health professionals could be implemented, such as the (C. Liu et al., 2016) database. This will ensure that the system is even more accurate when it comes to recognizing heart sounds and defining time boundaries. Also, it is always recommended to have a larger number of samples when using neural networks to ensure robust system performance against new samples.

It would be interesting to study and develop a method for diagnosing heart diseases or abnormalities based on the information provided by this system. The presence of abnormal sounds such as murmurs or non-fundamental heart sounds (such as S_3 and S_4) could be studied, and even related to a diagnosis of the heart. The timing of each phase of the cardiac cycle could also be studied to report problems related to heart rhythm. This implementation could even be expanded to develop applications that deliver this information in real time (or with a reasonable delay).

In relation to the source separation, it is expected that new strategies for assigning components will be designed to improve the results presented. In particular, could be considered the use of segment reconstruction techniques is proposed as an alternative method to the source separation to obtain lung sounds. One idea that might be interesting is to use the heart sounds positions in the input signal, remove those segments, and use compressed sensing to reconstruct them based on the remaining information. The authors recommend using random sensing matrices in this kind of applications (Kutyniok, 2013). However, because the nature of this problem is the reconstruction of a segment, there may be difficulties with this technique.

Finally, the design of a classification system for lung diseases for clinical applications is proposed as future work, or a system that detects symptoms or abnormalities in lung sounds. For this, an approach based on the detection of events within the signal is recommended (similar to that performed in the heart sound detection block) since it would allow to inform the precise location of sounds such as crackles or wheezes. To implement this, the CNN method based on encoder-decoder architecture can be useful, or even a combination with Recurrent Neural Networks (RNN).

The robust design of a system like this could have a positive impact in the field of telemedicine. Indeed, electronic stethoscopes could be designed with an embedded system, or connected to applications or platforms to perform analyzes that allow automatic symptoms detection and diagnosis of patients from the auscultation of respiratory sounds. This could generate great advantages in terms of the operation logistics of the health centers, since not only would it give patients the opportunity to make part of their diagnosis remotely, but it would also allow reduce the levels of occupational load of medical centers, especially in winter periods. This could mark a great improve in the development of modern medicine.

Appendix

Appendix A

Heart sound detection: More analysis results

In recent decades, neural networks have become a powerful tool to carry out various applications. One of the types of neural networks most used in signal and image processing are convolutional neural networks (CNN), first proposed in 1989 with the work of LeCun (Y. LeCun et al., 1989).

CNNs are a type of multilayer feedback network, where each layer uses filter banks of a certain size. These filters are adjusted via backpropagation as the network is trained. One of its characteristics is its ability to exploit the temporal or spatial correlation of the data due to its convolutional nature, from which the network learn the inherent characteristics of the input data (A. Khan et al., 2020).

Although CNN have been developed mainly for applications related to image processing (2D), in this chapter they will be modified to operate with audio signals (1D). This chapter will detail the implementation of a CNN network based on a classical architecture, its parameters and its results compared to a CNN based on an encoder-decoder architecture.

A.1 CNN based on classical architectures

In this proposal, classic CNN network architectures such as the LeNet-5 (Yann LeCun et al., 1998) or the AlexNet (Krizhevsky et al., 2012) are used as inspiration.

This type of architecture can be divided into three main sections (see figure A.1):

1. A series of convolutional layers with their respective activation functions, alternated with pooling layers.

2. A flattening layer that connects the output of the last convolutional layer to the next series of layers.

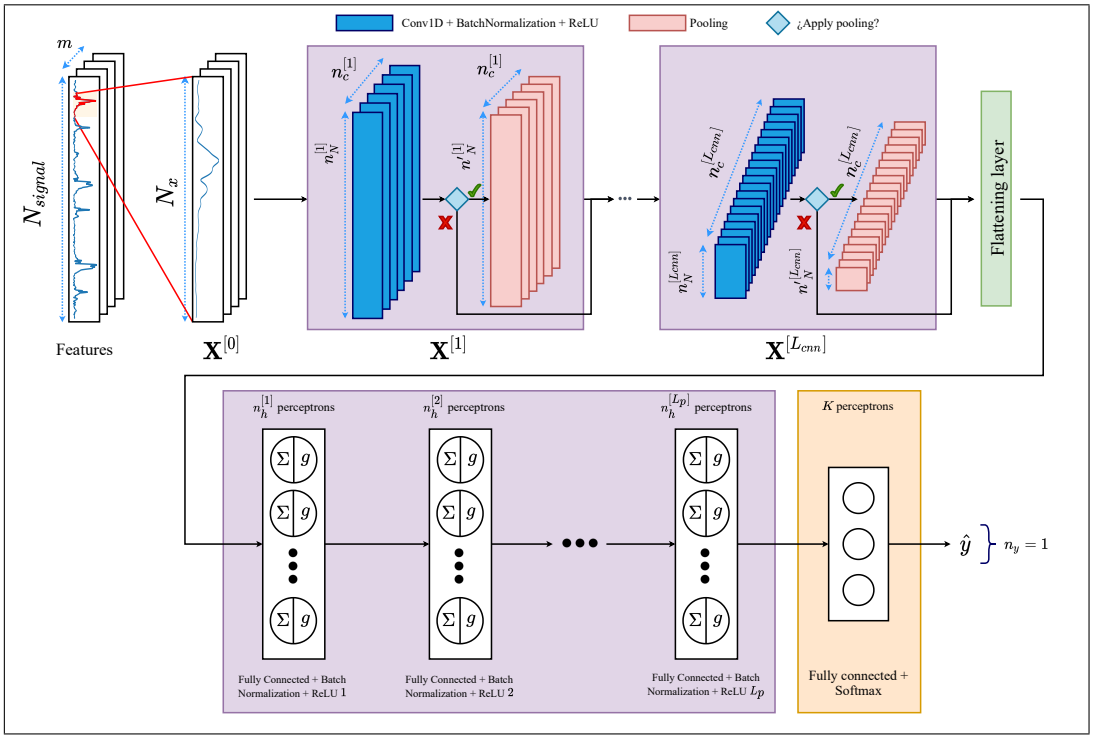

Figure A.1: General operation scheme of a CNN based on classical architectures.

3. A MLP with their respective activation functions. For the case of MLPs, the relationships between layers are defined by (Goodfellow et al., 2016):

$$\begin{aligned} a^{[l]} &= g^{[l]}\left({W^{[l]}}^T a^{[l-1]} + b^{[l]}\right) \\ &= g^{[l]}\left(z^{[l]}\right) \\ &= \forall l = \{1, ..., L\} \end{aligned} \tag{A.1}$$

Where $a^{[l]}, b^{[l]} \in \mathbb{R}^{n_h^{[l]}}$ correspond to the output vectors and their bias parameter for the $n_h^{[l]}$ perceptrons of the l-th layer, $g^{[l]}$ to their activation function and $W^{[l]} \in \mathbb{R}^{(n_h^{[l-1]} \times n_h^{[l]})}$ corresponds to the weight matrix used by the perceptrons in the l-th layer.

As can be seen, in this type of implementation $n_c^{[0]} = m$ features envelopes are used, obtained through different methods detailed in section C.1. For each audio sample in the database, the m signal features are windowed in segments of length N_x with step τ_x, obtaining the input $\mathbf{X}^{[0]} \in \mathbb{R}^{N_x \times m}$ to the network. The first part of the network is made up of L_{cnn} convolutional layers alternated with pooling layers. Note from figure A.1 that, in the implementation used for this chapter, each convolutional layer will be immediately followed by a batch normalization and then

a ReLU activation function. However, in the case of pooling layers, it will be decided if it is included for each output of the convolutional layers mentioned above. Once the L_{cnn} convolutional layers are finished, a flattening layer is applied that allows transforming an array $\mathbf{X}^{[L_{cnn}]} \in \mathbb{R}^{n_N'^{[L_{cnn}]} \times n_c^{[L_{cnn}]}}$ to a one-dimensional array with $(n_N'^{[L_{cnn}]} \cdot n_c^{[L_{cnn}]})$ inputs, which allows to connect the convolutional stage with the MLP stage. Then L_p perceptron layers are applied followed by batch normalization and ReLU activation. Finally, a layer of K perceptrons is used (for the K classes of interest) with softmax function, which will indicate the probability of membership that the segment of length N_x has for those K classes. The output $\hat{y} \in \{1, ..., K\}$ will indicate the class with the highest probability predicted on the input segment. Therefore, this type of network reduces a segment of length N_x to a label of length 1.

A.2 Initial network design

This network is presented in the figure A.2. As indicated, this network begins with 4 consecutive layers consisting of a convolutional filter bank, followed by batch normalization and ReLU activation for each one. For convolutional filters, a bank of $n_c^{[l]} = 20$ filters of length $H_l = 200$ is used for each of these layers. Furthermore, these convolutional layers are padded in such a way that the output maintain the length of the input. This is achieved by setting the `padding = valid` option in Keras `Conv1D` layer. In one of the variations of this network, a maxpooling layer is included after ReLU activation, reducing the length of the input segments s_{in} in half each time these blocks are presented.

Then, a flattening layer is used to adjust the output of the CNN stage for use as input in the MLP stage. This consists of 3 layers of perceptrons with batch normalization and ReLU activation, where each layer has $n_h^{[l]} = 50$ perceptrons.

Finally, unlike the works presented in the literature where 4 classes are used (S_1, systole, S_2, diastole), the output will consist of a softmax layer with $K = 3$ classes: S_1, S_2 and non-heart sound (S_0).

A proposed variant is the use of a different channel for each of the m descriptors used. A diagram of this type of network is shown in figure A.3. As can be seen, each feature is entered independently to each channel network represented by the red box in figure A.2.

At the output of each of the channels of this network, a concatenation layer is used to join the output of each channel in a one-dimensional arrangement. Finally, the output of this layer is connected to a softmax layer with the $K = 3$ classes defined above.

This network was built using the Keras API built into Python's Tensorflow library (Abadi et al., 2016). All parameters of convolutional layers (defined by the `Conv1D` function of the `tensorflow.keras` environment) and of perceptrons (defined by the

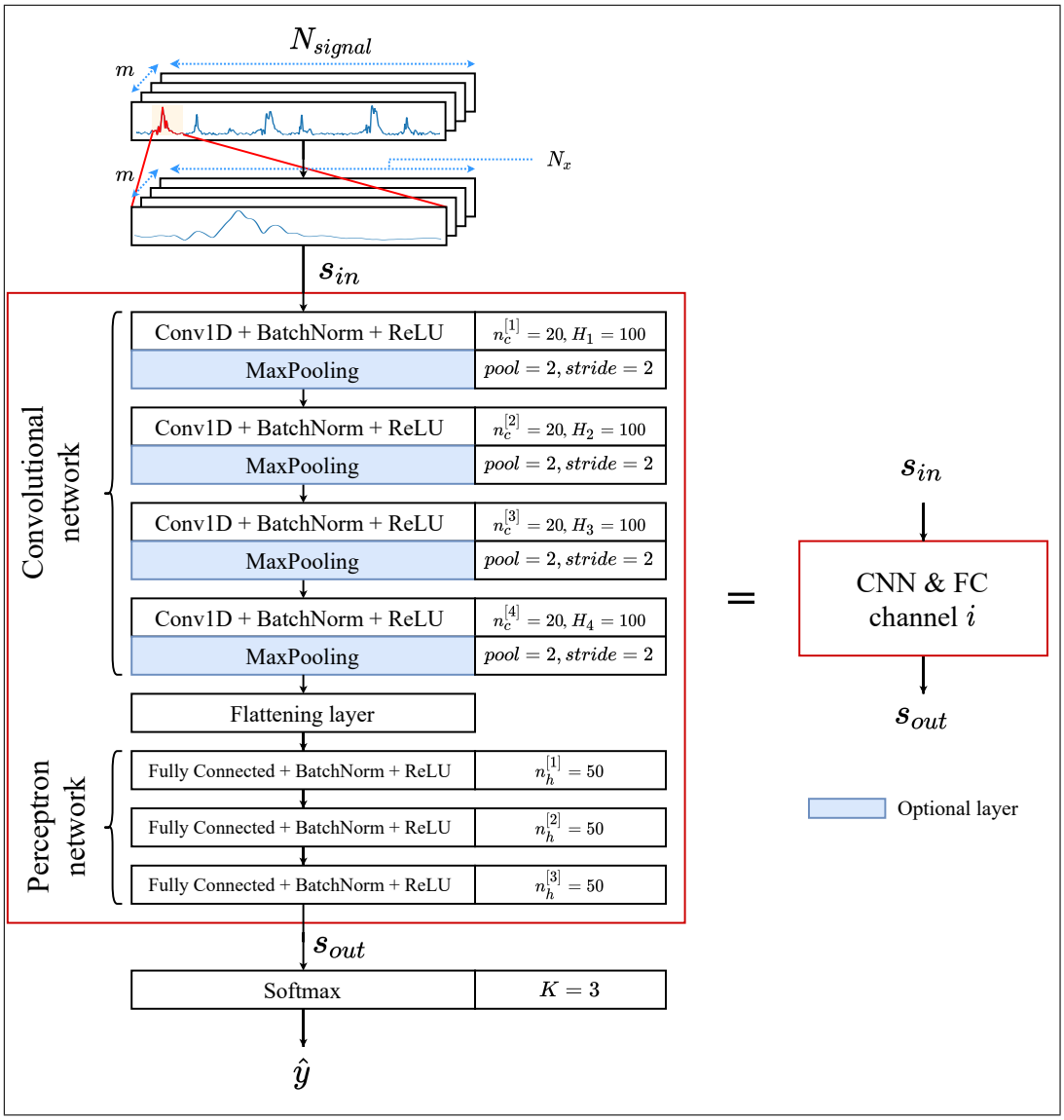

Figure A.2: Proposed CNN based on classical architectures. In this network we consider a convolutional stage with optional maxpooling (available for one of the variants to be tested), then a flattening layer and finally a MLP stage. The output of this last layer is connected to a softmax layer. The red box synthesizes this network into a single block, which will be used in the multichannel network diagram.

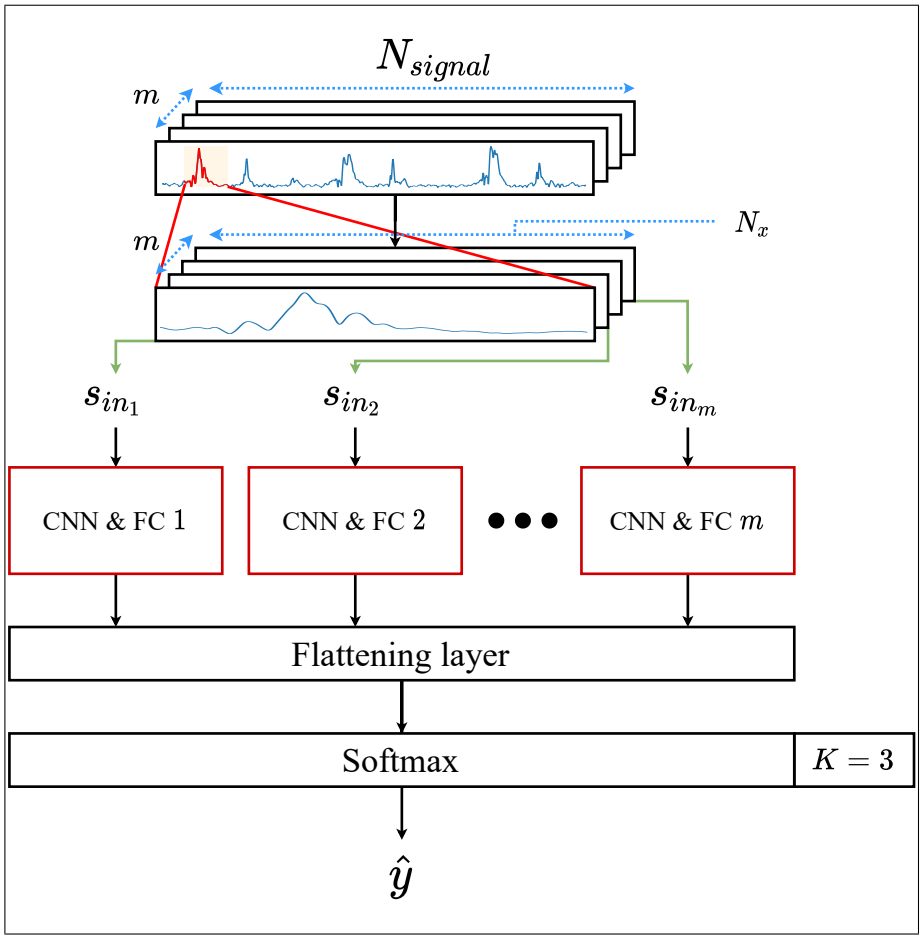

Figure A.3: Diagram of the network using multiple channels. In this type of network, a subnet for each feature in the input is used. The output of each network is concatenated to be the input of a softmax layer.

Table A.1: Results of CNN based on classical architectures. For each of the metrics used, the architecture that achieves the best performance is highlighted in green.

CNN & MLP, $N_x = 128, \tau_x = 16$													
Input channels	Maxpool	Trainable parameters	Train				Validation				Test			
			Accuracy	Recall	Precision	F_1	Accuracy	Recall	Precision	F_1	Accuracy	Recall	Precision	F_1
1	No	389843	89.502	88.462	90.490	89.464	87.963	86.877	89.046	87.948	88.376	87.362	89.403	88.371
4	No	1511363	90.269	89.347	91.221	90.274	87.408	86.584	88.374	87.470	87.872	87.124	88.708	87.909
1	Yes	293843	89.347	88.306	90.419	89.350	86.998	85.718	88.241	86.961	88.052	86.847	89.253	88.034
4	Yes	1127363	90.673	89.828	91.549	90.681	86.683	85.732	87.602	86.657	87.922	87.124	88.846	87.977

`Dense` function of the `tensorflow.keras` environment) presented will be initialized using the `'he_normal'` method described in (He et al., 2015).

To train the networks, the Adam optimization algorithm (from Adaptive Moment Estimation) (Kingma and Ba, 2015) is used on a cross-entropy cost function. $\beta_1 = 0.9$ (associated with momentum), $\beta_2 = 0.999$ (associated with RMSprop) and a learning rate of $\alpha = 0.001$ are used. The network will be trained for 20 epochs using batches of 70 segments of dimension (N_x, m), where N_x corresponds to the length of each segment and m to the number of features to use.

A.3 Architecture analysis

In this section, CNN based on classical architectures and CNN with encoder-decoder architecture will be compared, using the same parameters described in this chapter and incorporating the multi-channel option for each architecture.

For each of the experiments performed in this section, the same features defined in works such as (Renna et al., 2019; Springer et al., 2016) will be used at the network input, which are: homomorphic filter envelope, Hilbert magnitude envelope, spectral energy and DWT on one level.

In the case of networks based on classical architectures, the parameters to vary are: the option of implementing a shared channel or a channel for each feature; and the option to incorporate maxpooling layers at the output of the activation functions for each layer in the convolutional stage. For each mix of parameters, windows of length $N_x = 128$ points with step $\tau_x = 16$ on the input signal will be used to define segments. The results of this network are presented in table A.1.

As can be seen, the architecture that best fits the data in the training is the one that uses one channel for each feature and implements maxpooling. However, this network does not obtain the best results on the validation and testing set. Indeed, the network that uses a shared channel for all features and does not incorporate maxpooling layers obtains better performance in validation and testing. This could indicate that the latter network allows for a better generalization of the heart sound characteristics. The multi-channel network with maxpooling is slightly more overfitting in comparison. However, and in general, the results are quite homogeneous.

In the case of CNN with encoder-decoder architecture, the parameters to vary

Table A.2: Results of CNN using encoder-decoder architectures. For each of the metrics used, the architecture that achieves the best performance is highlighted in green.

CNN Encoder decoder													
Input channels	(N_x, τ_x)	**Trainable parameters**	**Train**				**Validation**				**Test**			
			Accuracy	**Recall**	**Precision**	F_1	**Accuracy**	**Recall**	**Precision**	F_1	**Accuracy**	**Recall**	**Precision**	F_1
1	(128, 16)	653422	87.911	86.394	89.398	87.870	87.257	85.519	88.908	87.181	87.944	86.345	89.553	87.920
4	(128, 16)	2582479	89.074	87.834	90.316	89.058	86.483	85.002	87.927	86.440	87.589	86.349	88.892	87.602
1	(1024, 64)	653422	95.161	95.123	95.200	95.162	92.909	92.839	92.990	92.915	93.463	93.421	93.512	93.467
4	(1024, 64)	2582479	96.698	96.684	96.712	96.698	92.874	92.825	92.931	92.878	93.657	93.623	93.699	93.661

are: the option of implementing a shared channel or a channel for each envelope; and the length N_x and step τ_x for the definition of the windows of the input signal in the network. The results of this network are presented in table A.2. From these results, it can be seen that for the case in which larger input windows are considered, the performance of the network improves considerably. It should be noted that although there is an improvement in using a channel for each descriptor, it does not constitute a significant improvement to the performance of the network. Furthermore, the number of trainable parameters increases approximately $m = 4$ times due to each channel, where m corresponds to the number of features used in the input.

Comparing both types of networks, it is possible to notice that for $N_x = 128$ and $\tau_x = 16$, the networks in general present similar results. However, when trying with $N_x = 1024$ and $\tau_x = 64$ the CNN based on the encoder-decoder architecture gives much better results. One of the qualitative advantages of this type of network is that it does not present resolution problems when classifying. Indeed, since each point in this network is classified independently, the N_x points at the input are classified resulting in N_x points at the output. In comparison, the CNN based on classical architectures reduces the N_x points at input to a single point, which could lead to resolution problems. It is for this reason that it is tested with $N_x = 1024$ in encoder-decoder networks and not in classical ones.

Because of this, the encoder-decoder architecture is selected. Furthermore, the use of a single channel architecture is preferred as it achieves quite similar results using considerably fewer trainable parameters, compared to the multi-channel network. From this point, it will continue to experimenting only with this network.

A.4 Class balance analysis

In (Badrinarayanan et al., 2017) one of the techniques used when training the SegNet network is class balancing when the labels in the datasets are imbalanced. Here we will try weighting each of the classes in the objective function considering their frequency of appearance in the database. As in (Badrinarayanan et al., 2017), in this work the median frequency balance will be used, which defines each class weight

Table A.3: Results of class balancing analysis for the weights in the objective function of the CNN-based encoder-decoder architecture. For each of the metrics used, the combination of input features that obtains the best performance is highlighted in green. The first row corresponds to the base architecture obtained from the previous analyses.

CNN Encoder-Decoder ($N_x = 16384$, $\tau_x = 128$, $n_c^{[l]} = 15$, $H^{[l]} = 150$, $c^{[l]} = [2, 2, 3, 3]$)												
Class balancing	**Train**				**Validation**				**Test**			
	Accuracy	**Recall**	**Precision**	F_1	**Accuracy**	**Recall**	**Precision**	F_1	**Accuracy**	**Recall**	**Precision**	F_1
Natural	98.885	98.884	98.886	98.885	95.567	95.563	95.570	95.566	94.050	94.046	94.058	94.052
Median	97.791	97.787	97.796	97.791	94.397	94.385	94.409	94.397	93.685	93.667	93.701	93.684

as:

$$\alpha_k = \frac{\text{median}(\#k)}{\#k}, \forall k = \{1, ..., K\} \tag{A.2}$$

Where $\#k$ corresponds to how many elements the class k are in the dataset. This expression allows assigning a greater weight α_k to the classes that have fewer instances, while the most frequent classes of the base will have a lower weight in the objective function. It should be noticed that only the training set was used to calculate $\#k$. The result of this implementation is presented in table A.3.

As can be seen, for all the metrics used the implementation of the natural frequency balance (without weighting) exceeds the median frequency balance implementation. Considering this results, it was decided not to use a class balance for this problem since it does not provide a significant improvement.

A.5 Skipping connections analysis

The study of (Ye and Sung, 2019) on encoder-decoder architectures points out that the use of information coming from the encoder in the decoding stage can increase the expressive power of networks. In this section, two skipping methods will be tested. One of them will realize the concatenation between the output of the upsampling layer of the decoder and the input of the maxpooling layer of its corresponding encoder (as proposed in the work of (Renna et al., 2019) that uses encoder-decoder architecture based on the U-net); while the second option will communicate the encoder with the decoder through the sum (as proposed in (Ye and Sung, 2019)).

The implementation of this idea can be seen in the skipped connections in figure 2.1, whose details are presented in figure A.4. The results of these experiments are presented in table A.4.

From these results it is possible to notice that skipping operations do not present a great improvement in network performance. Indeed, for each of the proposed variations, the results do not vary significantly. For this reason, it is decided to continue

Table A.4: Results of the analysis of the skipping connections between the encoder and decoder layers of the CNN based encoder-decoder architecture. For each of the metrics used, the combination of input features that obtains the best performance is highlighted in green. The first row corresponds to the base architecture obtained from the previous analyses.

CNN Encoder-Decoder ($N_x = 16384$, $\tau_x = 128$, $n_c^{[l]} = 15$, $H^{[l]} = 150$, $c^{[l]} = [2, 2, 3, 3]$)												
Skipping type	**Train**				**Validation**				**Test**			
	Accuracy	**Recall**	**Precision**	F_1	**Accuracy**	**Recall**	**Precision**	F_1	**Accuracy**	**Recall**	**Precision**	F_1
No	98.885	98.884	98.886	98.885	95.567	95.563	95.570	95.566	94.050	94.046	94.058	94.052
Sum (Badrinarayanan et al., 2017)	98.652	98.650	98.655	98.652	94.607	94.602	94.612	94.607	94.296	94.285	94.307	94.296
Concatenate (Renna et al., 2019)	98.779	98.777	98.780	98.779	95.100	95.096	95.107	95.101	94.250	94.243	94.258	94.251

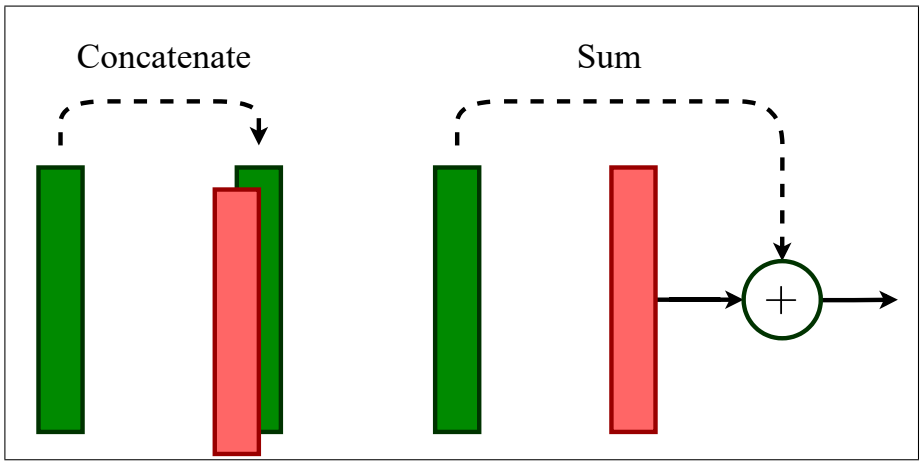

Figure A.4: Connections between the corresponding encoder and decoder layers. Two possibilities are presented: concatenate and sum of the corresponding maxpooling-upsampling layer pairs.

using the encoder-decoder network without the inclusion of operations between the maxpooling layers at encoder and the upsampling layers at decoder.

Appendix B

Source separation: More analysis results

B.1 Spectrogram parameters

The window size N_{wind} and overlap used to obtain the spectrogram will be studied, in order to reduce the dimensionality of the analysis.

Reviewing the results presented in the histograms in the figure B.1 for different window sizes it is possible to notice that when $N_{wind} = 1024$ and $N_{wind} = 2048$, better results are obtained independently of the decomposition method used. Indeed, they concentrate a large part of their results in high values for the case of temporal correlation and SDR, while for the MSE they tend to concentrate on values close to zero. In relation to the spectral correlation, there is not a great variation between the different options, so it will not be considered in this analysis. Since for $N_{wind} = 2048$ the results tend to be slightly more concentrated in high values of temporal correlation and SDR (low for MSE), it will be defined as the size of the window to be used to obtain the spectrogram.

Regarding the overlap used, in the histograms of figure B.2 it is possible to see that for the base NMF and segment-replacing NMF methods, the decision of the amount of overlap presents relatively similar results. However, for the NMF on heart sound segments method, the results decrease considerably if an overlap of 50% is used (especially in terms of temporal correlation and SDR), and they improve slightly when the overlap is 90% compared to the 75% overlap. In the case of NMF method on masked signal, there is not a clear enough pattern to indicate whether the option of 75% or 90% overlap is better, but it is possible to appreciate that for the temporal correlation, MSE and SDR, the option 50% overlap produces slightly worse results. For these reasons, it was decided to use a 90% overlap to obtain the spectrogram of the signals.

Therefore, and from this point on, a window size of $N_{wind} = 2048$ points and an 90% overlap will be used to obtain the spectrograms.

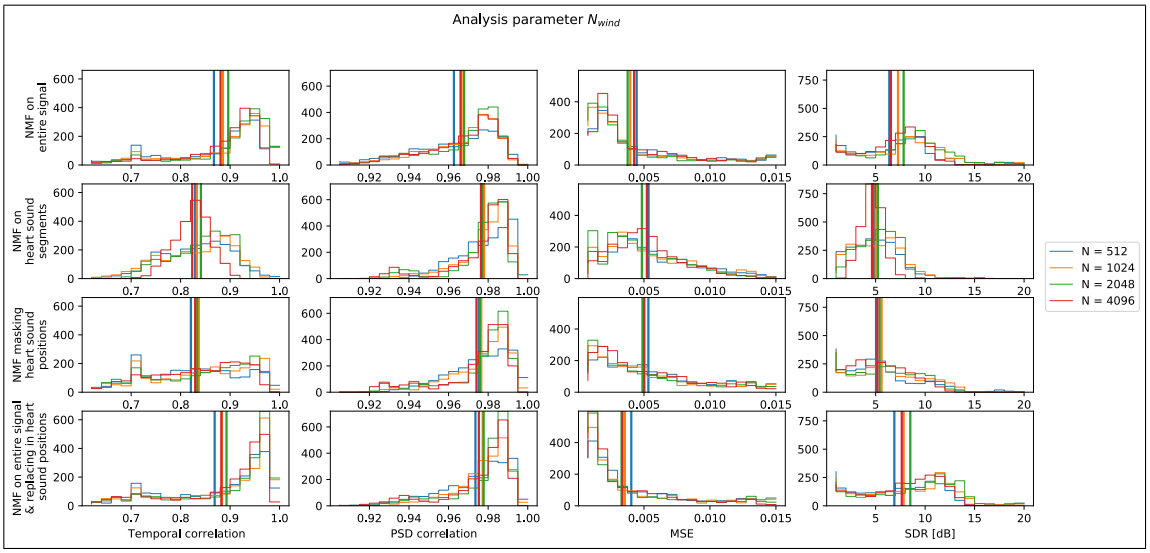

Figure B.1: Histogram of results for $N = \{512, 1024, 2048, 4096\}$ using the temporal correlation, PSD correlation, MSE and SDR metrics for each type of NMF decomposition. The vertical lines indicates the mean of each histogram.

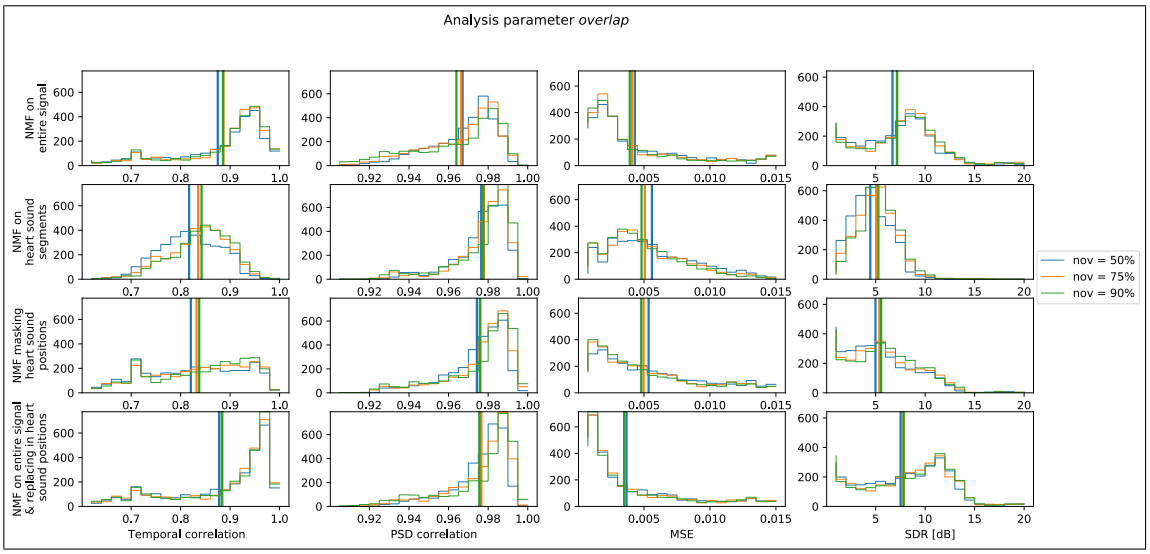

Figure B.2: Histogram of results for overlap of {50%, 75%, 90%} using the temporal correlation, PSD correlation, MSE and SDR metrics for each type of NMF decomposition. The vertical lines indicates the mean of each histogram.

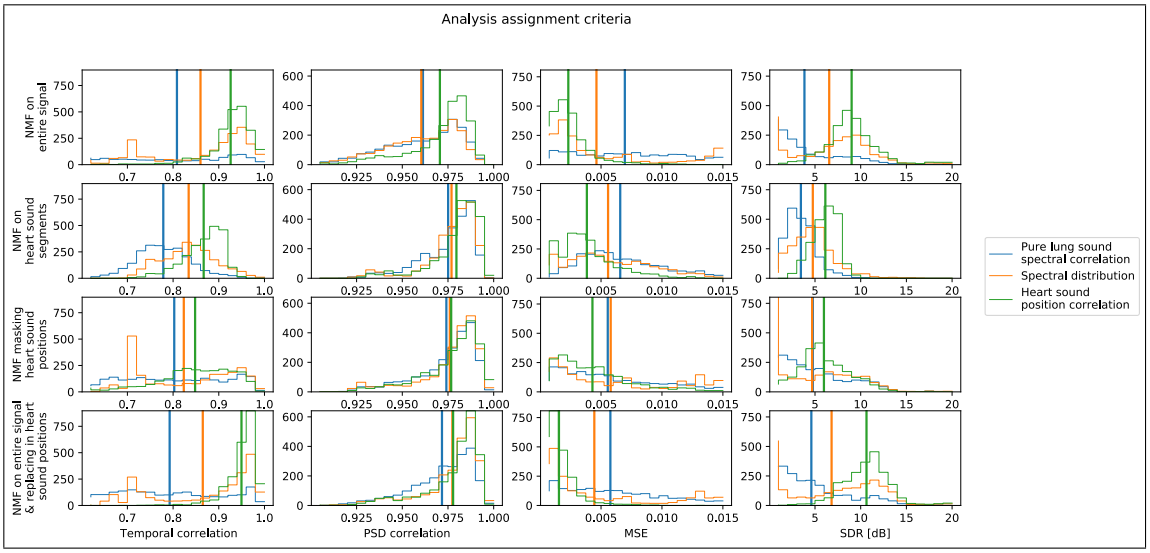

Figure B.3: Histogram of results for assignment criteria using the temporal correlation, PSD correlation, MSE and SDR metrics for each type of NMF decomposition. The vertical lines indicates the mean of each histogram.

B.2 Simulation results over all parameters

This section presents the simulations performed considering all the combinations of the parameters presented in this study: N_{wind}, overlap, number of components K, cost function β-divergence, decomposition method of using NMF and component assignment criteria.

Below are the histograms of the results, differentiating each of the metrics used on the horizontal axis, each decomposition method on the vertical axis, and using a different color for each of interest parameter in histogram.

B.3 Simulation results restricted to heart sound position correlation criterion

This section presents the results of the simulations performed, restricting the use of windows of size $N_{wind} = 2048$, with 90% overlap and heart sound position correlation as assignment criterion. The objective of these results is to know what is the behavior of the system against certain previous decisions for the spectrogram parameters and the assignment criteria. Each one is presented below.

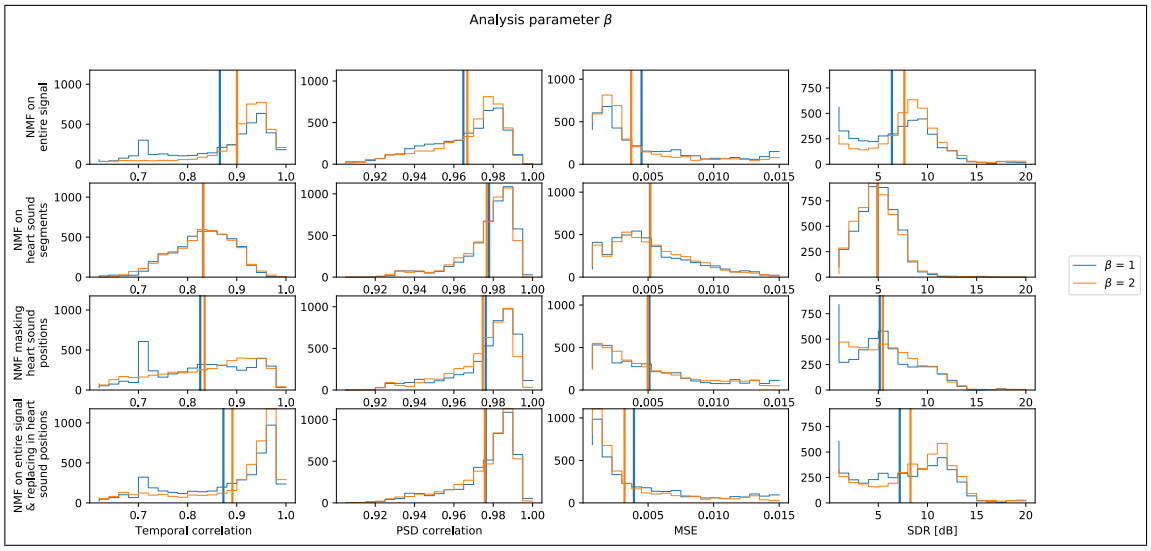

Figure B.4: Histogram of results for $\beta = \{1, 2\}$ using the temporal correlation, PSD correlation, MSE and SDR metrics for each type of NMF decomposition. The vertical lines indicates the mean of each histogram.

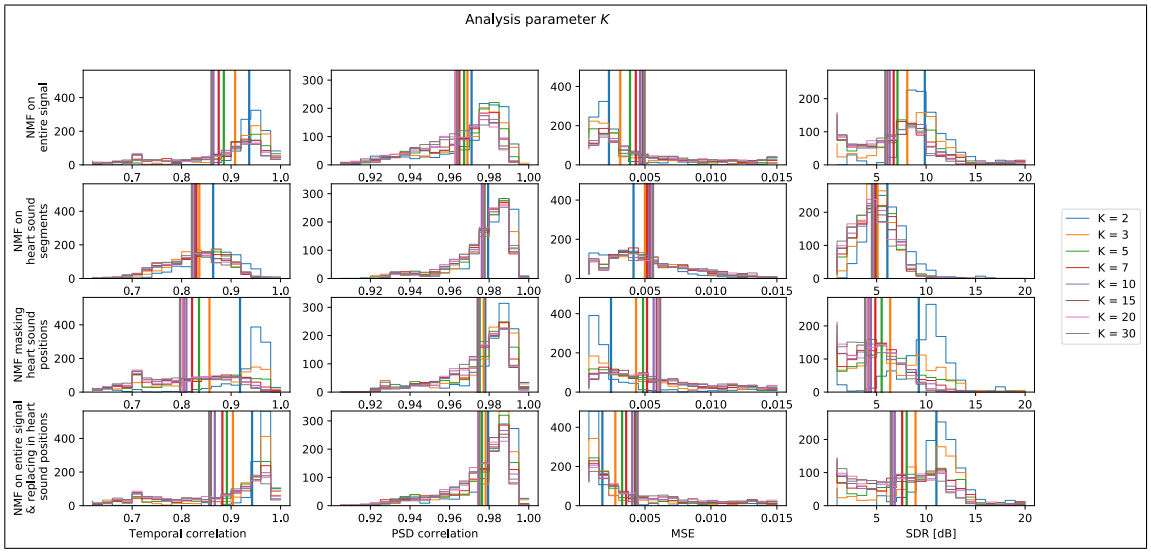

Figure B.5: Histogram of results for the number of components $K = \{2, 3, 5, 7, 10, 15, 20, 30\}$ using the temporal correlation, PSD correlation, MSE and SDR metrics for each type of NMF decomposition. The vertical lines indicates the mean of each histogram.

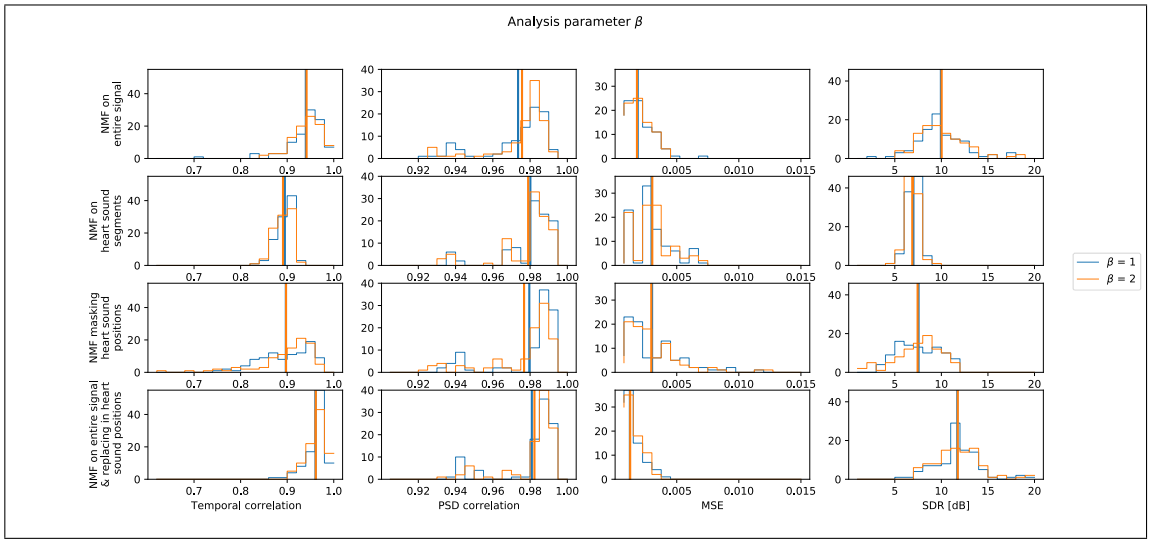

Figure B.6: Histogram of results for $\beta = \{1, 2\}$ using the temporal correlation, PSD correlation, MSE and SDR metrics for each type of NMF decomposition. The vertical lines indicates the mean of each histogram. It should be noted that this histogram incorporates only the simulations that use $N_{wind} = 2048$, 90% overlap and heart sound position correlation as assignment criterion.

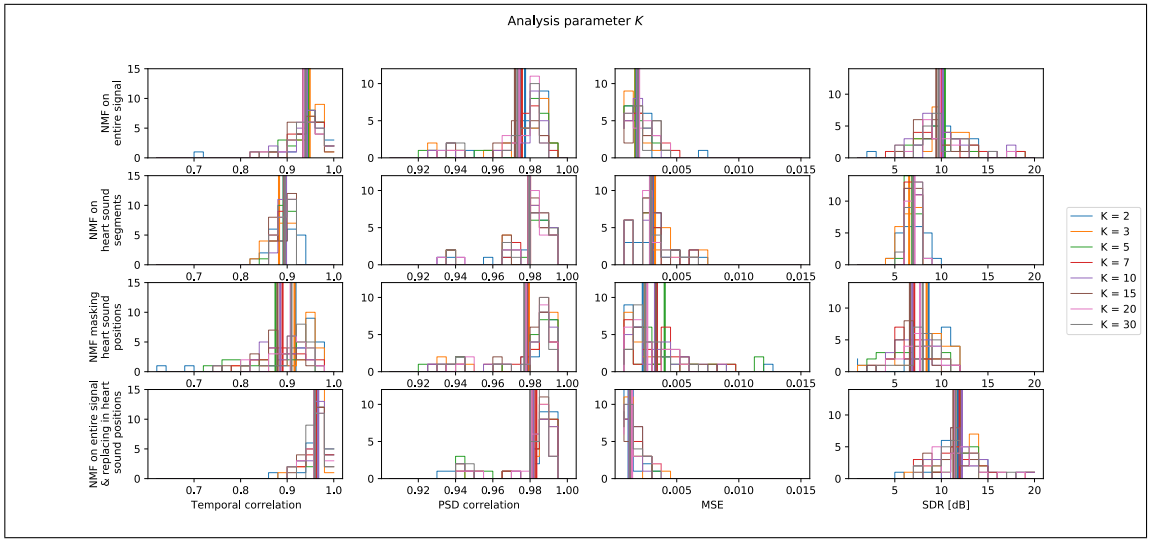

Figure B.7: Histogram of results for the number of components $K = \{2, 3, 5, 7, 10, 15, 20, 30\}$ using the temporal correlation, PSD correlation, MSE and SDR metrics for each type of NMF decomposition. The vertical lines indicates the mean of each histogram. It should be noted that this histogram incorporates only the simulations that use $N_{wind} = 2048$, 90% overlap and heart sound position correlation as assignment criterion.

Appendix C

Theory and analysis of input features at CNN

C.1 Features implemented

For the training of neural networks it is necessary to obtain features that will be used as input. The theoretical foundations that will explain the features used in the chapter 2 are presented below.

C.1.1 Homomorphic filters

In this type of filter, the signal $s(n)$ is modeled as the multiplication of a low-frequency signal $a(n)$ (associated with the envelope) and a high-frequency signal $f(n)$, just like in an AM modulation:

$$s(n) = a(n)f(n) \tag{C.1}$$

From which it is defined:

$$z(n) = \log(|s(n)|) = \log(|a(n)|) + \log(|f(n)|) \tag{C.2}$$

Applying a linear low pass filter L it is then possible to define:

$$\begin{aligned} z_{low}(n) = L\{z(n)\} &= L\{\log(|a(n)|) + \log(|f(n)|)\} \\ &= L\{\log(|a(n)|)\} + L\{\log(|f(n)|)\} \\ &= L\{\log(|a(n)|)\} \end{aligned} \tag{C.3}$$

Finally, by applying the exponential function it is possible to obtain the low frequency component $a(n)$ that can be used as an envelope:

$$|a(n)| = e^{z_{low}(n)} \tag{C.4}$$

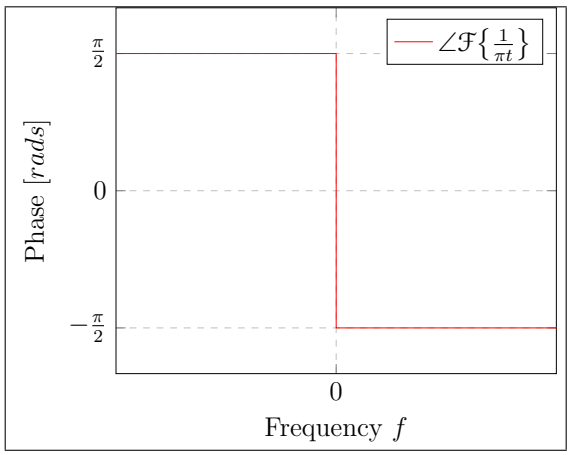

Figure C.1: Phase frequency response of the Hilbert transform.

C.1.2 Hilbert transform

The Hilbert transform of a real continuous-time signal $s(t)$ is defined as (Choi and Jiang, 2008; Feldman, 2008; Varghees and Ramachandran, 2014):

$$\mathcal{H}\{s(t)\} = \frac{1}{\pi}\int_{-\infty}^{\infty}\frac{s(\tau)}{t-\tau}d\tau = s(t) * \frac{1}{\pi t} \tag{C.5}$$

Where the operator "$*$" corresponds to the convolution. Analyzing the frequency response of the expression $\frac{1}{\pi t}$ we have:

$$\mathcal{F}\left\{\frac{1}{\pi t}\right\} = -j \cdot \text{sgn}(f) = \begin{cases} j & \text{if } f < 0 \\ 0 & \text{if } f = 0 \\ -j & \text{if } f > 0 \end{cases} \tag{C.6}$$

Where j is the imaginary unit, that is, $j = \sqrt{-1}$. From this it is possible to notice that the Hilbert transform modifies the phase of the signal without modifying its amplitude. Indeed, for $f < 0$ it shifts the phase of the signal by $\pi/2$, while for $f > 0$ it generates a phase shift of $-\pi/2$, which can be illustrated in figure C.1.

For the case of discrete signals, it is possible to express the Hilbert transform of the signal $s(n)$ as (Nivitha Varghees and Ramachandran, 2017):

$$\mathcal{H}\{s(n)\} = \text{IDFT}\{S_{\mathcal{H}}(k)\} \tag{C.7}$$

Y:

$$S_{\mathcal{H}}(k) = \begin{cases} -jS(k) & k = 0, 1, \ldots, \frac{N}{2} - 1 \\ jS(k) & k = \frac{N}{2}, \frac{N}{2} + 1, \ldots, (N-1) \end{cases} \tag{C.8}$$

Where IDFT$(\cdot)$ corresponds to the Inverse Discrete Fourier Transform, and $S(k)$ corresponds to the DFT (Discrete Fourier Transform) of the signal $s(n)$.

An analytical signal $s_a(n)$ is a type of signal whose frequency spectrum is only defined in non-negative frequencies (Smith, 2007). Using the Hilbert transform, it is possible to define an analytical signal from a real signal $s(n)$ by (Choi and Jiang, 2008; Feldman, 2008; Smith, 2007; Varghees and Ramachandran, 2014):

$$s_a(n) = s(n) + j \cdot \mathcal{H}\{s(n)\} \tag{C.9}$$

Since the analytical signal is a complex signal, it is possible to express it in its polar form:

$$s_a(n) = A(n)e^{j\phi(n)} \tag{C.10}$$

Where $A(n)$ is the amplitude of the analytical signal $s_a(n)$, and is defined as:

$$A(n) = |s_a(n)| = \sqrt{s(n)^2 + \mathcal{H}\{s(n)\}^2} \tag{C.11}$$

While $\phi(n)$corresponds to its instantaneous phase, and is defined as:

$$\phi(n) = \angle s_a(n) = \arctan\left(\frac{\mathcal{H}\{s(n)\}}{s(n)}\right) \tag{C.12}$$

To understand the properties of the amplitude envelope presented by $A(n)$, consider a discrete amplitude-modulated (AM) sinusoidal signal $s(n) = A(n)\cos(2\pi fn)$, where $A(n)$ corresponds to the signal modulated at a carrier frequency f, whose rate of variation of $A(n)$ is much less than f (Smith, 2007). Applying the Hilbert transform, it is possible to obtain that $\mathcal{H}\{s(n)\} = A(n)\sin(2\pi fn)$. Therefore, the analytical signal would be given by:

$$s_a(n) = A(n)\big[\cos(2\pi fn) + \sin(2\pi fn)\big] = A(n)e^{j2\pi fn}$$

Then:

$$|s_a(n)| = A(n)$$

Therefore, and understanding the signal $s(n)$ as a sum of sinusoids, through the amplitude $A(n)$ of the analytical signal $s_a(n)$ presented in (C.11) it is possible obtain an envelope from the original signal.

C.1.3 Wavelet transform

The discrete Wavelet transform (DWT) of a signal $s(n)$ is defined from a scaling function $\phi_{j,k}(t)$ and a Wavelet function $\psi_{j,k}(t)$ as (C.-L. Liu, 2010):

$$\begin{aligned} W_\phi(j,k) &= \frac{1}{\sqrt{M}}\sum_n s(n)\phi_{j,k}(n) \\ W_\psi(j,k) &= \frac{1}{\sqrt{M}}\sum_n s(n)\psi_{j,k}(n) \end{aligned} \tag{C.13}$$

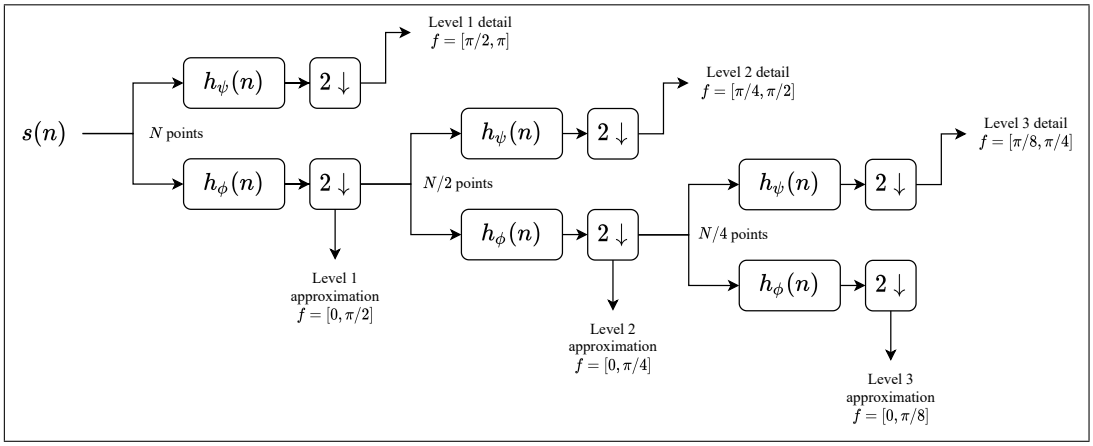

Figure C.2: Esquema de descomposición DWT en múltiples niveles.

Where j corresponds to the scale parameter of the transform, k to the translation parameter, $W_\phi(j,k)$ corresponds to the approximation coefficients and $W_\psi(j,k)$ to the detail coefficients of the transform. It is possible to express the relationship between different scales of the form (for more details on the basis of these relationships, it is recommended to review (C.-L. Liu, 2010)):

$$\begin{aligned} W_\phi(j,k) &= g_\phi(n) * W_\phi(j+1, 2k) \\ W_\psi(j,k) &= g_\psi(n) * W_\phi(j+1, 2k) \end{aligned} \tag{C.14}$$

Where $g_\phi(n)$ and $g_\psi(n)$ can be modeled as the filters in figure C.2. As can be seen, for each level the number of points is reduced by half. A variant of the DWT is the Stationary Wavelets Transform (SWT) which performs the same process, but without the downsampling step. This makes it possible to conserve the number of points in the different levels of decomposition, being able to make point-to-point comparisons between the different levels of decomposition, and implement techniques such as the Multi-scale Wavelets product.

C.1.4 Multi-scale Wavelet product

A useful property of the multiplication of adjacent levels in a SWT is that it allows to attenuate the presence of white noise, and to enhance the maximum values that propagate through the scales, which could allow the identification of signals like the heart sounds in PCG (Bao and L. Zhang, 2003; Yadollahi and Z. M. Moussavi, 2006). This property is used in applications such as highlighting edges of an image (Bao and L. Zhang, 2003) and denoising (Flores-Tapia et al., 2007). In works such as (Flores-Tapia et al., 2007; Yadollahi and Z. M. Moussavi, 2006) this type of technique is used to detect the singularities that represent the heart sounds in a PCG.

The Multi-scale Wavelets product between the levels j and $(j+k)$ is defined as (Flores-Tapia et al., 2007):

$$P_{j,k}(s(n)) = \prod_{i=k}^{k+j} \mathcal{W}_i(s(n)) \tag{C.15}$$

C.1.5 Variance fractal dimension

The fractal dimension is a type of metric that allows to quantify or characterize the complexity of some pattern in terms of morphology, entropy, spectrum or variance (Carvalho et al., 2005; Phinyomark et al., 2014). In relation to the analysis of signals, this feature allows to emphasize the underlying complexity in the structure of a signal (Phinyomark et al., 2014).

The Variance fractal dimension (VFD) is a type of fractal dimension that has been used in previous studies such as (Carvalho et al., 2005; J. Gnitecki and Z. Moussavi, 2003) for heart sound detection. This expression is determined by the Hurst exponent, which is obtained from the power law relationship that exists between the variance of the signal amplitude increments $s(t)$ (denoted as $(\Delta s)_{\Delta t} = s(t_{i+1}) - s(t_i)$) over the increments in time ($\Delta t = t_{i+1} - t_i$). Mathematically this can be expressed as:

$$\mathrm{Var}((\Delta s)_{\Delta t} \sim \Delta t^{2H} \tag{C.16}$$

Where H corresponds to the Hurst exponent. From the relation (C.16), and considering a window $w(n)$ with length N as the input signal, the Hurst exponent is defined as (Phinyomark et al., 2014):

$$H = \lim_{\Delta t \to 0} \frac{1}{2} \cdot \frac{\log(\mathrm{Var}[(\Delta w)_{\Delta t})])}{\log(\Delta t)} \tag{C.17}$$

And:

$$\mathrm{Var}[(\Delta w)_{\Delta t}] = \frac{1}{N_k - 1} \left[\sum_{j=1}^{N_k} (\Delta w)_{jk}^2 - \frac{1}{N_k} \left(\sum_{j=1}^{N_k} (\Delta w)_{jk} \right)^2 \right] \tag{C.18}$$

With:

$$(\Delta w)_{jk} = w(jn_k) - w((j-1)n_k), \forall j = 1, ..., N_k \tag{C.19}$$

Where n_k corresponds to the size of the step between two instants, $N_k = \lfloor N/n_k \rfloor$ corresponds to the number of sub-windows generated by the step n_k from the window of length N, and $\lfloor \cdot \rfloor$ corresponds to the floor operator. The VFD is defined from the Hurst exponent as (Carvalho et al., 2005; J. Gnitecki and Z. Moussavi, 2003):

$$FD_\sigma = E + 1 - H \tag{C.20}$$

Where E corresponds to the inherent Euclidean dimension, which in the case of audio signals corresponds to $E = 1$. The VFD is calculated over a window of the signal using the following series of steps (Carvalho et al., 2005; J. Gnitecki and Z. Moussavi, 2003; Phinyomark et al., 2014):

1. First, the minimum (k_{min}) and maximum (k_{max}) step sizes are selected. Between those the time increment n_k will be defined. In addition, the scale in which each increment will be made is defined, which can be based on a unit scale ($n_k = k$), or on a dyadic scale ($n_k = 2^k$).

2. For each $k = \{k_{min}, ..., k_{max}\}$:

   i. The number $\lfloor N_k = N/n_k \rfloor$ of sub-windows generated by the temporary increments n_k is defined.

   ii. Using the expressions in (C.18) and (C.19) we obtain $\mathrm{Var}[(\Delta w)_{\Delta t}]$.

   iii. From the results obtained, we define $X_k = \log(n_k)$ and $Y_k = \log(\mathrm{Var}[(\Delta w)_{\Delta t}])$.

3. From the X_k and Y_k obtained for each $k = \{k_{min}, ..., k_{max}\}$, a linear regression based on least squares is calculated that allows adjusting a slope β for the log-log plot obtained for this set of points. From these, β is defined from the solution of the least squares slope as:

$$\beta = \frac{K \sum_{i=1}^{K} X_i Y_i - \sum_{i=1}^{K} X_i \sum_{i=1}^{K} Y_i}{K \sum_{i=1}^{K} X_i^2 - \left(K \sum_{i=1}^{K} X_i \right)^2} \tag{C.21}$$

4. Then, the Hurst exponent is defined as:

$$H = \frac{1}{2}\beta \tag{C.22}$$

5. Finally, the fractal dimension of the window $w(n)$ is defined as:

$$FD_\sigma(w(n)) = 2 - H \tag{C.23}$$

In order to obtain the time evolution of the VFD on a real discrete signal $s(n)$, the signal must be windowed using windows of length N with a step of N_{step} points. For each window defined from the signal, the procedure mentioned between points 2 and 5 must be repeated.

C.1.6 Frequency bands energy

The calculation of the frequency bands energy makes it possible to know how much energy exists in a specific frequency interval, for a given time interval. To perform this calculation, the spectrogram $S(m,k)$ is defined from the short-time Fourier transform (STFT) of a discrete signal $s(n)$, using an analysis window $w(t)$ of length N. The spectrogram $S(m,k)$ is defined as:

$$S(m,k) = \sum_{n=-\infty}^{\infty} s(n)w(n-m)e^{-j2\pi k\frac{n}{N}} \tag{C.24}$$

From this, it is possible to calculate the energy of certain specific frequency bands. Defining a band of interest for the frequency bins between $k \in [k_{low}, k_{high}]$, the energy envelope by bands $p(m)$ is defined as:

$$p(m) = \sum_{k=k_{low}}^{k_{high}} |S(m,k)|^2 \tag{C.25}$$

In the case of defining a frequency interval $f \in [f_{low}, f_{high}]$ in Hz for this calculation, you will have to choose the frequency bins closest to each one. This is because the representation of each frequency bin is given in discrete steps, and generally does not coincide with the limits of the specified frequency interval.

C.1.7 Spectral tracking

This type of technique consists of tracking the amplitude at a specific frequency over time. In (A. Iwata et al., 1980) a model based on spectral linear predictions is used that defines a discrete transfer function using only poles, which is defined in the form (Makhoul, 1975):

$$\hat{S}(z) = \frac{G}{1 + \sum_{k=1}^{p} a_k z^{-k}} \tag{C.26}$$

From which it is possible to characterize the power spectrum defined as:

$$\hat{P}(f) = |\hat{S}(z = e^{j2\pi f})|^2 = \frac{G^2}{|1 + \sum_{k=1}^{p} a_k e^{-j2\pi fk}|^2} \tag{C.27}$$

In (A. Iwata et al., 1980) this model is used with a number of $p = 8$ poles to track frequencies that characterize S_1 and S_2 from their dominant frequencies. To obtain the values of a_k an iterative algorithm is used which is detailed in (Akira Iwata et al., 1977), whose demonstration is available in (Makhoul, 1975).

However, it is also possible to implement this idea using the spectrogram of a signal. Indeed, let $S(m,k)$ the STFT of a signal $s(n)$ as defined in (C.24). It is

Table C.1: Study of the correlations between the binary signal of heart sound positions and the different levels of detail of a DWT with mother Wavelet *db6*. The best option is highlighted in green.

Discrete Wavelet Transform	
Parameters	**Mother Wavelet *db6***
Level 1 detail	0.2899 ± 0.1083
Level 2 detail	0.3305 ± 0.0968
Level 3 detail	0.4093 ± 0.0853
Level 4 detail	0.4671 ± 0.1034
Level 5 detail	0.4001 ± 0.1365
Level 6 detail	0.1331 ± 0.1178

Table C.2: Study of the correlations between the binary signal of heart sound positions and the product between different levels of detail of a discrete Wavelet transform with mother Wavelet db6. The best option is highlighted in green.

Parameters	**Multi-scale Wavelet Product**
Levels $\{1, 2, 3, 4\}$	0.1199 ± 0.0585
Levels $\{2, 3, 4\}$	0.1941 ± 0.0695
Levels $\{3, 4\}$	0.3226 ± 0.0796
Levels $\{2, 3\}$	0.2351 ± 0.0771
Levels $\{1, 2, 3\}$	0.1319 ± 0.0653
Levels $\{3, 4, 5\}$	0.2579 ± 0.0763
Levels $\{4, 5\}$	0.3635 ± 0.0965

possible to obtain the spectral tracking $\tilde{P}(f)$ of a specific frequency f_0 by evaluating:

$$\tilde{P}(f_0) = |S(m, k_{f_0})| \tag{C.28}$$

Where k_{f_0} corresponds to the frequency bin closest to f_0, as mentioned in section C.1.6.

C.2 Pearson correlation coefficient analysis

This section presents the results of a previous analysis for the selection of parameters that define each feature. To do this, Pearson's correlation coefficient is used to compare each feature with the binary signal that describes the heart sound position.

Below are the tables that indicate each of the alternatives studied for each feature.

Table C.3: Study of the correlations between the binary signal of heart sound positions and the energy for different frequency bands with different window length and step parameters. The best options for each set of parameters are highlighted in yellow, and the best option is highlighted in green.

Frequency bands energy			
Parameters	$f = [30, 180]$	$f = [30, 100]$	$f = [30, 60]$
$N = 32$, $step = 4$	0.6118 ± 0.108	0.6152 ± 0.11	0.6131 ± 0.1236
$N = 32$, $step = 8$	0.6099 ± 0.1077	0.6133 ± 0.1097	0.6114 ± 0.1233
$N = 32$, $step = 16$	0.6002 ± 0.1067	0.6034 ± 0.1085	0.6033 ± 0.1222
$N = 64$, $step = 8$	0.6772 ± 0.1181	0.6847 ± 0.1194	0.673 ± 0.1325
$N = 64$, $step = 16$	0.6737 ± 0.1175	0.6812 ± 0.1188	0.6696 ± 0.1318
$N = 64$, $step = 32$	0.6568 ± 0.1151	0.6641 ± 0.1164	0.653 ± 0.1294
$N = 128$, $step = 16$	0.6815 ± 0.1295	0.6928 ± 0.1291	0.6832 ± 0.1377
$N = 128$, $step = 32$	0.6761 ± 0.1281	0.6873 ± 0.1278	0.6779 ± 0.1363
$N = 128$, $step = 64$	0.6535 ± 0.1232	0.664 ± 0.1229	0.6541 ± 0.1317
$N = 256$, $step = 32$	0.4495 ± 0.1244	0.4623 ± 0.1226	0.4585 ± 0.1245
$N = 256$, $step = 64$	0.4474 ± 0.1228	0.4602 ± 0.1206	0.4565 ± 0.1221
$N = 256$, $step = 128$	0.4221 ± 0.1178	0.4349 ± 0.1144	0.4322 ± 0.1139
$N = 512$, $step = 64$	0.1032 ± 0.1421	0.106 ± 0.1374	0.1022 ± 0.1309
$N = 512$, $step = 128$	0.0949 ± 0.1421	0.0975 ± 0.1373	0.0935 ± 0.1314
$N = 512$, $step = 256$	0.0768 ± 0.1426	0.0814 ± 0.1377	0.0802 ± 0.1328
$N = 1024$, $step = 128$	0.0295 ± 0.1056	0.0303 ± 0.1052	0.0303 ± 0.1015
$N = 1024$, $step = 256$	0.0389 ± 0.1077	0.0404 ± 0.1072	0.0404 ± 0.1034
$N = 1024$, $step = 512$	0.0304 ± 0.0922	0.0306 ± 0.0922	0.0295 ± 0.0899

Table C.4: Study of the correlations between the binary signal of heart sound positions and the VFD for different parameters of window length and step. The best option is highlighted in green.

Parameters	**Variance Fractal Dimension**
$N = 32$, $step = 4$	0.6703 ± 0.1085
$N = 32$, $step = 8$	0.6679 ± 0.1085
$N = 32$, $step = 16$	0.6591 ± 0.1083
$N = 64$, $step = 8$	0.7213 ± 0.1101
$N = 64$, $step = 16$	0.718 ± 0.1096
$N = 64$, $step = 32$	0.7056 ± 0.1084
$N = 128$, $step = 16$	0.6628 ± 0.1065
$N = 128$, $step = 32$	0.659 ± 0.106
$N = 128$, $step = 64$	0.6408 ± 0.1029
$N = 256$, $step = 32$	0.4087 ± 0.1078
$N = 256$, $step = 64$	0.4047 ± 0.1061
$N = 256$, $step = 128$	0.3883 ± 0.1065
$N = 512$, $step = 64$	0.245 ± 0.1188
$N = 512$, $step = 128$	0.2411 ± 0.1183
$N = 512$, $step = 256$	0.2016 ± 0.1191
$N = 1024$, $step = 128$	0.0905 ± 0.1218
$N = 1024$, $step = 256$	0.0748 ± 0.1189
$N = 1024$, $step = 512$	0.0561 ± 0.102

Table C.5: Study of the correlations between the binary signal of heart sound positions and the spectral tracking for different frequencies with different window length and step parameters. The best options for each set of parameters are highlighted in yellow, and the best option is highlighted in green.

	Spectral tracking							
Parameters	$f = 30$	$f = 40$	$f = 50$	$f = 60$	$f = 70$	$f = 90$	$f = 100$	$f = 120$
$N = 32$, $step = 4$	0.6131 ± 0.1236	0.6131 ± 0.1236	0.5959 ± 0.1098	0.5959 ± 0.1098	0.5443 ± 0.1085	0.5443 ± 0.1085	0.5443 ± 0.1085	0.4788 ± 0.1302
$N = 32$, $step = 8$	0.6114 ± 0.1233	0.6114 ± 0.1233	0.5937 ± 0.1095	0.5937 ± 0.1095	0.5419 ± 0.1082	0.5419 ± 0.1082	0.5419 ± 0.1082	0.4765 ± 0.1297
$N = 32$, $step = 16$	0.6033 ± 0.1222	0.6033 ± 0.1222	0.5811 ± 0.1083	0.5811 ± 0.1083	0.5274 ± 0.1061	0.5274 ± 0.1061	0.5274 ± 0.1061	0.4611 ± 0.1273
$N = 64$, $step = 8$	0.6541 ± 0.143	0.6634 ± 0.1299	0.6634 ± 0.1299	0.6392 ± 0.1244	0.6124 ± 0.1281	0.578 ± 0.1373	0.578 ± 0.1373	0.5021 ± 0.1619
$N = 64$, $step = 16$	0.6511 ± 0.1421	0.6596 ± 0.1291	0.6596 ± 0.1291	0.6352 ± 0.1237	0.6084 ± 0.1274	0.5741 ± 0.1366	0.5741 ± 0.1366	0.4985 ± 0.1608
$N = 64$, $step = 32$	0.6372 ± 0.1401	0.6373 ± 0.126	0.6373 ± 0.126	0.6107 ± 0.1222	0.5861 ± 0.1244	0.5522 ± 0.1319	0.5522 ± 0.1319	0.4792 ± 0.1562
$N = 128$, $step = 16$	0.6211 ± 0.1644	0.6466 ± 0.1477	0.6396 ± 0.1445	0.6247 ± 0.1429	0.596 ± 0.1505	0.5567 ± 0.16	0.5403 ± 0.1653	0.5001 ± 0.1779
$N = 128$, $step = 32$	0.6164 ± 0.163	0.6412 ± 0.1463	0.6342 ± 0.1431	0.6193 ± 0.1415	0.5909 ± 0.1488	0.552 ± 0.1584	0.5358 ± 0.1635	0.4957 ± 0.1761
$N = 128$, $step = 64$	0.5957 ± 0.1588	0.6135 ± 0.1435	0.6025 ± 0.1419	0.5918 ± 0.137	0.5641 ± 0.1434	0.5288 ± 0.1549	0.512 ± 0.1593	0.4754 ± 0.1724
$N = 256$, $step = 32$	0.3941 ± 0.1418	0.4008 ± 0.1289	0.3815 ± 0.1341	0.3858 ± 0.1351	0.3598 ± 0.1346	0.3349 ± 0.1393	0.3191 ± 0.1429	0.2812 ± 0.1503
$N = 256$, $step = 64$	0.3929 ± 0.1398	0.3992 ± 0.1272	0.3797 ± 0.1322	0.3846 ± 0.134	0.3584 ± 0.1334	0.334 ± 0.1382	0.3182 ± 0.1424	0.2803 ± 0.15
$N = 256$, $step = 128$	0.3714 ± 0.1335	0.3757 ± 0.1198	0.3557 ± 0.125	0.3622 ± 0.128	0.336 ± 0.1296	0.3126 ± 0.1352	0.299 ± 0.1388	0.2635 ± 0.1436
$N = 512$, $step = 64$	0.0817 ± 0.1113	0.0811 ± 0.1126	0.0785 ± 0.1134	0.0838 ± 0.1162	0.0784 ± 0.1205	0.0742 ± 0.123	0.0719 ± 0.1225	0.0651 ± 0.1189
$N = 512$, $step = 128$	0.0746 ± 0.1106	0.0746 ± 0.1129	0.0723 ± 0.1122	0.0779 ± 0.1152	0.0732 ± 0.1201	0.0701 ± 0.1211	0.068 ± 0.121	0.0607 ± 0.1174
$N = 512$, $step = 256$	0.0658 ± 0.114	0.0647 ± 0.118	0.0607 ± 0.1181	0.0655 ± 0.1216	0.0602 ± 0.1253	0.0555 ± 0.1315	0.0549 ± 0.1293	0.0437 ± 0.1261
$N = 1024$, $step = 128$	0.0258 ± 0.0844	0.0212 ± 0.0853	0.0252 ± 0.0819	0.0243 ± 0.0813	0.0203 ± 0.0801	0.021 ± 0.0796	0.0225 ± 0.0783	0.0176 ± 0.073
$N = 1024$, $step = 256$	0.0312 ± 0.0906	0.0281 ± 0.0876	0.0311 ± 0.0866	0.0274 ± 0.0888	0.0249 ± 0.0871	0.0266 ± 0.0858	0.0252 ± 0.0861	0.0192 ± 0.0825
$N = 1024$, $step = 512$	0.0219 ± 0.0813	0.0214 ± 0.0802	0.0254 ± 0.0776	0.0229 ± 0.0802	0.019 ± 0.0811	0.0208 ± 0.0808	0.019 ± 0.0764	0.0158 ± 0.0771

Table C.6: Table of descriptors ordered by correlation (Pearson's coefficient) with the heart sound positions.

Pearson correlation		
Feature	μ	σ
Variance Fractal Dimension	0.7213	0.1101
Modified Hilbert Envelope	0.7034	0.1324
Frequency band energy	0.6909	0.1274
Spectral tracking $f = 40$ Hz	0.6634	0.1299
Homomorphic filter	0.6550	0.1114
Spectral tracking $f = 60$ Hz	0.6392	0.1244
Classic Hilbert Envelope	0.5168	0.1024
DWT level 4 detail con *db6*	0.4720	0.1068
Multi-scale Wavelet Product	0.3635	0.0965

Appendix D

NMF theory: more about

D.1 Results interpretation

Once the NMF decomposition process is finished, the matrices $\mathbf{W}$ and $\mathbf{H}$ are obtained. On the one hand, $\mathbf{W} = [w_1, w_2, \ldots, w_K]$ can be understood as a dictionary of recurrent patterns or bases, where each column $\mathbf{W} = [w_1, w_2, \ldots, w_K]$ represents a pattern or base. On the other hand, $\mathbf{H} = [h_1, h_2, \ldots, h_K]^T$ can be understood as a matrix containing the activation or encoding coefficients of the bases $\mathbf{W}$, where each row $h_i \in \mathbb{R}^{1\times M}$ represents the weight that the base w_i has for each instant. If a coefficient h_{ij} is small then the base w_i will have little presence at moment j. On the contrary, if h_{ij} is large then the base w_i will have a great presence at that moment (Févotte, Vincent, et al., 2018; Ganesh R. Naik, 2016).

This can be graphically represented in the figure D.1 (obtained from (Févotte, Vincent, et al., 2018)). On the left is the original matrix and on the right the matrices $\mathbf{W}$ and $\mathbf{H}$. Note that in the original matrix there are 3 components: red, green and yellow, where yellow corresponds to the addition between red and green. Based on this idea, it is possible to apply NMF considering $k = 2$ base components (those representing red and green). As can be seen, the first column of $\mathbf{W}$ corresponds to the pattern of the red color, while the second column corresponds to the pattern of the green color. Meanwhile, the first row of $\mathbf{H}$ indicates the activation moments (1 in black; 0 in white) of the red pattern, while the second row to the activation moments of the green pattern. A simple inspection will show that $\mathbf{WH}$ returns the original matrix.

In case the matrix $\mathbf{X} \in \mathbb{R}^{N\times M}$ is the magnitude of a spectrogram, each column of $\mathbf{W} \in \mathbb{R}^{N\times K}$ will correspond to the magnitude of the characteristic frequency spectrum of each component, while each row of $\mathbf{H} \in \mathbb{R}^{K\times M}$ will indicate the weighting at the time of each spectrum defined in the matrix $\mathbf{W}$. Mathematically, each matrix

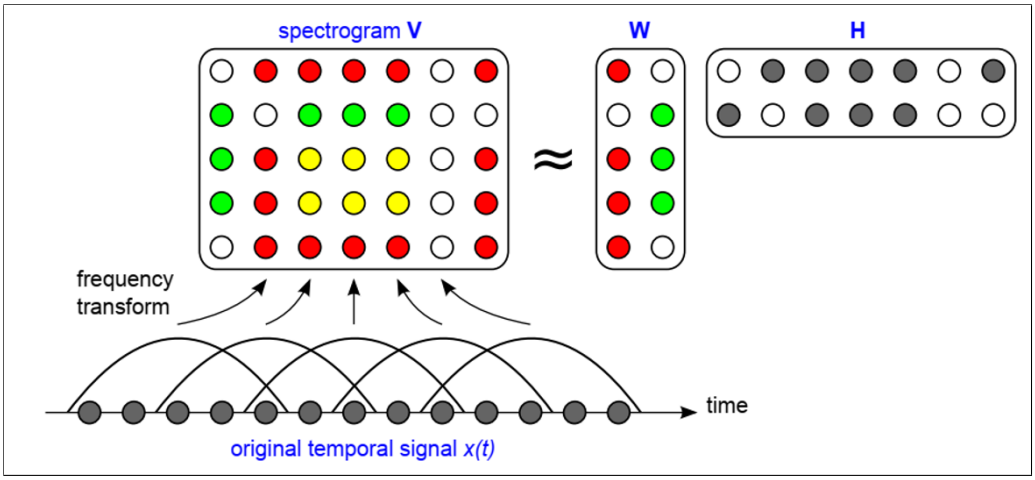

Figure D.1: Illustrative NMF decomposition example. In this image it is possible to see 2 components: red and green. Yellow corresponds to the sum of red and green. In each column of **W** there is a characteristic pattern, while in each row of **H** there is the activation of the patterns in time. Extracted from (Févotte, Vincent, et al., 2018).

can be expressed as:

$$\begin{aligned}\mathbf{W} &= \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1K} \\ w_{21} & w_{22} & \cdots & w_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ w_{N1} & w_{N2} & \cdots & w_{NK} \end{bmatrix} \\ &= \begin{bmatrix} | & | & & | \\ w_1 & w_2 & \cdots & w_K \\ | & | & & | \end{bmatrix}\end{aligned} \tag{D.1}$$

$$\begin{aligned}\mathbf{H} &= \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1M} \\ h_{21} & h_{22} & \cdots & h_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ h_{K1} & h_{K2} & \cdots & h_{KM} \end{bmatrix} \\ &= \begin{bmatrix} - & h_1^T & - \\ - & h_2^T & - \\ & \vdots & \\ - & h_K^T & - \end{bmatrix}\end{aligned} \tag{D.2}$$

From this representation, it is a bit easier to visualize that the column w_i will always be multiplied by the row h_i^T. Indeed, we can express the multiplication **WH**

as:

$$\mathbf{X} \approx \mathbf{WH} = w_1 h_1^T + w_2 h_2^T + \cdots + w_K h_K^T \tag{D.3}$$

Then, the definition of a component $\mathbf{X}_i \in \mathbb{R}^{N \times M}$ from the NMF decomposition of a spectrogram $\mathbf{X} \in \mathbb{R}^{N \times M}$ is given by:

$$\mathbf{X}_i = w_i h_i^T \tag{D.4}$$

Where:

$$\mathbf{WH} = \sum_{i=1}^{K} \mathbf{X}_i \tag{D.5}$$

D.2 Binary Mask

Another used mask in source separation with NMF is the binary mask (Lin and Hasting, 2013; Shah, Koch, et al., 2015; Shah and C. B. Papadias, 2013). Unlike the Wiener filter, the binary mask does not perform a partial weighting on each entry in the matrix, but instead assigns each entry to a single component. The binary mask is defined as:

$$\mathbf{M}_i = \begin{cases} 1 & \forall \mathbf{X}_i > \mathbf{X}_j, j \in \{1, 2, \ldots, k\}, j \neq i \\ 0 & \text{otherwise} \end{cases} \tag{D.6}$$

From (D.6) it can be deduced that, for an input (n, m) of the components, the mask of the component i will have a 1 if and only if the entry (n, m) of $\mathbf{X}_i$ is greater than all the (n, m) entries of the rest of the components. That is, if $\mathbf{X}_i(n, m)$ is the maximum of the set $\{X_k(n, m), \forall k = 1, 2, \ldots, K\}$. Otherwise, $\mathbf{X}_i(n, m)$ will be 0. As a result of this, it is possible to conclude that the entry (n, m) of the original $\mathbf{X}$ matrix will be assigned to a single component.

An example of the binary mask can be seen in figure D.2, where a decomposition is done for $K = 3$ components. In the case of the binary mask, the input (n, m) is non-zero in only one component, while in the rest it is equal to zero. Therefore, the components obtained by binary mask present more discontinuities than in the Wiener filter presented.

D.3 NMF properties

Source separation by NMF has characteristics that make it a preferable alternative to other types of Blind Source Separation techniques. However, it also presents certain difficulties that pose challenges to consider. Next, properties that are considered relevant in the development of this approach are mentioned.

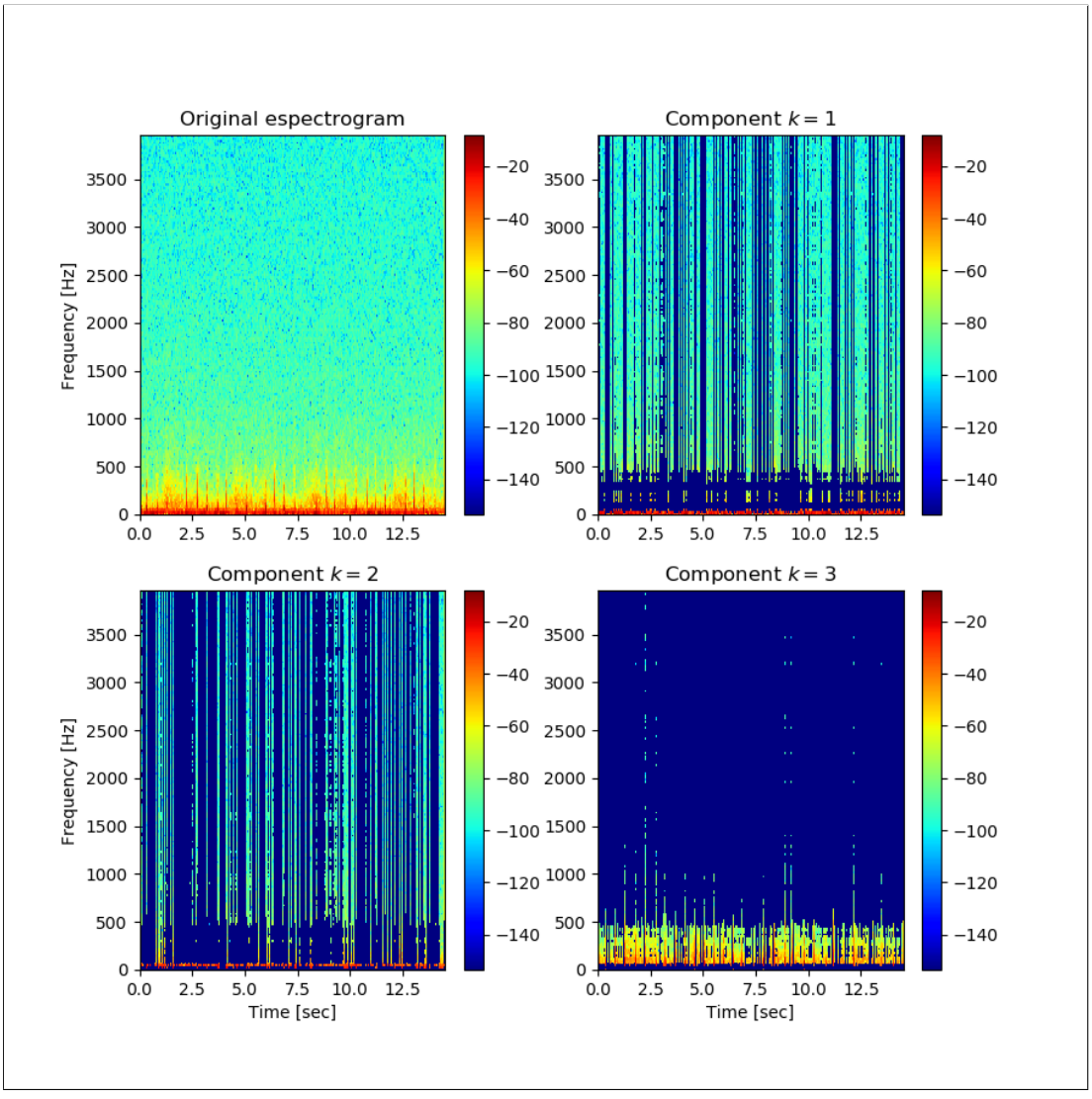

Figure D.2: Results of the NMF decomposition (in dB) with binary mask for $K = 3$ components. As can be seen, a single component has the entry (n, m) of the original signal, while the rest of the components have a 0 at that entry. The sum of the 3 components results in the original spectrogram.

D.3.1 Intuitively interpretable results

One of the main advantages that NMF has over other similar methods (such as PCA (Ganesh R. Naik, 2016; Z. Y. Zhang, 2012)) is the ease of interpreting its results. In section D.1 the logic that allows us to understand the results as the temporal weighting of different recognized bases in matrix $\mathbf{W}$ is developed. However, this advantage goes even further. The results can be understood as components that are additively combined whose sum approximates the matrix to be decomposed, which allows us to think of the result as a part based representation (Févotte, Vincent, et al., 2018; Ganesh R. Naik, 2016). Therefore, because subtractive combinations are not allowed, each component is a constituent part of the original result (Févotte, Vincent, et al., 2018). This property is desirable because there are certain applications where the intrinsic nature of the data is non-negative (for example, images).

D.3.2 Bounded solution

Among the characteristics that NMF presents is that the decomposition solutions are bounded. Geometrically, the base vectors generate a cone that contains the original data in the positive orthant (Ganesh R. Naik, 2016). In (Z. Zhang et al., 2007) this property is demonstrated which indicates that for some diagonal matrix $\mathbf{D} \geq 0$ such that if $\mathbf{W}^*$ and $\mathbf{H}^*$ are solutions of the problem posed in (3.2) then the solutions $\mathbf{W}^*\mathbf{D}$ and $\mathbf{H}^*\mathbf{D}^{-1}$ also will be bounded (Z. Y. Zhang, 2012). Intuitively this makes sense since if we consider what is mentioned in the D.3.1 section, each component can be understood as a constitutive part of a whole, which in this case would be the matrix $\mathbf{X}$. Therefore, it is expected that if the matrix $\mathbf{X}$ is bounded by some constant L, that is, $0 \leq \mathbf{X} \leq L$, then any pair of matrices $\mathbf{W}$ and $\mathbf{H}$ are too.

D.3.3 Number of components to decompose K

To define the number K of components to decompose, certain considerations must be taken. According to authors such as (Z. Y. Zhang, 2012) and (Shah, Koch, et al., 2015), $K \ll \min\{N, M\}$ should be used, where N corresponds to the row dimension and M to the column dimension from $\mathbf{X}$. However, other authors such as (Canadas-Quesada et al., 2017), (Lin and Hasting, 2013), (Févotte and Idier, 2011) and (Ganesh R. Naik, 2016) specify this relationship a little more, indicating that must satisfy $(N+M)K \leq NM$. However, as a basic idea, it is important to consider that the number of components to choose from is not close to the dimensions of the matrix $\mathbf{X}$.

On the other hand, it is important to consider that the larger K, the multiplication $\mathbf{WH}$ will be more similar to $\mathbf{X}$, therefore the error $\mathbf{E}$ will be less (Essid and Ozerov, 2014). This can be seen in the result of figure D.3, where the decomposition for different values of K is shown. Note that as K is larger, the multiplication $\mathbf{WH}$

naturally (without applying masks) becomes similar to $\mathbf{X}$.

D.3.4 Non-unique solution

One difficulty with NMF is that the solution to the problem presented in (3.2) is not unique. Indeed, considering an ideal case, let the matrices $\mathbf{W}^*$ and $\mathbf{H}^*$ be solutions of the problem (3.2) such that $\mathbf{X} = \mathbf{W}^*\mathbf{H}^*$, and let $\mathbf{Q}$ be an invertible matrix. Then:

$$\mathbf{X} = \mathbf{W}^*\mathbf{H}^* = \mathbf{W}^*\mathbf{Q}^{-1}\mathbf{Q}\mathbf{H}^* \tag{D.7}$$

From this it is possible to define a solution of the form $\mathbf{X} = \tilde{\mathbf{W}}^*\tilde{\mathbf{H}}^*$ where $\tilde{\mathbf{W}}^* = \mathbf{W}^*\mathbf{Q}^{-1}$ and $\tilde{\mathbf{H}}^* = \mathbf{Q}\mathbf{H}^*$, which are still valid solutions to the problem (Essid and Ozerov, 2014; Ganesh R. Naik, 2016). Also, different initial values of the algorithm (for $\mathbf{W}$ and $\mathbf{H}$) generally result in different local minima (M. N. Schmidt, 2008). Some authors such as (M. N. Schmidt, 2008) and (Ganesh R. Naik, 2016) recommend perform more executions with different starting points. In (Canadas-Quesada et al., 2017) multiple runs of the algorithm are used to average the solutions in order to obtain a representative sample.

D.3.5 Cost function selection

The selection of the cost function is influential when obtaining the solutions of the problem (3.2). There is a great variety of families of divergence functions defined in the literature which can be used as a cost function in NMF problem (Z. Y. Zhang, 2012). One of them is the β-divergence family, which presents three notable cases: the Itakura-Saito divergence ($\beta = 0$), the Kullback-Leibler divergence ($\beta = 1$) and the quadratic divergence or Euclidean distance ($\beta = 2$) (Févotte, Vincent, et al., 2018). However, depending on the type of application to perform, there will be more appropriate divergences than others. An interesting property of the β-divergence mentioned above is that it maintains the following relationship with respect to scale (Févotte, Vincent, et al., 2018):

$$d_\beta(\lambda x \mid \lambda y) = \lambda^\beta d_\beta(x \mid y) \tag{D.8}$$

From this expression it can be inferred that for decompositions with $\beta > 0$, large values will be considered more and a low precision will be expected in the estimation of small values. Small β values are useful in factorizations that have exponential decay across their frequency spectrum and that also exhibit low-energy transients. While for $\beta < 0$ small values will be considered more, having a low precision in estimating high values. Finally, for the case in which $\beta = 0$ it is said that the divergence is invariant to the scale since it is true that $d_\beta(\lambda x \mid \lambda y) = d_\beta(x \mid y)$ (Févotte, Vincent, et al., 2018).

Another property to consider is the convergence. The Euclidean distance $d_Q(x \mid y)$ is one of the most used cost functions due to its simplicity and that satisfy with

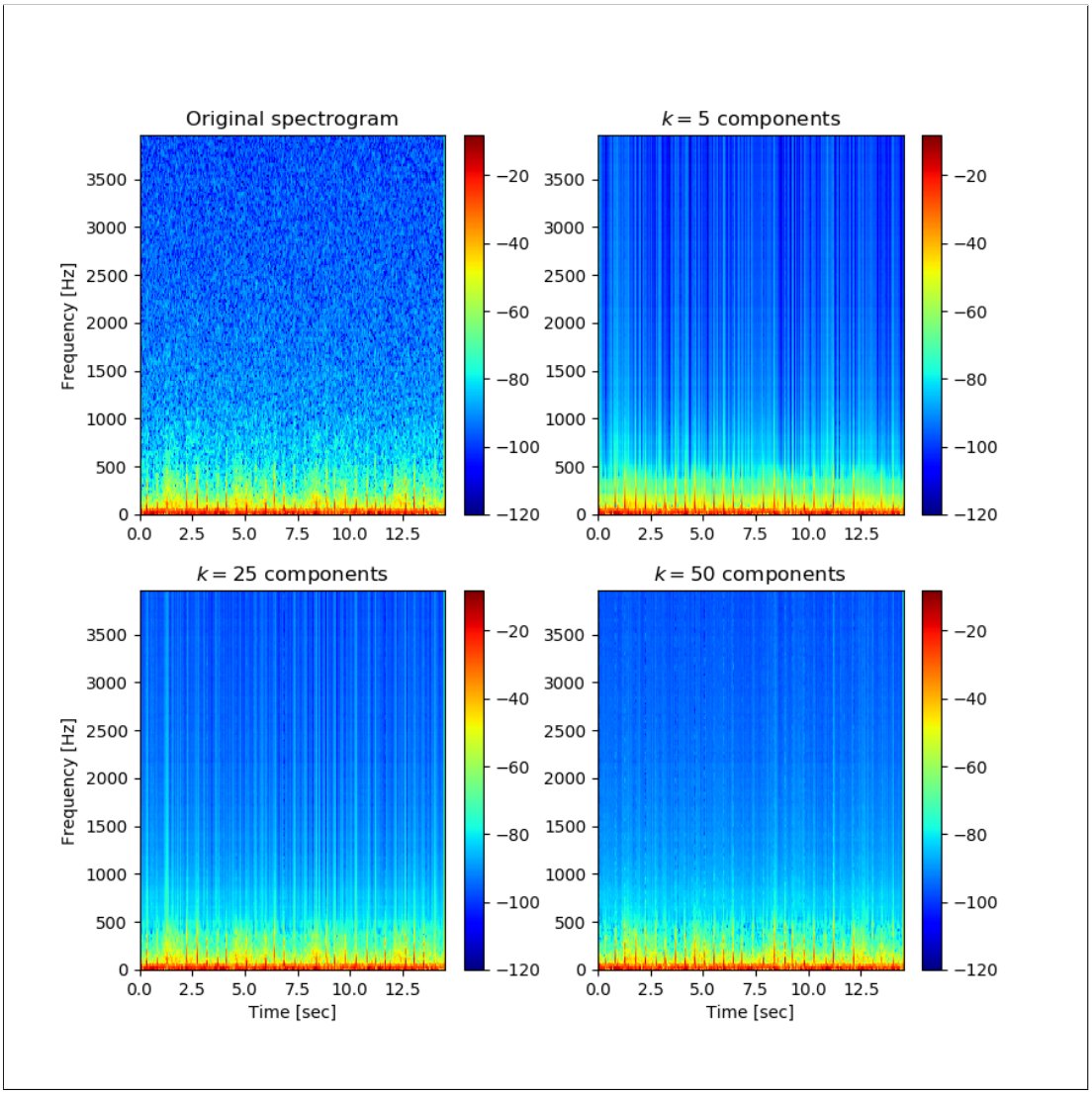

Figure D.3: Effect of K on NMF. The spectrogram of the original signal and NMF decomposition for $K = \{5, 25, 50\}$ components are presented. Notice that as the value of K increases the multiplication $\mathbf{WH}$ becomes more and more similar to the original spectrogram.

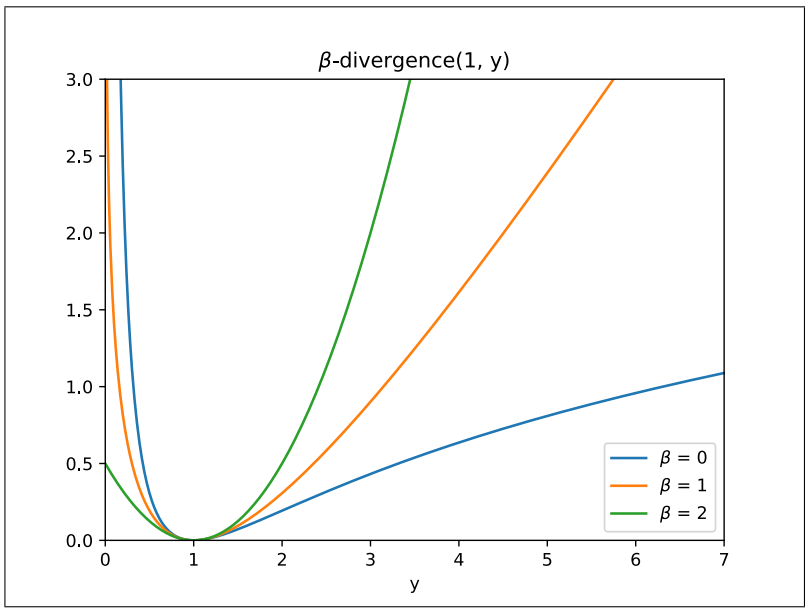

Figure D.4: β-divergence plots for different values of β, setting $x = 1$.

being a convex and differentiable function. In (Lee and Seung, 2001) it is also shown that through the multiplicative update method (MU) the Kullback-Leibler divergence reduces the value of its objective function at each step. However, this does not ensure that it converges to a local minimum or a stationary point (M. N. Schmidt, 2008). Finally, in the case of the Itakura-Saito divergence, problems may arise because it is not convex in both variables, as can be seen in figure D.4 (Essid and Ozerov, 2014).

D.3.6 β-divergence and from statistics

Depending on the chosen function β-divergence, it is possible to show that the resolution of the NMF factorization corresponds to the calculation of the maximum likelihood estimators under certain assumptions of the data (Févotte, Bertin, et al., 2009; Févotte, Vincent, et al., 2018; M. N. Schmidt, 2008).

For the case of the Euclidean distance, the factorization problem posed in (3.2) corresponds to the maximum likelihood estimator of $\mathbf{W}$ and $\mathbf{H}$ assuming a Gaussian additive noise model and that it is independently and identically distributed (i.i.d.). For the Kullback-Leibler divergence, it corresponds to the calculation of the maximum likelihood estimator assuming that the estimation noise distributes Poisson and i.i.d. Finally, for the case of the Itakura-Saito divergence, it corresponds to the calculation of the maximum likelihood estimate assuming that the noise of the estimate distributes exponential i.i.d. (Févotte, Bertin, et al., 2009; Févotte, Vincent, et al., 2018; M. N. Schmidt, 2008).

D.4 Solution algorithms

To solve this problem there are different algorithms, which adjust to the requirements of the problem and the chosen divergence function (Z. Y. Zhang, 2012). Some typical solving algorithms for this optimization problem are presented below.

D.4.1 Multiplicative update (MU) algorithm

Initially presented in (Lee and Seung, 2001), it is an algorithm based on gradient descent that has become quite popular due to its effectiveness and ease of implementation (Z. Y. Zhang, 2012). Because the overall optimization problem is not jointly convex at $\mathbf{W}$ y $\mathbf{H}$, updating stationary points together might not be minima (global or local). Based on this, it was decided to restrict the study to only one variable. For example, we will seek to update the value of $\mathbf{H}$ given $\mathbf{W}$ resulting in the following optimization problem (Févotte and Idier, 2011; Févotte, Vincent, et al., 2018):

$$\begin{aligned} \mathbf{H} &= \min_{\mathbf{H}} C(\mathbf{H}) \\ &\text{s.t. } \mathbf{H} \geq 0 \end{aligned} \tag{D.9}$$

Where $C(\mathbf{H}) \triangleq D(\mathbf{X} \mid \mathbf{WH})$ and $\mathbf{W}$ is fixed. From this expression an auxiliary function is defined that acts as the upper bound of $C(\mathbf{H})$, and that passes through the current point $\hat{\mathbf{H}}$. This upper bound is constructed from the decomposition of $C(\mathbf{H})$ into the sum of its convex and concave part (and some constant). Then, it is possible to construct an upper bound using the Jensen inequality (convexity definition) for the case of the convex part (Févotte and Idier, 2011; Févotte, Vincent, et al., 2018). While for the concave part it is used that the tangent at any point of the curve constitutes an upper bound (Févotte and Idier, 2011). From these two ideas the upper bound is built, in which its minimum point is sought. Finally this minimum point is projected onto the matrix $\mathbf{H}$, constituting the next point in the algorithm. It should be noted that this same procedure can be performed for $\mathbf{W}$ leaving $\mathbf{H}$ fixed.

In short, for the particular case of β-divergences the algorithm in each iteration must be updated using (Févotte, Bertin, et al., 2009; Févotte and Idier, 2011; Févotte, Vincent, et al., 2018):

$$\mathbf{H} \leftarrow \mathbf{H} \odot \frac{\mathbf{W}^T \left((\mathbf{WH})^{\odot[\beta-2]} \odot \mathbf{X} \right)}{\mathbf{W}^T (\mathbf{WH})^{\odot[\beta-1]}} \tag{D.10}$$

$$\mathbf{W} \leftarrow \mathbf{W} \odot \frac{\left((\mathbf{WH})^{\odot[\beta-2]} \odot \mathbf{X} \right) \mathbf{H}^T}{(\mathbf{WH})^{\odot[\beta-1]} \mathbf{H}^T} \tag{D.11}$$

Where the operator $\odot$ corresponds to the element-wise product (Hadamard Product) or exponential-wise (in case of superscript) of the arrays, and the division is also element-wise (Févotte, Vincent, et al., 2018).

Given the way this method is developed, it is possible to ensure the non-negativity of $\mathbf{W}$ and $\mathbf{H}$ (Essid and Ozerov, 2014). This is achieved since the factor by which each matrix must be multiplied is always non-negative. Furthermore, in the case of the β-divergences, their monotonicity is demonstrated. However, compared to other algorithms, their convergence may not be the fastest (Essid and Ozerov, 2014).

D.4.2 Gradient descent algorithm

It corresponds to a simple optimization strategy that advances by iteratively searching for the local minimum based on the information provided by the gradient of the function at the point where it is located. In general, this method is designed for optimization problems with limited constraints and with a least squares cost function (Z. Y. Zhang, 2012). A point x of the cost function $f(x)$ is updated through: (M. N. Schmidt, 2008):

$$x \leftarrow x - \lambda \nabla f(x) \tag{D.12}$$

Where λ corresponds to the step size and $\nabla f(x)$ to the gradient of the function. In the context of the NMF problem, this algorithm can be used for a quadratic cost function, from which the updates for $\mathbf{W}$ and $\mathbf{H}$ are defined as (Z. Y. Zhang, 2012):

$$\mathbf{W}_n \leftarrow F\big(\mathbf{W}_{n-1} - \lambda \nabla_{\mathbf{W}} D(\mathbf{X} \mid \mathbf{W}_{n-1}\mathbf{H}_{n-1})\big) \tag{D.13}$$

$$\mathbf{H}_n \leftarrow F\big(\mathbf{H}_{n-1} - \lambda \nabla_{\mathbf{H}} D(\mathbf{X} \mid \mathbf{W}_{n}\mathbf{H}_{n-1})\big) \tag{D.14}$$

Where the subscript n corresponds to the n-th iteration of the algorithm and F is a function defined as (M. N. Schmidt, 2008; Z. Y. Zhang, 2012):

$$F(x) = \begin{cases} L, & x \geq L \\ x, & x > 0 \\ 0, & \text{otherwise} \end{cases} \tag{D.15}$$

One of the advantages of this approach compared to multiplicative update algorithm is that it can converge faster (Essid and Ozerov, 2014). However, this method can have problems with divergences that are not well defined throughout its domain (for example, the Kullback-Leibler that has a logarithm and therefore is undefined at 0) (Z. Y. Zhang, 2012).

D.4.3 Newton descent algorithm

It is a method similar to the gradient algorithm, but use second-order information from the cost function. Indeed, updating a point x of a cost function $f(x)$ is done through Schmidt2008Single-channelFactorization, Nocedal2006NumericalEdition:

$$x \leftarrow x - \big[\nabla^2 f(x)\big]^{-1} \nabla f(x) \tag{D.16}$$

Where $\nabla^2 f(x)$ corresponds to the Hessian of the cost function, which is defined as a matrix of the form:

$$\nabla^2 f(x) = \left[\frac{\partial^2 f}{\partial x_i \partial x_j} \right] \tag{D.17}$$

In terms of convergence it is faster than the gradient method, since it converges quadratically towards the optimum. Instead the gradient method converges linearly (Nocedal and Wright, 2006). However, Newton algorithm can be computationally expensive because it is necessary to find the inverse of the Hessian at each iteration. Therefore, in situations where the geometry of the problem does not fit properly, it may even take longer to reach the optimum.

An alternative to solve this problem is the quasi-Newton method, where a estimate B_k of the Hessian matrix is made from the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm (Nocedal and Wright, 2006; M. N. Schmidt, 2008). This reduces the computational complexity of the problem, going from an order of $O(n^3)$ operations to an order of $O(n^2)$ operations.

REFERENCES

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D. G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., ... Zheng, X. (2016). TensorFlow: A system for large-scale machine learning. *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation, OSDI 2016*.

Aggarwal, C. C. (2018a). Teaching Deep Learners to Generalize. *Neural networks and deep learning*. Springer International Publishing. https://doi.org/10.1007/978-3-319-94463-0{_}4

Aggarwal, C. C. (2018b). Training Deep Neural Networks. *Neural networks and deep learning*. Springer International Publishing. https://doi.org/10.1007/978-3-319-94463-0{_}3

Ahlstrom, C., Liljefeldt, O., Hult, P., & Ask, P. (2005). Heart sound cancellation from lung sound recordings using recurrence time statistics and nonlinear prediction. *IEEE Signal Processing Letters*. https://doi.org/10.1109/LSP.2005.859528

Ali, M. N., El-Dahshan, E. S. A., & Yahia, A. H. (2017). Denoising of Heart Sound Signals Using Discrete Wavelet Transform. *Circuits, Systems, and Signal Processing*. https://doi.org/10.1007/s00034-017-0524-7

Al-Naggar, N. Q. [Noman Q.], & Al-Udyni, M. H. (2018). Performance of Adaptive Noise Cancellation with Normalized Last-Mean-Square Based on the Signal-to-Noise Ratio of Lung and Heart Sound Separation. *Journal of Healthcare Engineering, 2018*. https://doi.org/10.1155/2018/9732762

Al-Naggar, N. Q. [Noman Qaid]. (2013). A new method of lung sounds filtering using modulated least mean square—Adaptive noise cancellation. *Journal of Biomedical Science and Engineering*, *06*(09). https://doi.org/10.4236/jbise.2013.69106

Ari, S., Kumar, P., & Saha, G. (2008). A robust heart sound segmentation algorithm for commonly occurring heart valve diseases. *Journal of Medical Engineering and Technology*, *32*(6). https://doi.org/10.1080/03091900601015162

ARI, S., & SAHA, G. (2007). ON A ROBUST ALGORITHM FOR HEART SOUND SEGMENTATION. *Journal of Mechanics in Medicine and Biology*, *07*(02). https://doi.org/10.1142/s0219519407002200

Ayari, F., Ksouri, M., & Alouani, A. T. (2012). Lung sound extraction from mixed lung and heart sounds FASTICA algorithm. *Proceedings of the Mediterranean Electrotechnical Conference - MELECON*. https://doi.org/10.1109/MELCON.2012.6196444

Badrinarayanan, V., Kendall, A., & Cipolla, R. (2017). SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *39*(12). https://doi.org/10.1109/TPAMI.2016.2644615

Bahoura, M. (2009). Pattern recognition methods applied to respiratory sounds classification into normal and wheeze classes. *Computers in Biology and Medicine*, *39*(9). https://doi.org/10.1016/j.compbiomed.2009.06.011

Banerjee, S., Mishra, M., & Mukherjee, A. (2016). Segmentation and detection of first and second heart sounds (Si and S2) using variational mode decomposition. *IECBES 2016 - IEEE-EMBS Conference on Biomedical Engineering and Sciences*. https://doi.org/10.1109/IECBES.2016.7843513

Bao, P., & Zhang, L. (2003). Noise Reduction for Magnetic Resonance Images via Adaptive Multiscale Products Thresholding. *IEEE Transactions on Medical Imaging*, *22*(9). https://doi.org/10.1109/TMI.2003.816958

Bardou, D., Zhang, K., & Ahmad, S. M. (2018). Lung sounds classification using convolutional neural networks. *Artificial Intelligence in Medicine*, *88*. https://doi.org/10.1016/j.artmed.2018.04.008

Bentley, P., Nordehn, G., Coimbra, M., & Mannor, S. (n.d.). The PASCAL Classifying Heart Sounds Challenge 2011 (CHSC2011) Results. http://www.peterjbentley.com/heartchallenge/index.html

Boutana, D., Benidir, M., & Barkat, B. (2011). Segmentation and identification of some pathological phonocardiogram signals using time-frequency analysis. *IET Signal Processing*, *5*(6). https://doi.org/10.1049/iet-spr.2010.0013

Bryan, N., & Sun, D. (2013). Source Separation Tutorial Mini-Series II: Introduction to Non-Negative Matrix Factorization.

Canadas-Quesada, F. J., Ruiz-Reyes, N., Carabias-Orti, J., Vera-Candeas, P., & Fuertes-Garcia, J. (2017). A non-negative matrix factorization approach based on spectro-temporal clustering to extract heart sounds. *Applied Acoustics*. https://doi.org/10.1016/j.apacoust.2017.04.005

Carvalho, P., Gil, P., Henriques, J., Eugénio, L., & Antunes, M. (2005). Low complexity algorithm for heart sound segmentation using the variance fractal dimension. *2005 IEEE International Workshop on Intelligent Signal Processing - Proceedings*. https://doi.org/10.1109/wisp.2005.1531657

Castro, A., Vinhoza, T. T., Mattos, S. S., & Coimbra, M. T. (2013). Heart sound segmentation of pediatric auscultations using wavelet analysis. *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*. https://doi.org/10.1109/EMBC.2013.6610399

Charleston, S., & Azimi-Sadjadi, M. R. (1996). Reduced order Kalman filtering for the enhancement of respiratory sounds. *IEEE Transactions on Biomedical Engineering*, *43*(4). https://doi.org/10.1109/10.486262

Chen, T. E., Yang, S. I., Ho, L. T., Tsai, K. H., Chen, Y. H., Chang, Y. F., Lai, Y. H., Wang, S. S., Tsao, Y., & Wu, C. C. (2017). S1 and S2 heart sound recognition using deep neural networks. *IEEE Transactions on Biomedical Engineering*, *64*(2). https://doi.org/10.1109/TBME.2016.2559800

Chien, J., Huang, M. C., Lin, Y. D., & Chong, F. C. (2006). A study of heart sound and lung sound separation by independent component analysis technique. *Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings*. https://doi.org/10.1109/IEMBS.2006.260223

Choi, S., & Jiang, Z. (2008). Comparison of envelope extraction algorithms for cardiac sound signal segmentation. *Expert Systems with Applications*, *34*(2). https://doi.org/10.1016/j.eswa.2006.12.015

Couch, L. W. (1983). *Digital and Analog Communication Systems*. Macmillan.

Demir, F., Sengur, A., & Bajaj, V. (2020). Convolutional neural networks based efficient approach for classification of lung diseases. *Health Information Science and Systems*, *8*(1). https://doi.org/10.1007/s13755-019-0091-3

Dokur, Z. (2009). Respiratory sound classification by using an incremental supervised neural network. *Pattern Analysis and Applications*, *12*(4). https://doi.org/10.1007/s10044-008-0125-y

El-Segaier, M., Lilja, O., Lukkarinen, S., Sörnmo, L., Sepponen, R., & Pesonen, E. (2005). Computer-based detection and analysis of heart sound and murmur. *Annals of Biomedical Engineering*, *33*(7). https://doi.org/10.1007/s10439-005-4053-3

Essid, S., & Ozerov, A. (2014). A tutorial on nonnegative matrix factorisation with applications to audiovisual content analysis.

Feldman, M. (2008). Hilbert Transform, Envelope, Instantaneous Phase, and Frequency. *Encyclopedia of structural health monitoring*. https://doi.org/10.1002/9780470061626.shm046

Fernando, T., Ghaemmaghami, H., Denman, S., Sridharan, S., Hussain, N., & Fookes, C. (2020). Heart Sound Segmentation Using Bidirectional LSTMs with Attention. *IEEE Journal of Biomedical and Health Informatics*, *24*(6). https://doi.org/10.1109/JBHI.2019.2949516

Févotte, C., Bertin, N., & Durrieu, J. L. (2009). Nonnegative matrix factorization with the Itakura-Saito divergence: With application to music analysis. https://doi.org/10.1162/neco.2008.04-08-771

Févotte, C., & Idier, J. (2011). Algorithms for nonnegative matrix factorization with the β-divergence. https://doi.org/10.1162/NECO{_}a{_}00168

Févotte, C., Vincent, E., & Ozerov, A. (2018). Single-channel audio source separation with NMF: Divergences, constraints and algorithms. *Signals and communication technology*. https://doi.org/10.1007/978-3-319-73031-8{_}1

Flores-Tapia, D., Moussavi, Z. M., & Thomas, G. (2007). Heart sound cancellation based on multiscale products and linear prediction. *IEEE Transactions on Biomedical Engineering*. https://doi.org/10.1109/TBME.2006.886935

Gamero, L. G., & Watrous, R. (2003). Detection of the First and Second Heart Sound Using Probabilistic Models. *Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings*, *3*. https://doi.org/10.1109/iembs.2003.1280519

Ganesh R. Naik (Ed.). (2016). *Non-negative Matrix Factorization Techniques. Advances in Theory and Applications*. Springer-Verlag Berlin Heidelberg. https://doi.org/10.1007/978-3-662-48331-2

Gavriely, N. [N.], Palti, Y., & Gideon, A. (1981). Spectral characteristics of normal breath sounds. *Journal of Applied Physiology Respiratory Environmental and Exercise Physiology*. https://doi.org/10.1152/jappl.1981.50.2.307

Gavriely, N. [Noam], Nissan, M., Rubin, A. H. E., & Cugell, D. W. (1995). Spectral characteristics of chest wall breath sounds in normal subjects. *Thorax*. https://doi.org/10.1136/thx.50.12.1292

Gavrovska, A., Bogdanović, V., Reljin, I., & Reljin, B. (2014). Automatic heart sound detection in pediatric patients without electrocardiogram reference via pseudo-affine Wigner-Ville distribution and Haar wavelet lifting. *Computer Methods and Programs in Biomedicine*, *113*(2). https://doi.org/10.1016/j.cmpb.2013.11.018

Ghaderi, F., Mohseni, H. R., & Sanei, S. (2011). Localizing heart sounds in respiratory signals using singular spectrum analysis. *IEEE Transactions on Biomedical Engineering*, *58*(12 PART 1). https://doi.org/10.1109/TBME.2011.2162728

Gharehbaghi, A., Dutoit, T., Sepehri, A., Hult, P., & Ask, P. (2011). An automatic tool for pediatric heart sounds segmentation. *Computing in Cardiology*, *38*.

Gill, D., Gavrieli, N., & Intrator, N. (2005). Detection and identification of heart sounds using homomorphic envelogram and self-organizing probabilistic model. *Computers in Cardiology*, *32*. https://doi.org/10.1109/CIC.2005.1588267

Gnitecki, J. [J.], & Moussavi, Z. [Z.]. (2003). Variance Fractal Dimension Trajectory as a Tool for Heart Sound Localization in Lung Sounds Recordings. *Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings*, *3*.

Gnitecki, J. [January], & Moussavi, Z. M. (2007). Separating heart sounds from lung sounds - Accurate diagnosis of respiratory disease depends on understanding noises. https://doi.org/10.1109/MEMB.2007.289118

Goldberger, A. L., Amaral, L. A., Glass, L., Hausdorff, J. M., Ivanov, P. C., Mark, R. G., Mietus, J. E., Moody, G. B., Peng, C. K., & Stanley, H. E. (2000). PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals. *Circulation*, *101*(23). https://doi.org/10.1161/01.cir.101.23.e215

Golpaygani, A. T., Abolpour, N., Hassani, K., Bajelani, K., & Doyle, D. J. (2015). Detection and identification of S1 and S2 heart sounds using wavelet decomposition method. *International Journal of Biomathematics*, *8*(6). https://doi.org/10.1142/S1793524515500783

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.

Gupta, C. N., Palaniappan, R., Swaminathan, S., & Krishnan, S. M. (2007). Neural network classification of homomorphic segmented heart sounds. *Applied Soft Computing Journal*, *7*(1). https://doi.org/10.1016/j.asoc.2005.06.006

Hadjileontiadis, L. J. (1997). Adaptive reduction of heart sounds from lung sounds using fourth-order statistics. *IEEE Transactions on Biomedical Engineering*, *44*(7). https://doi.org/10.1109/10.594906

Hadjileontiadis, L. J. (2005). Wavelet-based enhancement of lung and bowel sounds using fractal dimension thresholding - Part I: Methodology. *IEEE Transactions on Biomedical Engineering*, *52*(6). https://doi.org/10.1109/TBME.2005.846706

Hafke-Dys, H., Breborowicz, A., Kleka, P., Kociński, J., & Biniakowski, A. (2019). The accuracy of lung auscultation in the practice of physicians and medical students. *PLOS ONE*, *14*(8). https://doi.org/10.1371/journal.pone.0220606

Haghighi-Mood, A., & Torry, J. N. (1995). Sub-band energy tracking algorithm for heart sound segmentation. *Computers in Cardiology*. https://doi.org/10.1109/cic.1995.482711

Hamza Cherif, L., Debbal, S. M., & Bereksi-Reguig, F. (2008). Segmentation of heart sounds and heart murmurs. *Journal of Mechanics in Medicine and Biology*, *8*(4). https://doi.org/10.1142/S0219519408002759

Hassani, K., Bajelani, K., Navidbakhsh, M., Doyle, D. J., & Taherian, F. (2014). Heart sound segmentation based on homomorphic filtering. *Perfusion (United Kingdom)*, *29*(4). https://doi.org/10.1177/0267659114523463

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. *2015 IEEE International Conference on Computer Vision (ICCV)*. https://doi.org/10.1109/ICCV.2015.123

Hossain, I., & Moussavi, Z. [Zahra]. (2003). An overview of heart-noise reduction of lung sound using wavelet transform based filter. *Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings*, *1*. https://doi.org/10.1109/iembs.2003.1279719

Huiying, L., Sakari, L., & Iiro, H. (1997). Heart sound segmentation algorithm using wavelet decomposition and reconstruction. *Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings*. https://doi.org/10.1109/iembs.1997.757028

Iaizzo, P. A. (2005). *Handbook of cardiac anatomy, physiology, and devices*. https://doi.org/10.1007/978-1-59259-835-9

Iwata, A. [A.], Ishii, N., Suzumura, N., & Ikegaya, K. (1980). Algorithm for detecting the first and the second heart sounds by spectral tracking. *Medical & Biological Engineering & Computing*, *18*(1). https://doi.org/10.1007/BF02442475

Iwata, A. [Akira], Suzumura, N., & Ikegaya, K. (1977). Pattern classification of the phonocardiogram using linear prediction analysis. *Medical & Biological Engineering & Computing*, *15*(4). https://doi.org/10.1007/BF02457994

Iyer, V. K., Ramamoorthy, P. A., Fan, H., & Ploysongsang, Y. (1986). Reduction of Heart Sounds from Lung Sounds by Adaptive Filtering. *IEEE Transactions on Biomedical Engineering*, *BME-33*(12). https://doi.org/10.1109/TBME.1986.325693

J. Hadjileontiadis, L., & M. Panas, S. (1998). A wavelet-based reduction of heart sound noise from lung sounds. *International Journal of Medical Informatics*. https://doi.org/10.1016/S1386-5056(98)00137-3

Jones, A., Jones, R. D., Kwong, K., & Burns, Y. (1999). Effect of positioning on recorded lung sound intensities in subjects without pulmonary dysfunction. *Physical Therapy*. https://doi.org/10.1093/ptj/79.7.682

Kandaswamy, A., Kumar, C. S., Ramanathan, R. P., Jayaraman, S., & Malmurugan, N. (2004). Neural classification of lung sounds using wavelet coefficients. *Computers in Biology and Medicine*, *34*(6). https://doi.org/10.1016/S0010-4825(03)00092-1

Kattepur, A. K., Jin, F., & Sattar, F. (2010). Single channel source separation for convolutive mixtures with application to respiratory sounds. *BIOSIGNALS 2010 - Proceedings of the 3rd International Conference on Bio-inpsired Systems and Signal Processing, Proceedings*.

Khan, A. K., Onoue, T., Hashiodani, K., Fukumizu, Y., & Yamauchi, H. (2010). Signal and noise separation in medical diagnostic system based on independent component analysis. *IEEE Asia-Pacific Conference on Circuits and Systems, Proceedings, APCCAS*. https://doi.org/10.1109/APCCAS.2010.5775018

Khan, A., Sohail, A., Zahoora, U., & Qureshi, A. S. (2020). A survey of the recent architectures of deep convolutional neural networks. *Artificial Intelligence Review*, *53*(8). https://doi.org/10.1007/s10462-020-09825-6

Khan, T. E. A., & Vijayakumar, P. (2010). Separating Heart Sound from Lung Sound Using LabVIEW. *International Journal of Computer and Electrical Engineering*. https://doi.org/10.7763/ijcee.2010.v2.188

Kingma, D. P., & Ba, J. L. (2015). Adam: A method for stochastic optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, *2*. https://doi.org/10.1061/(ASCE)GT.1943-5606.0001284

Kumar, D. [D.], Carvalho, P., Antunes, M., Gil, P., Henriques, J., & Eugénio, L. (2006). A new algorithm for detection of S1 and S2 heart sounds. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, *2*. https://doi.org/10.1109/icassp.2006.1660559

Kumar, D. [D.], Carvalho, P., Antunes, M., Henriques, J., Eugénio, L., Schmidt, R., & Habetha, J. (2006). Detection of S1 and S2 heart sounds by high frequency signatures. *Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings*. https://doi.org/10.1109/IEMBS.2006.260735

Kumar, D. [Dinesh], Carvalho, P., Antunes, M., Henriques, J., Maldonado, M., Schmidt, R., & Habetha, J. (2006). Wavelet transform and simplicity based heart murmur segmentation. *Computers in Cardiology*, *33*.

Kutyniok, G. (2013). Theory and applications of compressed sensing. *GAMM-Mitteilungen*, *36*(1). https://doi.org/10.1002/gamm.201310005

LeCun, Y. [Y.], Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation*, *1*(4). https://doi.org/10.1162/neco.1989.1.4.541

LeCun, Y. [Yann], Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11). https://doi.org/10.1109/5.726791

Lee, D. D., & Seung, H. S. (2001). Algorithms for non-negative matrix factorization. *Advances in Neural Information Processing Systems*.

Lehner, R. J., & Rangayyan, R. M. (1987). A Three–Channel Microcomputer System for Segmentation and Characterization of the Phonocardiogram. *IEEE Transactions on Biomedical Engineering*. https://doi.org/10.1109/TBME.1987.326060

Liang, H., Lukkarinen, S., & Hartimo, I. (1997). Heart sound segmentation algorithm based on heart sound envelogram. *Computers in Cardiology*. https://doi.org/10.1109/cic.1997.647841

Liang, H., Lukkarinen, S., & Hartimo, I. (1998). A boundary modification method for heart sound segmentation algorithm. *Computers in Cardiology*, *0*(0). https://doi.org/10.1109/cic.1998.731943

Lima, C. S., & Barbosa, D. (2008). Automatic segmentation of the second cardiac sound by using wavelets and hidden Markov models. *Proceedings of the 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS'08 - "Personalized Healthcare through Technology"*. https://doi.org/10.1109/iembs.2008.4649158

Lin, C. S., & Hasting, E. (2013). Blind source separation of heart and lung sounds based on nonnegative matrix factorization. *ISPACS 2013 - 2013 International Symposium on Intelligent Signal Processing and Communication Systems*. https://doi.org/10.1109/ISPACS.2013.6704646

Liu, C., Springer, D., Li, Q., Moody, B., Juan, R. A., Chorro, F. J., Castells, F., Roig, J. M., Silva, I., Johnson, A. E., Syed, Z., Schmidt, S. E., Papadaniil, C. D., Hadjileontiadis, L., Naseri, H., Moukadem, A., Dieterlen, A., Brandt, C., Tang, H., ... Clifford, G. D. (2016). An open access database for the evaluation of heart sound algorithms. *Physiological Measurement*, *37*(12). https://doi.org/10.1088/0967-3334/37/12/2181

Liu, C.-L. (2010). A Tutorial of the Wavelet Transform. *History*.

Luisada, A. A., Mendoza, F., & Alimurung, M. M. (1949). The duration of normal heart sounds. *British heart journal*, *11*(1). https://doi.org/10.1136/hrt.11.1.41

Makhoul, J. (1975). Spectral Linear Prediction: Properties and Applications. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, *23*(3). https://doi.org/10.1109/TASSP.1975.1162685

Malarvili, M. B., Kamarulafizam, I., Hussain, S., & Helmi, D. (2003). Heart sound segmentation algorithm based on instantaneous energy of electrocardiogram. *Computers in Cardiology*, *30*. https://doi.org/10.1109/cic.2003.1291157

Martinez-Alajarin, J., & Ruiz-Merino, R. (2005). Efficient method for events detection in phonocardiographic signals. *Bioengineered and Bioinspired Systems II*, *5839*. https://doi.org/10.1117/12.608203

Messer, S. R., Agzarian, J., & Abbott, D. (2001). Optimal wavelet denoising for phonocardiograms. *Microelectronics Journal*. https://doi.org/10.1016/S0026-2692(01)00095-7

Messner, E., Zöhrer, M., & Pernkopf, F. (2018). Heart sound segmentation - An event detection approach using deep recurrent neural networks. *IEEE Transactions on Biomedical Engineering*, *65*(9). https://doi.org/10.1109/TBME.2018.2843258

Meziani, F., Debbal, S. M., & Atbi, A. (2012). Analysis of phonocardiogram signals using wavelet transform. https://doi.org/10.3109/03091902.2012.684830

Mondal, A., Banerjee, P., & Somkuwar, A. (2017). Enhancement of lung sounds based on empirical mode decomposition and Fourier transform algorithm. *Computer Methods and Programs in Biomedicine*. https://doi.org/10.1016/j.cmpb.2016.10.025

Mondal, A., Bhattacharya, P. S., & Saha, G. (2011). Reduction of heart sound interference from lung sound signals using empirical mode decomposition technique. *Journal of Medical Engineering and Technology*. https://doi.org/10.3109/03091902.2011.595529

Mondal, A., Bhattacharya, P., & Saha, G. (2013). An automated tool for localization of heart sound components S1, S2, S3 and S4 in pulmonary sounds using Hilbert transform and Heron's formula. *SpringerPlus*, *2*(1). https://doi.org/10.1186/2193-1801-2-512

Mondal, A., Saxena, I., Tang, H., & Banerjee, P. (2018). A Noise Reduction Technique Based on Nonlinear Kernel Function for Heart Sound Analysis. *IEEE Journal of Biomedical and Health Informatics*. https://doi.org/10.1109/JBHI.2017.2667685

Moukadem, A., Dieterlen, A., Hueber, N., & Brandt, C. (2013). A robust heart sounds segmentation module based on S-transform. *Biomedical Signal Processing and Control*, *8*(3). https://doi.org/10.1016/j.bspc.2012.11.008

Moukadem, A., Schmidt, S., & Dieterlen, A. (2015). High order statistics and time-frequency domain to classify heart sounds for subjects under cardiac stress test. *Computational and Mathematical Methods in Medicine*, *2015*. https://doi.org/10.1155/2015/157825

Mubarak, Q. u. A., Akram, M. U., Shaukat, A., Hussain, F., Khawaja, S. G., & Butt, W. H. (2018). Analysis of PCG signals using quality assessment and homomorphic filters for localization and classification of heart sounds. *Computer Methods and Programs in Biomedicine*, *164*. https://doi.org/10.1016/j.cmpb.2018.07.006

Naseri, H., & Homaeinezhad, M. R. (2013). Detection and boundary identification of phonocardiogram sounds using an expert frequency-energy based metric. *Annals of Biomedical Engineering*, *41*(2). https://doi.org/10.1007/s10439-012-0645-x

Nazeran, H. (2007). Wavelet-based segmentation and feature extraction of heart sounds for intelligent PDA-based phonocardiography. *Methods of Information in Medicine*, *46*(2). https://doi.org/10.1055/s-0038-1625394

Nersisson, R., & Noel, M. M. (2017). Hybrid Nelder-Mead search based optimal Least Mean Square algorithms for heart and lung sound separation. *Engineering Science and Technology, an International Journal*. https://doi.org/10.1016/j.jestch.2017.02.005

Nigam, V., & Priemer, R. (2005). Accessing heart dynamics to estimate durations of heart sounds. *Physiological Measurement*, *26*(6). https://doi.org/10.1088/0967-3334/26/6/010

Nivitha Varghees, V., & Ramachandran, K. I. (2017). Effective heart sound segmentation and murmur classification using empirical wavelet transform and instantaneous phase for electronic stethoscope. *IEEE Sensors Journal*, *17*(12). https://doi.org/10.1109/JSEN.2017.2694970

Nocedal, J., & Wright, S. J. (2006). Numerical Optimization Second Edition. *Ear and hearing*. https://doi.org/10.1097/00003446-199604000-00005

Noman, F., Salleh, S. H., Ting, C. M., Samdin, S. B., Ombao, H., & Hussain, H. (2020). A Markov-Switching Model Approach to Heart Sound Segmentation and Classification. *IEEE Journal of Biomedical and Health Informatics*, *24*(3). https://doi.org/10.1109/JBHI.2019.2925036

Oliveira, J., Renna, F., Mantadelis, T., & Coimbra, M. (2019). Adaptive Sojourn Time HSMM for Heart Sound Segmentation. *IEEE Journal of Biomedical and Health Informatics*, *23*(2). https://doi.org/10.1109/JBHI.2018.2841197

Omari, T., & Bereksi-Reguig, F. (2015). An automatic wavelet denoising scheme for heart sounds. *International Journal of Wavelets, Multiresolution and Information Processing*. https://doi.org/10.1142/S0219691315500162

Oskiper, T., & Watrous, R. (2002). Detection of the first heart sound using a time-delay neural network. *Computers in Cardiology*, *29*. https://doi.org/10.1109/cic.2002.1166828

Palaniappan, R., Sundaraj, K., & Ahamed, N. U. (2013). Machine learning in lung sound analysis: A systematic review. *Biocybernetics and Biomedical Engineering*, *33*(3). https://doi.org/10.1016/j.bbe.2013.07.001

Papadaniil, C. D., & Hadjileontiadis, L. J. (2014). Efficient heart sound segmentation and extraction using ensemble empirical mode decomposition and kurtosis features. *IEEE Journal of Biomedical and Health Informatics*, *18*(4). https://doi.org/10.1109/JBHI.2013.2294399

Pasterkamp, H., Kraman, S. S., & Wodicka, G. R. (1997). Respiratory sounds: Advances beyond the stethoscope. https://doi.org/10.1164/ajrccm.156.3.9701115

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*.

Pedrosa, J., Castro, A., & Vinhoza, T. T. (2014). Automatic heart sound segmentation and murmur detection in pediatric phonocardiograms. *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC 2014*. https://doi.org/10.1109/EMBC.2014.6944078

Phinyomark, A., Phukpattaranont, P., & Limsakul, C. (2014). Applications of variance fractal dimension: A survey. https://doi.org/10.1142/S0218348X14500030

Potdar, R. M., Mishra, A., Sharma, V., & Roy, T. (2012). *"Performance Evaluation of Different Adaptive Filtering Algorithms for Reduction of Heart Sound from Lung Sound"* (tech. rep.).

Pourazad, M. T., Moussavi, Z. [Z.], Farahmand, F., & Ward, R. K. (2005). Heart sounds separation from lung sounds using independent component analysis. *Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings*. https://doi.org/10.1109/iembs.2005.1617037

Pourazad, M. T., Moussavi, Z. [Z.], & Thomas, G. (2006). Heart sound cancellation from lung sound recordings using time-frequency filtering. *Medical and Biological Engineering and Computing*. https://doi.org/10.1007/s11517-006-0030-8

Powers, D. M. W. (2020). Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *arXiv preprint arXiv:2010.16061*.

Proakis, J. G. (2001). *Digital Communications*. McGraw-Hill. https://books.google.cl/books?id=aUp2QgAACAAJ

Rajan, S., Budd, E., Stevenson, M., & Doraiswami, R. (2006). Unsupervised and uncued segmentation of the fundamental heart sounds in phonocardiograms using a time-scale representation. *Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings*. https://doi.org/10.1109/IEMBS.2006.260777

Reed, T. R., Reed, N. E., & Fritzson, P. (2004). Heart sound analysis for symptom detection and computer-aided diagnosis. *Simulation Modelling Practice and Theory*. https://doi.org/10.1016/j.simpat.2003.11.005

Reichert, S., Gass, R., Brandt, C., & Andrès, E. (2008). Analysis of Respiratory Sounds: State of the Art. *Clinical medicine. Circulatory, respiratory and pulmonary medicine*, *2*. https://doi.org/10.4137/ccrpm.s530

Renna, F., Oliveira, J., & Coimbra, M. T. (2019). Deep Convolutional Neural Networks for Heart Sound Segmentation. *IEEE Journal of Biomedical and Health Informatics*, *23*(6). https://doi.org/10.1109/JBHI.2019.2894222

Rocha, B. M., Filos, D., Mendes, L., Serbes, G., Ulukaya, S., Kahya, Y. P., Jakovljevic, N., Turukalo, T. L., Vogiatzis, I. M., Perantoni, E., Kaimakamis, E., Natsiavas, P., Oliveira, A., Jácome, C., Marques, A., Maglaveras, N., Pedro Paiva, R., Chouvarda, I., & De Carvalho, P. (2019). An open access database for the evaluation of respiratory sound classification algorithms. *Physiological Measurement*, *40*(3). https://doi.org/10.1088/1361-6579/ab03ea

Rudnitskii, A. G. (2014). Using nonlocal means to separate cardiac and respiration sounds. *Acoustical Physics*. https://doi.org/10.1134/S1063771014050121

Saha, G., & Kumar, P. (2004). An efficient heart sound segmentation algorithm for cardiac diseases. *Proceedings of the IEEE INDICON 2004 - 1st India Annual Conference*. https://doi.org/10.1109/indico.2004.1497768

Sankur, B., Kahya, Y. P., Çağatay Güler, E., & Engin, T. (1994). Comparison of AR-based algorithms for respiratory sounds classification. *Computers in Biology and Medicine*, *24*(1). https://doi.org/10.1016/0010-4825(94)90038-8

Sarkar, M., Madabhavi, I., Niranjan, N., & Dogra, M. (2015). Auscultation of the respiratory system. *Annals of Thoracic Medicine*, *10*(3). https://doi.org/10.4103/1817-1737.160831

Schmidt, M. N. (2008). Single-channel source separation using non-negative matrix factorization. *Computational Intelligence*.

Schmidt, S. E., Holst-Hansen, C., Graff, C., Toft, E., & Struijk, J. J. (2010). Segmentation of heart sound recordings by a duration-dependent hidden Markov model. *Physiological Measurement*, *31*(4). https://doi.org/10.1088/0967-3334/31/4/004

Sedighian, P., Subudhi, A. W., Scalzo, F., & Asgari, S. (2014). Pediatric heart sound segmentation using Hidden Markov Model. *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC 2014*. https://doi.org/10.1109/EMBC.2014.6944869

Sengupta, N., Sahidullah, M., & Saha, G. (2016). Lung sound classification using cepstral-based statistical features. *Computers in Biology and Medicine*, *75*. https://doi.org/10.1016/j.compbiomed.2016.05.013

Sepehri, A. A., Gharehbaghi, A., Dutoit, T., Kocharian, A., & Kiani, A. (2010). A novel method for pediatric heart sound segmentation without using the ECG. *Computer Methods and Programs in Biomedicine*, *99*(1). https://doi.org/10.1016/j.cmpb.2009.10.006

Shah, G., Koch, P., & Papadias, C. B. (2015). On the blind recovery of cardiac and respiratory sounds. *IEEE Journal of Biomedical and Health Informatics*. https://doi.org/10.1109/JBHI.2014.2349156

Shah, G., & Papadias, C. (2013). Separation of cardiorespiratory sounds using time-frequency masking and sparsity. *2013 18th International Conference on Digital Signal Processing, DSP 2013*. https://doi.org/10.1109/ICDSP.2013.6622792

Shah, G., & Papadias, C. B. (2013). Blind recovery of cardiac and respiratory sounds using non-negative matrix factorization & time-frequency masking. *13th IEEE International Conference on BioInformatics and BioEngineering, IEEE BIBE 2013*. https://doi.org/10.1109/BIBE.2013.6701542

Shanthakumari, G., & Priya, E. (2019). Performance Analysis: Preprocessing of Respiratory Lung Sounds. *Communications in Computer and Information Science*, *890*. https://doi.org/10.1007/978-981-13-9129-3{_}21

Smith, J. O. (2007). Mathematics of the Discrete Fourier Transform. *Identity*.

Springer, D. B., Tarassenko, L., & Clifford, G. D. (2016). Logistic regression-HSMM-based heart sound segmentation. *IEEE Transactions on Biomedical Engineering*, *63*(4). https://doi.org/10.1109/TBME.2015.2475278

Sun, H., Chen, W., & Gong, J. (2013). An improved empirical mode decomposition-wavelet algorithm for phonocardiogram signal denoising and its application in the first and second heart sound extraction. *Proceedings of the 2013 6th International Conference on Biomedical Engineering and Informatics, BMEI 2013*. https://doi.org/10.1109/BMEI.2013.6746931

Sun, S., Jiang, Z., Wang, H., & Fang, Y. (2014). Automatic moment segmentation and peak detection analysis of heart sound pattern via short-time modified Hilbert transform. *Computer Methods and Programs in Biomedicine*, *114*(3). https://doi.org/10.1016/j.cmpb.2014.02.004

Sun, S., Wang, H. [Haibin], Jiang, Z., Fang, Y., & Tao, T. (2014). Segmentation-based heart sound feature extraction combined with classifier models for a VSD diagnosis system. *Expert Systems with Applications*, *41*(4 PART 2). https://doi.org/10.1016/j.eswa.2013.08.076

Tang, H., Li, T., Qiu, T., & Park, Y. (2012). Segmentation of heart sounds based on dynamic clustering. *Biomedical Signal Processing and Control*, *7*(5). https://doi.org/10.1016/j.bspc.2011.09.002

Tsalaile, T., Naqvi, S. M., Nazarpour, K., Sanei, S., & Chambers, J. A. (2008). Blind source extraction of heart sound signals from lung sound recordings exploiting periodicity of the heart sound. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*. https://doi.org/10.1109/ICASSP.2008.4517646

Tsao, Y., Lin, T. H., Chen, F., Chang, Y. F., Cheng, C. H., & Tsai, K. H. (2019). Robust S1 and S2 heart sound recognition based on spectral restoration and multi-style training. *Biomedical Signal Processing and Control*, *49*. https://doi.org/10.1016/j.bspc.2018.10.014

Tseng, Y. L., Ko, P. Y., & Jaw, F. S. (2012). Detection of the third and fourth heart sounds using Hilbert-Huang transform. *BioMedical Engineering Online*, *11*. https://doi.org/10.1186/1475-925X-11-8

Vannuccini, L., Earis, J. E., Helistö, P., Cheetham, B. M., Rossi, M., Sovijärvi, A. R., & Vanderschoot, J. (2000). Capturing and preprocessing of respiratory sounds. *European Respiratory Review*, *10*(77).

Várady, P. (2001). Wavelet-based adaptive denoising of phonocardiographic records. *Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings*. https://doi.org/10.1109/iembs.2001.1020582

Varghees, V. N., & Ramachandran, K. I. (2014). A novel heart sound activity detection framework for automated heart sound analysis. *Biomedical Signal Processing and Control*, *13*(1). https://doi.org/10.1016/j.bspc.2014.05.002

Vepa, J., Tolay, P., & Jain, A. (2008). Segmentation of heart sounds using simplicity features and timing information. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*. https://doi.org/10.1109/ICASSP.2008.4517648

Vincent, E., Gribonval, R., & Févotte, C. (2006). Performance measurement in blind audio source separation. *IEEE Transactions on Audio, Speech and Language Processing*, *14*(4). https://doi.org/10.1109/TSA.2005.858005

Wang, H. [Hong], & Wang, L. Y. (2003). Multi-Sensor Adaptive Heart and Lung Sound Extraction. *Proceedings of IEEE Sensors*.

Wang, H. [Hong], Wang, L. Y., Zheng, H., Haladjian, R., & Wallo, M. (2004). Lung sound/noise separation for anesthesia respiratorymonitoring. *WSEAS Transactions on Systems*, *3*(4), 1839–1844.

Wang, P., Kim, Y., Ling, L. H., & Soh, C. B. (2005). First heart sound detection for phonocardiogram segmentation. *Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings*, *7 VOLS*. https://doi.org/10.1109/iembs.2005.1615733

Wang, Z., Nuno Da Cruz, J., & Wan, F. (2015). Adaptive Fourier decomposition approach for lung-heart sound separation. *2015 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications, CIVEMSA 2015*. https://doi.org/10.1109/CIVEMSA.2015.7158631

Welsby, P. D., Parry, G., & Smith, D. (2003). The stethoscope: Some preliminary investigations. *Postgraduate Medical Journal*, *79*(938).

Xie, S., Jin, F., Krishnan, S., & Sattar, F. (2012). Signal feature extraction by multi-scale PCA and its application to respiratory sound classification. *Medical & Biological Engineering & Computing*, *50*(7). https://doi.org/10.1007/s11517-012-0903-y

Yadollahi, A., & Moussavi, Z. M. (2006). A robust method for heart sounds localization using lung sounds entropy. *IEEE Transactions on Biomedical Engineering*, *53*(3). https://doi.org/10.1109/TBME.2005.869789

Yamaçh, M., Dokur, Z., & Ölmez, T. (2008). Segmentation of S1-S2 sounds in phonocardiogram records using wavelet energies. *2008 23rd International Symposium on Computer and Information Sciences, ISCIS 2008*. https://doi.org/10.1109/ISCIS.2008.4717964

Yan, Z., Jiang, Z., Miyamoto, A., & Wei, Y. (2010). The moment segmentation analysis of heart sound pattern. *Computer Methods and Programs in Biomedicine*, *98*(2). https://doi.org/10.1016/j.cmpb.2009.09.008

Ye, J. C., & Sung, W. K. (2019). Understanding geometry of encoder-decoder CNNs.

Yip, L., & Zhang, Y. T. (2001). Reduction of heart sounds from lung sound recordings by automated gain control and adaptive filtering techniques. *Annual Reports of the Research Reactor Institute, Kyoto University*, *3*. https://doi.org/10.1109/iembs.2001.1017196

Yuenyong, S., Nishihara, A., Kongprawechnon, W., & Tungpimolrut, K. (2011). A framework for automatic heart sound analysis without segmentation. *BioMedical Engineering Online*, *10*. https://doi.org/10.1186/1475-925X-10-13

Zhang, Z. Y. (2012). Nonnegative Matrix Factorization: Models, Algorithms and Applications. *Intelligent Systems Reference Library*, *24*. https://doi.org/10.1007/978-3-642-23242-8{_}6

Zhang, Z., Li, T., Ding, C., & Zhang, X. (2007). Binary matrix factorization with applications. *Proceedings - IEEE International Conference on Data Mining, ICDM*. https://doi.org/10.1109/ICDM.2007.99

Zheng, H., Wang, H., Wang, L. Y., & Yin, G. G. (2007). Cyclic system reconfiguration and time-split signal separation with applications to lung sound pattern analysis. *IEEE Transactions on Signal Processing*. https://doi.org/10.1109/TSP.2007.893736