National And Kapodistrian University Of Athens

Master Thesis

Comparison of parametric hidden semi-Markov models for analysing the different phases of cardiac cycle

Author:
Nektarios Kyrios

Supervisor:
Dr. Samis Trevezas
Assistant Professor of the
Department of Mathematics

Department of Mathematics

September 30, 2022

NATIONAL AND KAPODISTRIAN UNIVERSITY OF ATHENS

Abstract

Department of Mathematics

Master In Statistical And Operational Research

Comparison of parametric hidden semi-Markov models for analysing the different phases of cardiac cycle

by Nektarios Kyrios

In this master thesis we present modern statistical techniques of heart sound segmentation via hidden semi-Markov models (HSMMs). The cardiac cycle can be broken down into different phases, these are the phase of S_1 sound, Systole, S_2 sound diastole. So modeling this heart cycle periodicity through an HSMM gives very important information about the health status of the heart and can be used to detect abnormalities in its operation. Using electric and magnetic signals we make an effort to accurately predict the sojourn times of cardiac cycle phases suggesting also the Hidden semi-Markov models appropriate to predict. In this thesis, different parametric Hidden semi-Markov models are compared in their predictive ability and this is done using real data. To achieve that we extracted three different signal features (Homomorphic envelogram, Hilbert envelope and Power Spectral Density envelope) apart from original signal. Continuously we test all the different modeling assumptions. The hidden structure of the model concerning the different phases of cardiac cycle which alternate in time and can be considered as the states of an unobserved semi-Markov chain. Using different assumptions. For the state-dependent sojourn time distributions, we create different HSMMs which compare. So, this thesis extends the results are presented in the thesis of Konstantina Katachana "Signal Processing and Statistical Analysis of the Cardiac Cycle via hidden semi-Markov models" comparing different HSMMs on their ability to accurately identify the fundamental phases of heart cycle.

Acknowledgements

I would like to take this opportunity to express my gratitude to each and everyone who helped me grow as a student and as a person all through this spell of my life.

It gives me great pleasure in acknowledging the support and help of my committee chair, Dr Samis Trevezas for all his guidance through every stage of this thesis research. I attribute the level of my Masters degree to his encouragement and effort and without him this thesis would not have been completed.

I would like to thank Dr Athanasia Manou and Dr Apostolos Burnetas for being the members of committe for this thesis. It was a pleasure taking courses offered by them which were not only interesting but offered a lot of challenges. I would also like to thank all my professors in the Department of Mathematics of National and Kapodistrian University of Athens for their great contribution to me, both for the wide background knowledge and personally.

This thesis is dedicated to my parents who have given me the opportunity of education and support throughout my life. Last but not least, I would like express my love to my friends for their constant understanding and backing me in the toughest of times. I owe a lot to each and every one of them.

Contents

Chapter 1

Biological Background

1

1.1 Cardiovascular Diseases

The cardiovascular system consists of the heart and blood vessels. The problems that can be arised within the cardiovascular system are many, for example, endocarditis rheumatic heart disease, abnormalities in the condition system, among others that we are going to refer below

1. Coronary artery disease (CAD): Sometimes referred to as Coronary Heart Disease (CHD), results from decreased myocardial perfusion that causes angina, myocardial infarction (MI) and/or heart failure. It accounts for one-third to one-half of the cases of CVD.

2. Cerebrovascular disease (CVD): Including stroke and transient ischemic attack (TIA)

3. Peripheral artery disease (PAD): Particularly arterial disease involving the limbs that may result in claudication.

4. Aortic atherosclerosis: Including thoracic and abdominal aneurysms

It is paramount the fact that cardiovascular diseases are the leading cause of death worldwide except Africa. In 2015, CVDs led to death 17,9 million people 32,18 % of global deaths, up from 12,3 million 25,8 % of global deaths in 1990. Deaths, at a given age, from CVD are more common and have been increasing in much of the developing world, while rates have declined in most of the developed world since the 1970s. It's worthwhile that tobacco use, unhealthy diet, obesity, high blood cholesterol, excessive alcohol consumption, lack of sleep and others consists risk factors which are associated with the most CVDs. It is estimated that up to 90 % of CVD involves improving risk factors through: healthy eating, exercise avoidance of tobacco smoke and limiting alcohol intake.

Treating risk factors, such as high blood pressure, blood lipids and diabetes is also beneficial. Unfortunately, CVDs have often no symptoms of the underlying disease of the blood vessels. A warming of underlying diseases may be a heart attack or stroke. That is, early detection plays an essential role.

As CVDs are the most important factor related to human mortality, a lot of research effort has been made in the development of medical techniques and associated tools for the diagnosis of such diseases. In this application we focus on a non-invasive automated method. Such methods are often preferable due to their simplicity and lower cost. Ordinary investigations in cardiovascular disease include :

- Myocardial perfusion scan (MPS)(sestamibi scan / thallium scan) MPS is a noninvasive nuclear medicine scan that examines myocardial perfusion both at rest and under stress using a small amount of a radioactive substance , called a radionuclide.
- Cardiac computerised tomography (CT) Cardiac CT uses CT technology to provide detailed heart image. This may include the identification of anatomical abnormalities such as aneurysms or valve dysfunction.
- Cardiac MRI uses high intensity magnetic fields and radio-frequency to produce 3D images with high resolution. The image provides accurate information about cardiac volumes, muscle mass, contractility, tissue scarring and ejection fraction.
- Echocardiography is the gold standard investigation for the diagnosis of heart failure and should be re-evaluated at least every 2 years after completion of the medication titration. It provides an ultrasound image of the cardiac anatomy and can identify the type and region of heart abnormalities.
- Blood tests Identifies the presence of infection, anaemia and other blood disorders.
- Electrocardiography (ECG) ECG records the electrical activity of the heart. It is a simple test that identifies heart rate, conduction disturbances, myocardial ischaemia and possible structural defects. ECG aids in the diagnosis of underlying causes of heart disease such as coronary artery disease or arrythmias.
- Chest X-Ray (CXR) A chest X-ray helps differentiate between respiratory and cardiac causes of dyspnoea. In people suffering from heart diseases cardiomegaly, interstitial oedema, pulmonary oedema and pleural effusions are common findings.

- Coronary angiography Coronary angiography examines the integrity of the coronary arteries by inserting a catheter into the coronary vasculature and using dye to produce the image. Possible sources of symptoms can be identified through the image.

1.2 Cardiac Cycle

The heart is a muscular organ that serves to collect deoxygenated blood from all parts of the body, carries it to the lungs to be oxygenated and release carbon dioxide. Then, it transports the oxygenated blood from the lungs and distributes it to all the body parts

- The heart pumps around 7.200 litres of blood in a day throughout the body.
- The heart is situated at the centre of the chest and points slightly towards the left.
- On average, the heart beats about 100.000 times a day, i.e, around 3 billion beats in a lifetime.
- An adults heart beats about 60 to 80 times per minute, and newborn babies heart faster that an adult which is about 70 to 190 beats per minute.

As for the structure of heart, this is subdivided by septa into right and left halves, and a constriction subdivides each half of the organ into two cavities, the upper cavity being called the atrium, the lower the ventricle. The heart, therefore, consists of four chambers

- right atrium
- left atrium
- right ventricle
- left ventricle

It is best to remember the four chambers and four valves in order of the series that blood travels through the heart:

- Venous blood returning from the body drains into the right atrium via the SVC, IVC and coronary sinus.
- The right atrium pumps blood through the tricuspid valve into the right ventricle.
- The right ventricle pumps blood through the pulmonary semilunar valve into the pulmonary trunk to be oxygenated in the lungs.
- Blood returning from the lungs drains into the left atrium via the four pulmonary veins.

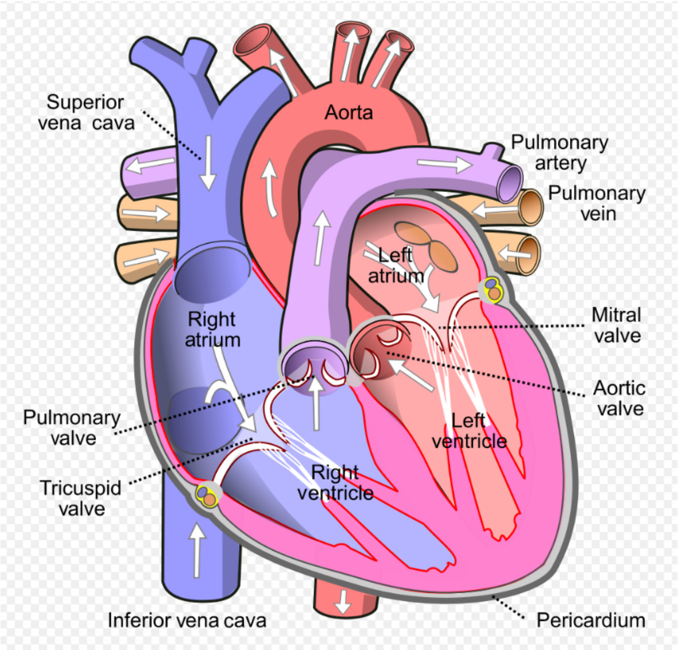

FIGURE 1.1: Anatomy of the heart
Source: Wikipedia

- The left atrium pumps blood through the bicuspid (mitral) valve into the left ventricle.

- The left ventricle pumps blood through the aortic semilunar valve into the ascending aorta to supply the body.

In addition the heart has four valves. All four valves of the heart have a singular purpose: allowing forward flow of blood but preventing backward flow. The outflow of each chamber is guarded by a heart valve:

Atrioventricular valves between the atria and ventricles

1. tricuspid valve (right side of the heart): Controls blood flow between the right atrium and right ventricle.

2. mitral valve/bicuspid valve (left side of the heart): Lets oxygen-rich blood from your lungs pass from the left atrium into the left ventricle.

Semilunar valves which are located in the outflow tracts of the ventricles

1. aortic valve(left side heart): Opens the way for oxygen-rich blood to pass from the left ventricle to your body's largest artery, called aorta.

2. pulmonary valve (right side heart): Controls blood flow from the right ventricle into pulmonary arteries, which carry blood to your lungs to pick up oxygen.

Lets describe the cardiac cycle. The cardiac cycle is the sequence of events in which the heart contracts and relaxes with every heartbeat. The period of time during which the ventricles contract, forcing blood out into the aorta and main pulmonary artery, is known as systole, while the period during which the ventricles relax and refill with blood is known as diastole. The atria and ventricles work in correct, so in systole when the ventricles are contracting, the atria are relaxed and collecting blood. When the ventricles are relaxed in diastole, the atria contract to pump blood to the ventricles. This coordination ensures blood is pumped efficiently to the body. In our application we focus on systole and diastole. Also we are interested of the rhythmic noises accompanying heartbeat. These are two distinct sounds which are heard through the stethoscope a low, slightly prologned "lub" S_1 sound occuring at the beginning of ventricular contraction or systole, and produced by closure of the mitral and tricuspid valves, and a sharper, higher-pitched "dup" S_2, caused by closure of aortic and pulmonary valves at the end of systole.

1.3 ECG and PCG

An electrocardiogram - abbreviated as EKG or ECG - is a test that measures the electrical activity of the heartbeat. With each beat, an electrical impulse (or "wave") travels through the heart. This wave causes the muscle to squeeze and pump blood from the heart. A normal heartbeat on ECG will show the timing of the top and lower chambers.

The right and left atria or upper chambers make the first wave called a "P-wave"-following a flat line when the electrical impulse goes to the bottom chambers. The right and left bottom chambers or ventricles make the next wave called a "QRS complex". The final wave or "T wave" represents electrical recovery or return to a resting state for the ventricles.

ECG is very useful because gives two major kinds of information. First, by measuring time intervals on the ECG a doctor can determine how long electrical wave takes to pass through the heart. Finding out how long a wave takes to travel from one part of the heart to the next shows if the electrical activity is normal or slow, fast irregular. Second, by measuring the amount of electrical activity passing through the heart muscle, a cardiologist may be able to find out if parts of the heart are too large or overworked.

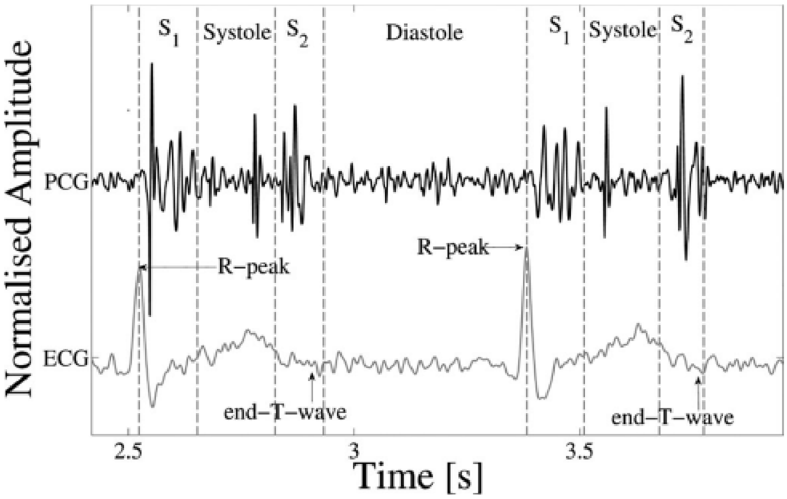

FIGURE 1.2: Example of an ECG-labeled PCG, with the ECG, PCG, and four states of the heart cycle (S1 , systole, S2 , and diastole) shown. The R-peak and end-T-wave are labeled as references for defining the approximate positions of S1and S2 , respectively. Midsystolic clicks, typical of mitral valve prolapse, can be seen.
Source: Physionet

As for the phonocardiogram, a phonocardiogram (or PCG) is a plot of high-fidelity recording of the sounds and murmurs made by the heart with the help of machine called the phonocardiogram; thus, phonocardiography is the recording of all the sounds made by the heart during a cardiac cycle. Heart sounds result from vibrations created by the closure of the heart valves. There are at least two; the first (S1) is produced when atrioventricular valves (tricuspid and mitral) close at the beginning of systole and the second (S2) when the aortic valve and pulmonary valve (semilunar valves) close at the end of systole. Phonocardiogram is a very important tool since it allows the detection of subaudible sounds and murmurs and makes a permanent record of these events. In contrast, the stethoscope cannot always detect all such sounds or murmurs and provides no record of their occurence.

The ability to quantitate the sounds made by the heart provides information not readily available from more sophisticated tests and provides vital information about the effects of certain drugs on the heart. It is also an effective method for tracking the progress of a patient's disease.

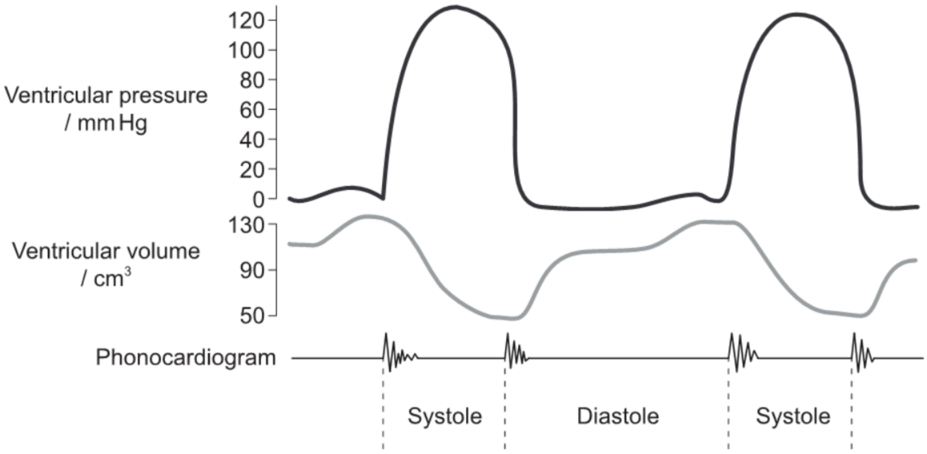

FIGURE 1.3: The diagram shows changes in the pressure and volume of the left ventricle during normal heart beat. The phonocardiogram records heart sounds during the cardiac cycle
Source: Wiggers, Carl J. 1923.Modern Aspects of Circulation in the Health and Disease, 2nd ed. Philadelphia: Lea and Febiger, p. 97.

Chapter 2

Data preprocessing

Data preprocessing is an iterative process for the transformation of the raw data into understandable and usable forms. Raw datasets are usually characterized by incompleteness, inconsistencies, lacking in behavior and also many times are noisy. Data preprocessing is essential to make our data easier to handle and to obtain more reliable results. There are many techniques to do it; we mainly work with data normalization and feature extraction[41]. In this chapter we describe the types of normalization and filtering techniques that are relative to the the dataset which is available in this master thesis, as well as the different features that are extracted and tested along with the original signal. Finally, we describe the way that the annotated data are used in order to properly label the heart sounds.

1 Normalization

Normalization techniques enable us to reduce the scale of the variables and make an entire set of values more balanced and easier to handle. Let's see some of them:

- In real world scenarios, we often work with unevenly distributed data, "suffering" from high skewness or/and values differing by several orders of magnitude. In such cases, the easiest way to scale them is through a *log-transformation*, e.g., in the following table

y_i	1000	29000	345	500	872	3223
$y_i^{'}$	6.90775	10.27505	5.84354	6.21460	6.77078	8.07806

notice that if $y_i^{'} = \log y_i$, then $\{y_i^{'}\}$ seem to be easier to handle than $\{y_i\}$.

- Another efficient way of Normalizing values is through the *Min-Max Scaling* method. With this method, the data values will finally range from 0 to 1. Consequently, the effect of outliers on the data values suppresses to a certain extent. Moreover, it often reduces standard deviation. This normalization procedure is described by the transformation

$$y_i' = \frac{y_i - \min_i y_i}{R},$$

where $\{y_i\}$ are the initial values of the signal and $R = \max_i y_i - \min_i y_i$ is the data range. In the dataset described above, we have $\min_i y_i = 50$, $\max_i y_i = 29000$ and $R = 28950$, so we have:

y_i	1000	29000	345	500	872	3223
y_i'	0.02286	1	0	0.00541	0.01849	0.10098

- A third method concerns the *Standard scaling*, also known as Standardization of values, which consists in scaling the data values in such a way that the mean and the variance of the values of every variable become 0 (centering) and 1 respectively. If $\overline{y}$ is the arithmetic mean and s_y the standard deviation of the initial values, then the normalized data result from the following standardization procedure:

$$y_i' = \frac{y_i - \overline{y}}{s_y}.$$

In our numerical example, the normalization corresponds to the following table:

y_i	1000	29000	345	500	872	3223
y_i'	-0.42301	2.03265	-0.48046	-0.46687	-0.43424	-0.22806

2 Butterworth band-pass filter

A Butterworth filter is a type of signal processing filter designed to have a frequency response as flat as possible in the passband. Hence the Butterworth filter is also known as "maximally flat magnitude". It was invented in 1930 by the British engineer and the physicist Stephen Butterworth in his paper entitled " On the Theory of filter Amplifiers". Butterworth had a reputation for solving "impossible" mathematical problems. At the time, filter design required a considerable amount of designer experience due to limitations of the theory then in use. The filter was not in common use for over 30 years after its publication. Butterworth stated that:

"An ideal electrical filter should not only completely reject the unwanted frequencies but should also have uniform sensitivity for the wanted frequencies. "

Such an ideal filter cannot be achieved, but Butterworth showed that succesively closer approximations were obtained with increasing numbers of filters elements of the right values. As for band-pass, a band-pass filter is a device that passes frequencies within a certain range and rejects (attenuates) frequencies outside that range.

A circuit determines which frequencies are going to pass at the output and which are not. It also determines these frequencies' attenuation. This circuit response is called the circuit's frequency response at the various input frequencies. In short, the way an object responds to sounds of different pitches is called its frequency response.

A flat response reproduces the input more accurately through the output without any improvements in a given area. The Butterworth filter is designed to have a frequency response as flat as possible in the passband. For an audio system the goal may be to reproduce the input signal without distortion. This would require a uniform (flat) response magnitude up to the system's bandwidth limit with the signal being delayed at all frequencies by exactly the same amount of time.

Unlike a low-pass filter where only signals with a low frequency range are allowed to pass

or, a high-pass filter, where only signals with a higher frequency range are allowed to pass, the band-pass filter is designed to accept signals within a certain frequency "band" without distorting the input signal or introducing extra noise.

For a band-pass filter, the upper and lower frequency value can be found by:

$$f_C = \frac{1}{2\pi RC} Hz,$$

where R is the resistance in ohm (Ω) and C is the capacitance in farad (F).

The complexity or filter type is defined by the filter's "order", which is dependent upon the number of reactive components such as capacitors or inductors within its design.

The frequency response of a filter can be defined mathematically by its transfer function, hence, the general equation for a Butterworth filter's frequency response is given by the following:

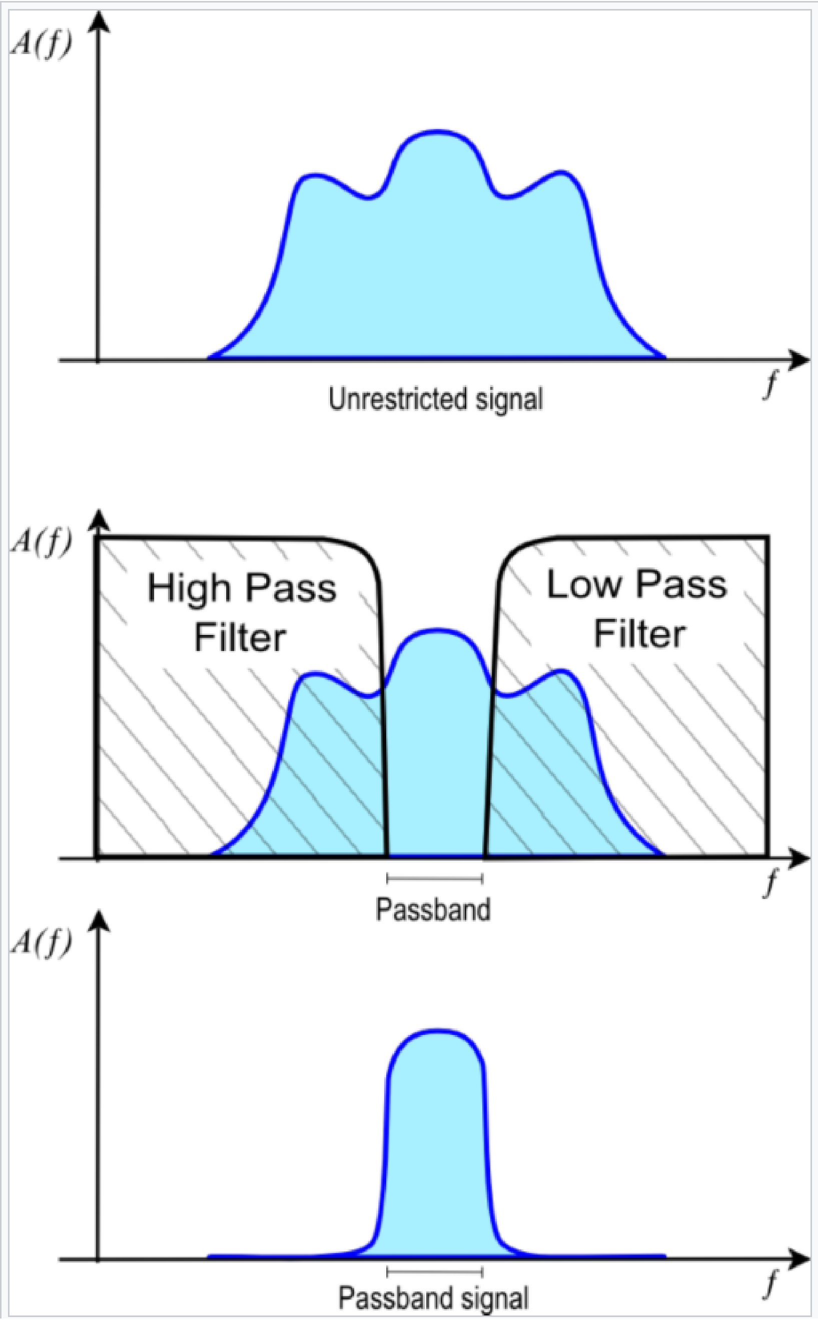

FIGURE 2.1: Unrestricted signal (upper diagram). Bandpass filter applied to signal (middle diagram). Resulting passband signal (bottom diagram). $A(f)$ is the frequency function of the signal or filter in arbitrary units.
Source: Wikipedia

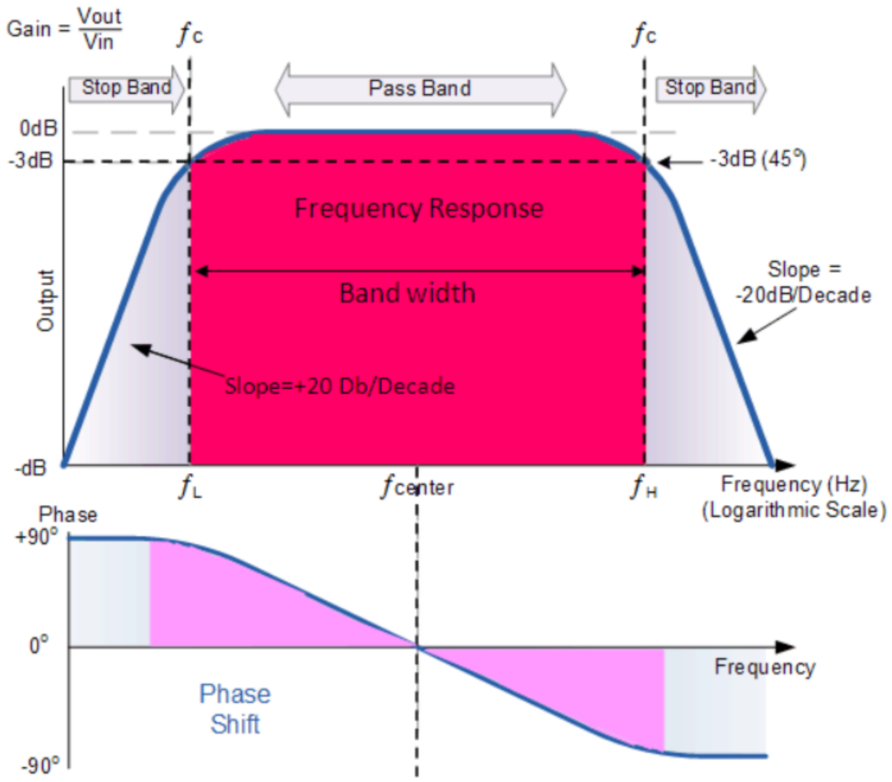

FIGURE 2.2: Frequency response (up) and phase shift (bottom) of a band-pass
Source: Electronicspot

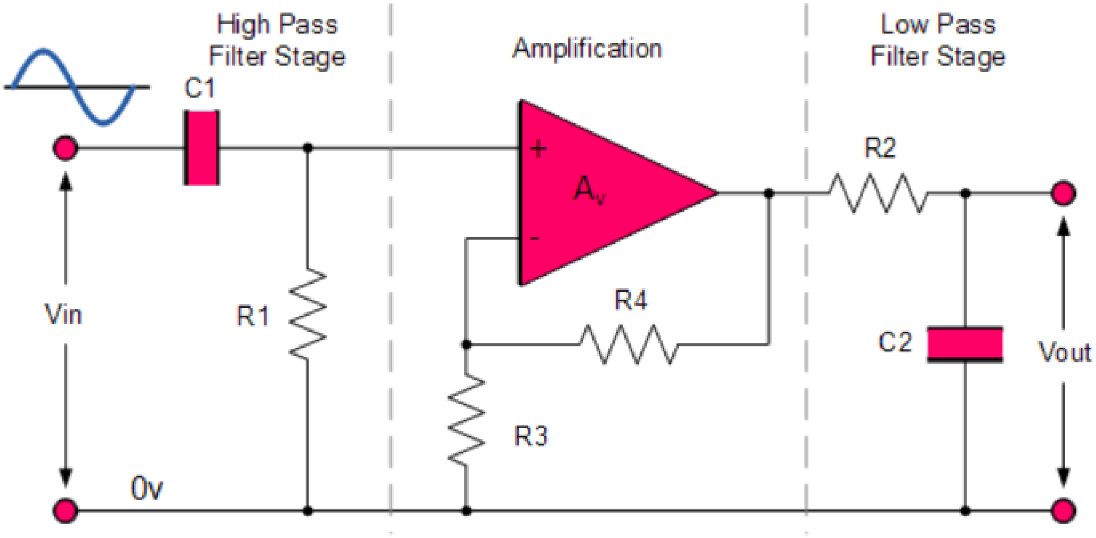

FIGURE 2.3: Band-pass filter circuit
Source: Electronicspot

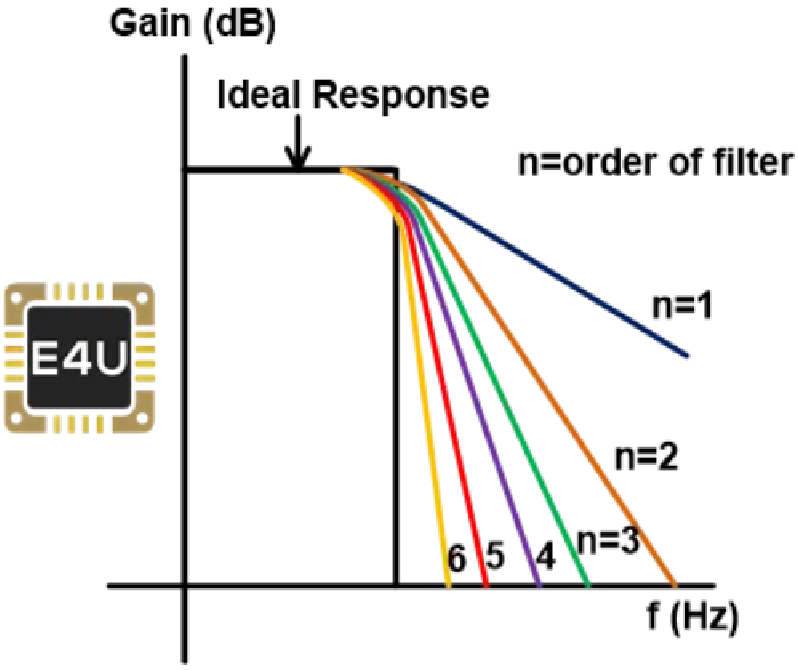

FIGURE 2.4: Frequency Response for a Butterworth Filter
Source: Electrical 4 U

$$|H_{i\omega}| = \frac{1}{\sqrt{1+\epsilon^2(\frac{\omega}{\omega_c})^{2n}}}$$

where n indicates the filter order, $\omega = 2\pi f$, ϵ is maximum pass band gain, (Amax). If we define Amax at cut-off frequency -3dB corner point (f_C), then ϵ will be equal to one and thus ϵ^2 will also be equal to one. But, if we want to define Amax at another voltage gain value, consider 1dB, or $1.1220(1dB = 20\log(Amax))$ then the value of ϵ can be found by:

$$H_1 = \frac{H_0}{\sqrt{1+\epsilon^2}},$$

where H_0 represents the maximum pass band gain and H_1 represents the minimum pass band gain. Now, if we transpose the above equation, then we will get

$$\frac{H_0}{H_1} = 1.1220 = \sqrt{1+\epsilon^2} \qquad \Rightarrow \qquad \epsilon = 0.5088.$$

Remark. *As we have mentioned the frequency response of a filter can be defined mathematically by its transfer function with the Voltage Transfer function the $H(i\omega)$ written as*

$$H_{(i\omega)} = \left[\frac{V_{out}(i\cdot\omega)}{V_{in}(i\cdot\omega)}\right]$$

where V_{out} = *voltage of output signal,* V_{in} = *input voltage signal,* $\omega = 2\pi f$ *is the radian frequency.*

After the implementation of the Butterworth band-pass filter on the signals we extract four different features. These features are described in the following sections. We start by the homomorphic envelogram.

3 Homomorphic Envelogram

Homomorphic filtering is a generalized technique for signal and image processing, involving a nonlinear mapping to a different domain in which filtering techniques are applied, followed by mapping back to the original domain. In the present approach, homomorphic filtering is used to extract a smooth envelogram, which enables the detection of events that are suspected to be S_1, S_2 or others. The advantage of such a homomorphic envelogram is its scalable smoothness, which handles a wide range of problems such as peak splits. Peak splitting is when a Gaussian peak gets a shoulder (see Figure 2.5) or a twin . They have the same base, are unexpected and can be caused by a number of factors.

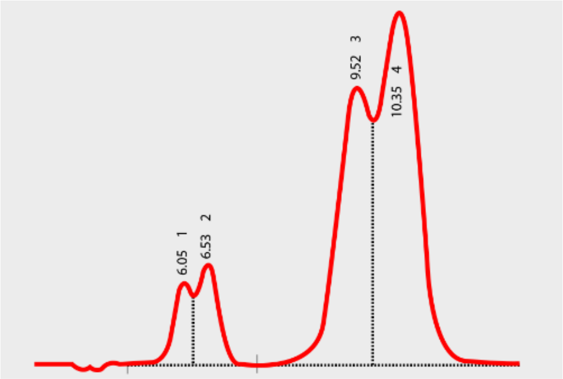

FIGURE 2.5: Each one of Gaussian peaks gets a shoulder.

A monocomponent AM-FM signal can be expressed as a product of its amplitude modulation (AM) and frequency modulation (FM) components:

$$x(t) = a(t)f(t), \qquad a(t) > 0, \tag{2.1}$$

where $a(t)$ is the AM component or instantaneous amplitude (IA) and $f(t)$ is the FM component and can be expressed as:

$$f(t) = \sin(\phi(t)), \tag{2.2}$$

where $\phi(t)$ corresponds to the instantaneous phase.

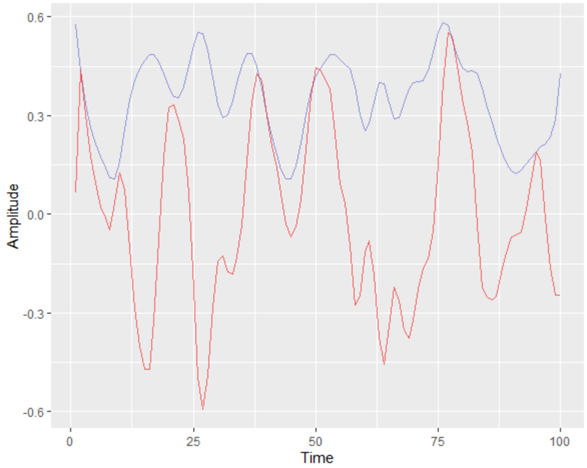

FIGURE 2.6: A phonocardiogram(red) and its Homomorphic Envelogram(blue).

By assuming a simple approximated model by which the PCG is a narrow-band non-stationary signal, we can express it as a monocomponent AM-FM signal (see [46]).We denote:

$$p(t) = \log |x(t)| \tag{2.3}$$

Remark. *In cases where* $x(t) = 0$ *we add a small positive value, and then we have* $p(t) = \log(a(t)) + \log |f(t)|$

By applying an appropriate linear low-pass filter on $p(t)$ we can eliminate the FM component, which is characterized by rapidly variations in time. We denote the low pass filter system by $\mathcal{L}$ and the filtered signal by $p_l(t)$. Because $\mathcal{L}$ is a linear system we obtain:

$$p_l(t) = \mathcal{L}(p(t)) = \mathcal{L}(\log |x(t)|) = \mathcal{L}(\log a(t)) + \mathcal{L}(\log |f(t)|) \tag{2.4}$$

By using a low-pass filter whose pass-band covers the typical frequencies of the AM component and attenuates the typical high frequencies of the FM component, we obtain:

$$p_l(t) = \mathcal{L}(\log a(t)) + \mathcal{L}(\log |f(t)|) = \log a(t) \tag{2.5}$$

The reversal procedure is done by an exponential operation and derives the Homomorphic Envelogram:

$$HomomorphicEnvelogram = \exp\{\mathcal{L}(p(t))\} = \exp\{\log a(t)\} = a(t) \tag{2.6}$$

In this study, the linear low-pass filter $\mathcal{L}$ that is used is a first order Butterworth filter cutoff frequency at 8 Hz.

4 Hilbert Transform

Let us first present the Cauchy principal value P[45]:

Definition 2.1. *Let $[\alpha,\beta]$ be a real interval and let f be a complex-valued function defined on $[\alpha,\beta]$. If f is unbounded near an interior point ξ of $[\alpha,\beta]$, the integral of f over $[\alpha,\beta]$, the integral of f over $[\alpha,\beta]$ does not always exist. However, the two limits*

$$\lim_{\epsilon\to 0}\int_{\alpha}^{\xi-\epsilon} f(x)dx \qquad \textit{and} \qquad \lim_{\epsilon\to 0}\int_{\xi+\epsilon}^{\beta} f(x)dx$$

still may exist, and if they do their sum is called the improper integral of f over $[\alpha,\beta]$ and is denoted by the ordinary integration symbol

$$\int_{\alpha}^{\beta} f(x)dx$$

Even if these two limits do not exist, it may happen that the "symmetric limit"

$$\lim_{\epsilon\to 0^+}\left(\int_{\alpha}^{\xi-\epsilon} f(x)dx+\int_{\xi+\epsilon}^{\beta} f(x)dx\right)$$

exists and if it does, it is called the principal value integral of f from α to β and is denoted by the symbol

$$P\int_{\alpha}^{\beta} f(x)dx.$$

Consequently we present the Hilbert Transform. Many of the common integral transforms can be written in the following form:

$$g(x)=\int_{a}^{b} k(x,y)f(y)dy, \tag{2.7}$$

k(x,y) is called the kernel function, or just the kernel of the equation.

Definition 2.2. *The Hilbert transform on $\mathbb{R}$, the real line, is defined by*

$$Hf(x)=\frac{1}{\pi}P\int_{-\infty}^{+\infty}\frac{f(y)}{x-y}dy, \qquad x\in\mathbb{R}, \tag{2.8}$$

where P is the Cauchy principal value.

The kernel function in this definition is given by

$$k(x,y)=\frac{1}{\pi(x-y)} \tag{2.9}$$

Example 2.4

i) If $f(x) = cos(\omega \cdot x)$, then

$$\begin{aligned}
Hf(\cos(\omega x)) &= -\frac{1}{\pi} P \int_{-\infty}^{+\infty} \frac{\cos(\omega \cdot y)}{y-x} dy \\
&= -\frac{1}{\pi} P \int_{-\infty}^{+\infty} \frac{\cos[\omega \cdot (z+x)]}{z} dz \\
&= -\frac{1}{\pi} \left\{ \cos(\omega \cdot x) P \int_{-\infty}^{+\infty} \frac{\cos(\omega \cdot z)}{z} dz - \sin(\omega \cdot x) P \int_{-\infty}^{+\infty} \frac{\sin(\omega \cdot z)}{z} dz \right\} \\
&= \sin(\omega \cdot z).
\end{aligned}$$

The result is due to the fact $\frac{cos(\omega \cdot z)}{z}$ is an odd function and $P \int_{-\infty}^{+\infty} \frac{\sin(\omega \cdot z)}{z} dz = \pi$.

ii) If $f(x) = p_a(x)$ then

$$\begin{aligned}
Hp_\alpha(x) &= \frac{-1}{\pi} P \int_{-\alpha}^{x-\epsilon} \frac{dy}{y-x} - \frac{1}{\pi x} P \int_{x+\epsilon}^{\alpha} \frac{dy}{y-x} \\
&= \lim_{\epsilon \to 0} \left\{ -\frac{1}{\pi} \log(y-x)|_{-\alpha}^{x-\epsilon} - \frac{1}{\pi} \log(y-x)|_{x+\epsilon}^{\alpha} \right\} \\
&= \frac{1}{\pi} \log \left| \frac{t+\alpha}{t-\alpha} \right| .
\end{aligned}$$

iii) If $f(x) = \alpha$, then

$$\alpha Hf(1) = \alpha \lim_{\alpha \to \infty} \frac{1}{\pi} \log \left| \frac{t+\alpha}{t-\alpha} \right| = 0$$

Hence, if $f_\alpha =$ constant is the mean value of a function, then $f(x) = f_0 + f_1(x)$. Therefore $H\{f_0 + f_1(x)\} = H\{f_1(x)\}$. This implies that the Hilbert transform cancels the mean value or the DC term in electrical engineering terminology.

4.1 The Hilbert transform of functions in L^1

An obvious omission from the discussion of the previous section and earlier parts of the book is the case of Hilbert transforms for functions that belong to $L^1(\mathbb{R})$. The reader is reminded, following the standard custom, that L^1 is abbreviated to L. Some specialized results are now considered for this class of functions.

So if f and $g = Hf$ and $f, g \in L(\mathbb{R})$, let's define the Hilbert Transform pair.

We have defined the Hilbert transform by the principal value integral:

$$Hf(x) = \frac{1}{\pi} P \int_{-\infty}^{\infty} \frac{f(y)}{x-y} dy \tag{2.10}$$

This integral is often written in the following form:

$$Hf(x) = \lim_{\epsilon \to 0^+} H_\epsilon f(x), \tag{2.11}$$

where

$$H_\epsilon f(x) = \frac{1}{\pi} \int_{|x-y|>\epsilon} \frac{f(y)}{x-y} dy \tag{2.12}$$

The function $H_\epsilon f$ is sometimes referred to as the "truncated" Hilbert transform of f. The designation "truncated" is also used to describe other variants of the standard Hilbert transform. Let

$$g(x) = \frac{1}{\pi} P \int_{-\infty}^{\infty} \frac{f(y)}{x-y} dy : \tag{2.13}$$

then the function f is connected to g by the following result

$$f(x) = -\frac{1}{\pi} P \int_{-\infty}^{\infty} \frac{g(y)}{x-y} dy \tag{2.14}$$

Equation (2.13) and (2.14) constitute a Hilbert transform pair.

Remark. *For the case of functions in $L^p(\mathbb{R})$, $p > 1$, it is only necessary to assume that one of the functions f or g belongs to L^p, in contrast to the requirement just stated that both functions $\in L(\mathbb{R})$. For $p > 1$, $f \in L^p(\mathbb{R})$ implies $Hf(x) \in L^p$ however, for $f \in L(\mathbb{R})$, $Hf(x)$ in general does not belong to $L(\mathbb{R})$. Consider the case $f(x) = \alpha(\alpha^2 + x^2)$ for $\alpha > 0$. Now $f \in L(\mathbb{R})$ and*

$$g(x) = Hf(x) = \frac{x}{\alpha^2 + x^2} \tag{2.15}$$

which does not belong to $L(\mathbb{R})$.

Kober (1942) gave the following result. If $f \in L(\mathbb{R})$ a necessary condition that $Hf \in L(\mathbb{R})$ is

$$\int_{-\infty}^{+\infty} f(x) dx = 0. \tag{2.16}$$

That this condition is not sufficient is attributed by Kober to H.R.Pitt. The latter result can be established as follows. Let

$$f(x) = \begin{cases} 0\,, & -\infty < x \leq 0 \\ x^{-1} \log^{-2}(x) - \frac{2}{log2}\,, & 0 < x < \frac{1}{2} \\ 0\,, & \frac{1}{2} \leq x < \infty \end{cases} \tag{2.17}$$

Using the change of variable $x = e^{-y}$ (or otherwise noting that the integrand is an exact differential) leads to

$$\int_{-\infty}^{+\infty} f(x)dx = \int_0^{\frac{1}{2}} x^{-1}\log^{-2}(x)dx - \frac{1}{\log 2} = -\frac{1}{\log 2} + \int_{\log 2}^{\infty} \frac{dy}{y^2} = 0. \tag{2.18}$$

so Equation (2.7) is satisfied. For $p > 1$, the integral $\int_{-\infty}^{+\infty} |f(x)|^p dx$ diverges. Now $f \in L(\mathbb{R})$ since

$$\begin{aligned}\int_{-\infty}^{+\infty} |f(x)|dx &= \int_0^{\frac{1}{2}} \left| x^{-1}\log^{-2}(x) - \frac{2}{\log 2}\right| dx \\ &= \frac{2}{\log x_2} - \frac{2}{\log x_1} + \frac{4(x_2 - x_1)}{\log 2} = 0\end{aligned} \tag{2.19}$$

where $x_1 \approx 0.026042$ and $x_2 \approx 0.389208$ are solutions of $x^{-1}\log^{-2}(x) - \frac{2}{\log 2} = 0$

Let $g(x) = Hf(x)$; for $x > 0$ it follows that

$$\begin{aligned}-g(-x) &= \frac{1}{\pi}P\int_{-\infty}^{+\infty} \frac{f(t)}{x+t}dt \\ &= \frac{1}{\pi}P\int_0^{\frac{1}{2}} \frac{dt}{(x+t)t\log^2 t} - \frac{2}{\pi\log 2}\int_0^{\frac{1}{2}} \frac{dt}{(x+t)}\end{aligned} \tag{2.20}$$

and, since $(x+t)^{-1} > (2x)^{-1}$ for $t \in (0, x)$,

$$\begin{aligned}-g(-x) &> \frac{1}{2\pi x}\int_0^x \frac{dt}{t\log^2 t} - \frac{2}{\pi\log 2}\log\left(\frac{2x+1}{2x}\right) \\ &= -\frac{1}{2\pi x}\int_0^x d[logt]^{-1} - \frac{2}{\pi\log 2}\log\left(\frac{2x+1}{2x}\right).\end{aligned} \tag{2.21}$$

The second contribution in the final result is not important for the argument that follows, so this term is dropped. Hence, for $x \in (0, \frac{1}{2})$,

$$-g(-x) > -\frac{1}{2\pi x\log x} \tag{2.22}$$

Now,

$$\int_{-\frac{1}{2}}^{0} |g(x)|dx = \int_0^{\frac{1}{2}} |-g(-x)|dx > \frac{1}{2\pi}\int_0^{\frac{1}{2}} \frac{1}{x\log x}dx = \infty; \tag{2.23}$$

that is, $Hf(x) \notin L(-\frac{1}{2}, 0)$, and since

$$\int_{-\infty}^{\infty} |Hf(x)|dx = \int_{-\infty}^{-\frac{1}{2}} |g(x)|dx + \int_{-\frac{1}{2}}^{0} |g(x)|dx + \int_0^{\infty} |g(x)|dx \tag{2.24}$$

then $Hf \notin L(\mathbb{R})$, which proves that Equation (2.16) is not a sufficient condition.

To establish Equation (2.25), suppose $f \in L(\mathbb{R})$, $Hf \in L(\mathbb{R})$, and

$$F(x) = f(x) + iHf(x) \tag{2.25}$$

Now we define the Fourier transform:

Definition 2.3. *Suppose f is an absolutely integrable function on $\mathbb{R}$, that is $\int_{-\infty}^{+\infty} |f(x)|dx < \infty$; then the Fourier transform of f, denoted by $\mathcal{F}f$, is defined by*

$$\mathcal{F}f(x) = \int_{-\infty}^{+\infty} f(s)e^{-ixs}ds.$$

Taking the Fourier transform of Equation (2.25)

$$\mathcal{F}F(x) = \mathcal{F}f(x) + \mathcal{F}iHf(x) = (1 + sgn(x))\mathcal{F}f(x), \tag{2.26}$$

where the second equality follows from the fact that $\mathcal{F}Hf(x) = i \cdot sgn(x)\mathcal{F}f(x)$ and $sgn(x)$ stands from the signum function.

Following we define the Cauchy integral theorem.

Definition 2.4. *if $f(z)$ is holomorphic in a simply connected domain Ω, then for any simply closed contour C in Ω, tha contour integral is zero*

$$\int_C f(z)dz = 0. \tag{2.27}$$

The function $F(z)$ is analytic in the upper half complex plane, and, by the definition (2.4),

$$\oint_C F(z)dz = 0 \tag{2.28}$$

where the contour C is a semicircle in the upper half plane centered at the origin and including the real axis. From Equation (2.18) it follows that

$$\int_{-\infty}^{\infty} f(x)dx = 0 \tag{2.29}$$

and hence

$$\mathcal{F}F(0) = 0 \tag{2.30}$$

From Equation (2.26) it follows that

$$\mathcal{F}f(0) = 0 \tag{2.31}$$

and this establishes Equation (2.16).

Moreover it is important to present some properties of the Hilbert transform.

4.2 Linearity

An important property of the Hilbert transform operator is that it is a linear operator. A linear operator L is a mapping from a vector space X into a vector space Y written $L : X \to Y$, such that for constants $\alpha, \beta \in X$, and function $f, g \in X$, then

$$L\{\alpha f + \beta g\} = \alpha L f + \beta L g.$$

For constants $\alpha, \beta \in \mathbb{C}$ and functions f and g, it follows that

$$H\{\alpha f + \beta g\} = \alpha L f + \beta L g.$$

In the preceding result, the separate integrals are assumed to exist, and this is true if the functions belong to the class $L^p(\mathbb{R})$ for $1 \leq p$.

The latter condition can be replaced by one where the functions satisfy a suitable asymptotic constraint as $|x| \to \pm\infty$ and are uniformly Holder continuous on every finite interval of $\mathbb{R}$.

4.3 Inversion property

Since $Hf(x) = g(x)$ implies $Hg(x) = -f(x)$, then

$$H^2 f(x) = H(Hf)(x) = -f(x). \tag{2.32}$$

This is referred to as the **inversion formula** for the Hilbert transform, and it is also called the iteration property for the Hilbert transform. An approach to obtain Equation (2.32) makes use of the Hardy-Poincare-Bertrand formula, which takes the following form:

$$\frac{1}{\pi}\int_{-\infty}^{\infty} \frac{\phi_1(x)}{x-t}dx \cdot \frac{1}{\pi}\int_{-\infty}^{\infty} \frac{\phi_2(x)}{y-x}dy = \frac{1}{\pi}P\int_{-\infty}^{\infty} \phi_2(y)dy \cdot P\int_{-\infty}^{\infty} \frac{\phi_1(x)}{(x-t)(y-x)}dx - \phi_1(t)\cdot\phi_2(t), \tag{2.33}$$

where $\phi_1(x)$ and $\phi_2(x)$ belong to the classes L^p and L^q, respectively, with the exponents satisfying $1 < p < \infty, 1 < q < \infty$, and $p^{-1} + q^{-1} = 1$. Let $\phi_1(x) = e^{-\alpha x^2}$ with $a > 0$, and set $\phi_2(y) = f(y)$. The function $\phi_1(x)$ is going to be treated as a convergence factor. From Equation (2.33) it follows that

$$\frac{1}{\pi}P\int_{-\infty}^{\infty}\frac{e^{-\alpha x^2}}{x-t}dx\cdot\frac{1}{\pi}P\int_{-\infty}^{\infty}\frac{f(y)}{y-x}dy=\frac{1}{\pi}\int_{-\infty}^{\infty}f(y)dy=\frac{1}{\pi}\cdot P\int_{-\infty}^{\infty}\frac{e^{-\alpha x^2}}{(x-t)(y-x)}dx-e^{-\alpha t^2}f(t). \tag{2.34}$$

Now,

$$P\int_{-\infty}^{\infty}\frac{e^{-\alpha x^2}}{(x-t)(y-x)}=\frac{1}{y-t}P\int_{-\infty}^{\infty}\left\{\frac{e^{-\alpha x^2}}{x-t}+\frac{e^{-\alpha x^2}}{y-x}\right\}dx \tag{2.35}$$

If the $\lim \alpha \to 0^+$ is examined, then the last integral evaluates to zero, and Equation (2.34) becomes

$$\frac{1}{\pi}\int_{-\infty}^{\infty}\frac{dx}{t-x}\frac{1}{\pi}P\int_{-\infty}^{\infty}\frac{f(y)}{x-y}dy=-f(t), \tag{2.36}$$

or, in compact notation,

$$HHf(t)=-f(t), \tag{2.37}$$

which is the desired result. The reader is invited to examine critically the validity of taking $\lim \alpha \to 0^+$ inside the integral in the preceding sequence of steps.

The obvious extension of Equation (2.32) becomes for non-negative integer n,

$$H^n f(x)=\begin{cases}(-1)^{\frac{n}{2}}f(x),\ for\ n\ even\\ \\ (-1)^{\frac{(n-1)}{2}}g(x),\ for\ n\ odd\end{cases} \tag{2.38}$$

This can be proved by repeated application of Equation (2.32).

From Equation (2.32) the operator equivalence can be written as follows:

$$H^2=-I \tag{2.39}$$

where I denotes the identity operator. From this result the inverse Hilbert transform operator can be written symbolically as

$$H^{-1}=-H \tag{2.40}$$

and so

$$H^{-1}(Hf)(x)=f(x)=-\frac{1}{\pi}P\int_{-\infty}^{\infty}\frac{Hf(t)}{x-t}dt \tag{2.41}$$

4.4 Derivatives of the Hilbert transform

Theorem 1.1 The Hilbert transform of the derivative of a function is equivalent to the derivative of the Hilbert transform of a function, that is

$$Hf'(t) = \frac{d}{dt}\hat{f}(t) \tag{2.42}$$

Proof We know that

$$\hat{f}(t) = \frac{1}{\pi} P \int_{-\infty}^{+\infty} \frac{f(t)}{t-s} ds$$

if we substitute s with t-z, then

$$\hat{f}(t) = \frac{1}{\pi} P \int_{-\infty}^{+\infty} \frac{f(t-z)}{z}$$

and then apply the derivative of t on both sides we get

$$\frac{d}{dt} f(\hat{t}) = \frac{1}{\pi} P \int_{-\infty}^{+\infty} \frac{f'(t-z)}{z} dz$$

The substitution z=t-s gives us that

$$\frac{d}{dt}\hat{f}(t) = \frac{1}{\pi} P \int_{-\infty}^{+\infty} \frac{f'(s)}{t-s} ds$$

and th relation in (3.2) is valid.

From the proof above we conclude that the relation can be used repeatedly. Let us look an example where we also make use of multiple Hilbert transforms(see Section 4.2).

Example 4.3 By (4.2) we may calculate the Hilbert transform of the delta function $\delta(t)$ and its derivatives. At the same time we get the Hilbert transform representation of the delta function. Consider the Hilbert transform of the data function.

$$H\delta(t) = \frac{1}{\pi t}$$

The derivative of the delta function is calculated to

$$H\delta'(t) = -\frac{1}{\pi t^2},$$

and if we apply the Hilbert transform on both sides then we get

$$\delta'(t) = H(-\frac{2}{\pi t^3})$$

This procedure can be continued.

4.5 Orthogonality properties

Definition 2.5. *A complex function is called Hermitian if its real part is even and its imaginary part is odd. From this we have that the Fourier transform $F(\omega)$ of a real function f(t) is Hermitian.*

Theorem 4.5 A real function f(t) and its Hilbert transform $\hat{f}(t)$ are orthogonal if f, $\hat{f}$ and F belong to $L^1(\mathbb{R})$ or if u and $\hat{f}$ belong to $L^2(\mathbb{R})$.

Theorem 4.6 If f,g and G belong to $L^1(\mathbb{R})$ or if f and g belong to $L^2(\mathbb{R})$ then

$$\int_{-\infty}^{+\infty} f(t)g_*(t)dt = \frac{1}{2\pi}\int_{-\infty}^{+\infty} F(\omega)G^*(\omega)d\omega$$

Proof From Theorem 4.6 we have that

$$\begin{aligned}\int_{-\infty}^{+\infty} f(t)\hat{f}(t)dt &= \frac{1}{2\pi}\int_{-\infty}^{+\infty} F(\omega)(-isgn(\omega)F(\omega))^*d\omega \\ &= \frac{i}{2\pi}\int_{-\infty}^{+\infty} sgn(\omega)F(\omega)F^*(\omega)d\omega \\ &= \frac{i}{2\pi}\int_{-\infty}^{+\infty} sgn(\omega)|F(\omega)^2|d\omega\end{aligned}$$

where $sgn(\omega)$ is an odd function and the fact that $F(\omega)$ is Hermitian gives us that $|F(\omega)|^2$ is an even function. We conclude that

$$\int_{-\infty}^{+\infty} f(t)\hat{f}(t)dt = 0$$

and a real function and its Hilbert transform are orthogonal.

4.6 Energy aspects of the Hilbert transform

The energy of a function f(t) is closely related to the energy of its Fourier transform $F(\omega)$. Theorem 4.6 with $f(t) = g(t)$ is called the Rayleigh theorem and it helps us to define the energy of f(t) and $F(\omega)$ as

$$E_u = \int_{-\infty}^{+\infty} |f(t)|^2 dt = \frac{1}{2\pi}\int_{-\infty}^{+\infty} |F(\omega)|^2 dw \tag{2.43}$$

Here it is natural to assume that $f \in L^2(\mathbb{R})$ which means that E_f is finite. The same theorem is used to define the energy of the Hilbert transform of $f(t)$ and $F(\omega)$, that is

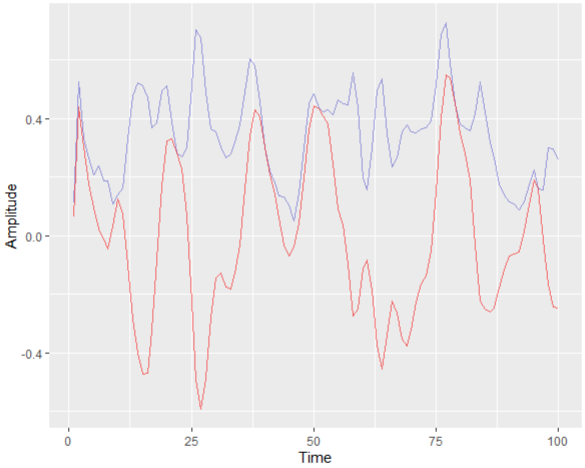

FIGURE 2.7: A phonocardiogram(red) and its Hilbert envelogram(blue).

$$E_{\hat{f}} = \int_{-\infty}^{+\infty} \left|\hat{f}\right|^2 dt = \frac{1}{2\pi} |-isgn(\omega)F(\omega)|^2 d\omega$$

where $|-isgn(\omega)|^2 = 1$ except for $\omega = 0$. But, since $F(\omega)$ does not contain any impulses at the origin we get $E_{\hat{f}} = E_f$.

A consequence of (4.5) is that $f \in L^2(\mathbb{R})$ induces that $\hat{f} \in L^2(\mathbb{R})$. The accuracy of the approximated Hilbert transform operator can be measured by comparing the energy in (2.2) and (4.5). However, a minor difference in energy always exists in real applications due to unavoidable truncation errors.

4.7 Hilbert Envelope

Having seen the definition and having seen some properties of Hilbert Transform, let us calculate the Hilbert Envelope. At first, let's define the analytic signal.

Definition 2.6. *An analytic signal is a complex-valued function that has no negative frequency components. The analytic signal $s(t)$ given by:*

$$s(t) = f(t) + i \cdot h(t)$$

where f(t) is the input signal and h(t) is its Hilbert Transform as described above.

A Hilbert Envelope is then constructed from the absolute value of the analytic signal:

$$HilEnv = |s(t)| = \sqrt{f^2(t) + h^2(t)}$$

5 Wavelet Envelope

A wavelet is a mathematical function used to divide a given function or continuous-time signal into different scale component.Usually one can assign a frequency range to each scale component. Each scale component can then be studied with a resolution that makes its scale. A wavelet transform is the representation of a function by wavelets. The wavelets are scaled and translated copies (known as "daughter wavelet") of a finite-length or fast-design oscillating waveform (known as the "mother wavelet").

As a mathematical tool, wavelets can be used to extract information from many different kinds of data, including- but not limited to- audio signals and images. Sets of wavelets are needed to analyze data fully. "Complementary" wavelets decompose a signal without gaps or overlaps so that the decomposition process is mathematically reversible. Thus, sets of complementary wavelets are useful in wavelet based sets of complementary wavelets are useful in wavelet based compression/decompression algorithms where it is desirable to recover the original information with minimal loss.

It's worthwhile to refer that the wavelet transform is similar to the Fourier Transform (or much more to the windowed Fourier Transform) with a completely different merit function. The wavelet transform is often compared to the Fourier Transform in which signals are represented as a sum of sinusoids. Less distortion to the spectral characteristics of the de-noised images distinguises wavelet transform from other techniques. The main difference between wavelet transform and Fourier Transform is that, in the Wavelet Transform, wavelets are only localized infrequency. The Short-time Fourier Tranform (STFT) is more similar to the wavelet tranform. In this also the wavelets are time and frequency localized but there are issues with frequency/time resolution trade-off. Wavelets often give a better signal representation using. Multi-resolution analysis with balanced resolution at anytime and frequency. While Fourier analysis consists of breaking up the signal into shifted and scaled versions of the original (or mother) wavelet just by analyzing the wavelets and sine waves, we can conclude intuitively that signals with sharp changes might be better analyzed with irregular wavelet than with a smooth sinusoid, just as some foods are better handled with a fork than a spoon.

Also there are many wavelets family, such as:

- **Haar**

  In mathematics, the Haar wavelet is a sequence of rescaled "square-shaped" functions which together form a wavelet family or basis. Wavelet analysis is similar to Fourier Analysis in that it allows a target function over an interval to be represented in terms of an orthogonal basis. The Haar sequence is now recognised as the first known wavelet basis and extensively used as a teaching example.

The Haar wavelet is also the simplest possible wavelet. The technical disadvantage of the Haar is that it is not continuous, and therefore not differentiable. This property can, however, be an advantage for the analysis of signals with sudden transitions (discrete signals), such as monitoring of tool failure in machines.

The Haar wavelet's mother wavelet function $\psi(t)$ can be described as:

$$\psi(t) = \begin{cases} 1 & 0 \leq t < 1/2 \\ -1 & 1/2 \leq x < 1 \\ 0 & otherwise \end{cases}$$

its scaling function $\phi(t)$ can be described as:

$$\phi(t) = \begin{cases} 1 & 0 \leq t < 1 \\ 0 & otherwise \end{cases}$$

- **Daubencies**

  The Daubencies wavelets, based on the work of Ingrid Daubencies, are a family of orthogonal wavelets defining a discrete wavelet transform and characterized by a maximal number of vanishing moments for some given support. With each wavelet type of this class, there is a scaling function (called the father wavelet) which generates an orthogonal multiresolution analysis.

- **Coiflets**

  Coiflets are discrete wavelets designed by Ingrid Daubechies, at the request of Ronald Coifman, to have scaling functions with vanishing moments. The wavelet is near symmetric, their wavelet functions have $N/3$ vanishing moments and scaling functions $N/3 - 1$, and has been used in many application using Calderon-Zygmund operators.

- **Symlet**

  In applied mathematics, symlet wavelets are a family of wavelets. They are a modified version of Daubenchies wavelets with increased symmetry.

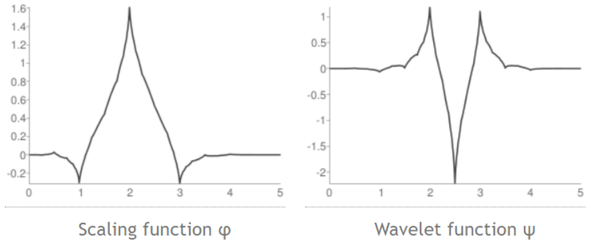

FIGURE 2.8: Wavelet Coiflet 1

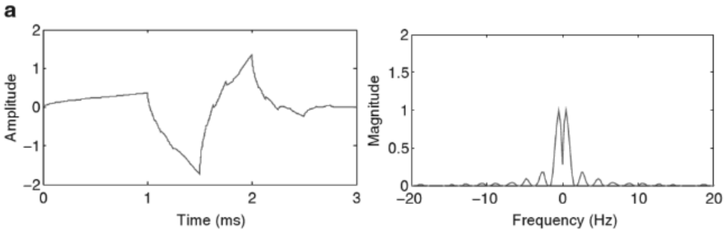

FIGURE 2.9: Symlet 2 base wavelet

6 Power Spectral Density Envelope

Power spectrum is a representation of the magnitude of the various frequency components of a signal. Spectrum simply answer the question "How much power contain each of signal's frequency component? ". The transformation from the time domain signal to the frequency domain and applications in engineering, in communication systems, in statistics made the power spectrum "famous". There are a couple of techniques for generating the Power spectrum. In our case the Power Spectral Density (PSD) was calculated using the Short-time Fourier transform (STFT) after Hamming windowing.

In many field of the research a window function also known as an apodization function is a mathematical function that is zero-valued outside of some chosen interval, normally symmetric around the middle of the interval, usually near a maximum in the middle of the interval, usually tapering away from the middle.

When a window function multiply a data-sequence the product is also zero-valued outside the interval: all that is left is the part where they overlap, the "view through the window".

All that is left is the part where they overlap, that is the segment of data within the window is first isolated, and then only that data is multiplied by the window function values.

Instead of looking at the whole signal, the STFT's main idea is to consider only a small part of the signal. To this end, the original signal is then multiplied by the window function to give a window signal. The window function varies over time to obtain frequency information at different times and a Fourier transform is computed for each resulting window signal. It is very essential to underline that STFT reflects not only the properties of the original signal but also those of the window function.

Let us now determine the most vital mathematical formulas practical applications. Let $x : \mathbb{Z} \to \mathbb{R}$ be a discrete time signal with a fixed sampling frequency Fs in Hertz. Consider $w = \{0, 1, \dots N - 1\} \to \mathbb{R}$ to be a sampled window function of length $N \in \mathbb{N}$. One also introduces an additional parameter $h \in \mathbb{N}$, which is referred to as the hop size. The hop size parameter is specified in samplees and determines the step size in which the window is to be shifted acroos the signal. Now, with regard to these parameters the discrete STFT Y_x of the signal x is given by:

$$Y_x(m,k) = \sum_{n=0}^{N-1} x(n+mh)w(n)exp\{\frac{-2\pi ikn}{N}\}$$

where $m \in \mathbb{Z}$ and $0 \leq k \leq K$. The number $K = \frac{N}{2}$, if we assume that N is even, is the frequency index corresponding to the Nyquist frequency. The length parameter N determines the duration of the considered section, that is $\frac{N}{Fs}$ seconds. The complex number $Y_x(m,k)$ denotes the k^{th} Fourier coefficient for m^{th} time frame. As for the temporal dimension, each Fourier coefficient $Y_x(m,k)$ is associated with the physical time position.

$$T_{coef}(m) = \frac{m \cdot h}{F_s}$$

given in seconds. For instance, for the smallest possible hop size h=1, one obtains $T_{coef}(m) = \frac{m}{Fs} = m \cdot T$.

In this case, one obtains a spectral vector for each sample of the signal x, which results in a huge increase in data volume. Furthermore, considering sections that are only shifted by one sample generally yields very similar spectral vectors. To reduce this type of redundancy, one typically relates the hop size to the length N of the window. For example, one often chooses $h = \frac{N}{2}$, which constitutes a good trade-off between a reasonable temporal resolution and the data volume comprising all generated spectral coefficients. As for the frequency dimension, the index k of $Y_x(m,k)$ corresponds to the physical frequency

$$F_{coef}(k) = \frac{k \cdot F_s}{N}$$

given in Hertz. Actually, the majority of the frequency content of the S_1 and S_2 sounds is below 150Hz, with a peak at 50Hz and concentrated with a high probability ±10 around the peak. Consequently, for this application the final feature was derived from the mean PSD computed only for frequencies from 40 to 60 Hz, found in overlapping windows of 0,05 seconds in width with 50 % overlap. This resulted in an envelope of PSD values. Based on these values and according to the notation described above, if $Fs = 1000Hz$ is the considered sampling frequency in this application we have: N=50, k=25 and h=25. The final envelope was derived from the mean PSD betwwen 40 and 60 Hz, that is according to the above, the mean PSD between k=2 and k=3.

Benefits of Power Spectral Density Profile

- Using the PSD profile, we can identify the frequency components having relatively weaker power levels in the given frequency range of interest.
- The received test signal (control signal) is studied a spectrum analyzer. The PSD profile in studied and compared to the PSD of the test signal before it was transmitted through the channel.
- By studying the PSD profile, we can determine the frequency components with reduced power levels as a result of channel noise. These frequency components have relatively been more vulnerable to the noise when compared to the other frequency components present in the signal.
- We can therefore adjust the power levels of the signal to be transmitted to combat the effect of channel.

7 Heart Labeling

In order to train the segmentation algorithms on the phonocardiogram (PCG) data, it is paramount to label S_1 and S_2 sounds. The positions of the R-peak connected with start of the S_1 sound. So, each R-peak position indicates the start of S_1 sound. The mean of S_1 sound duration is 122ms with a standard deviation of 32ms according to [43], thus due to the low variability the interval that is labeled as S_1 sound is assumed constant and equals with [R-peak,R-peak+122]. As for S_2 sound, S_2 sound does not coincide with the position of the end-T-wave so it is more difficult to label it. However, the amplitude of the S_2 sound reach a maximum in a neighborhood of the end-T-wave. As a result, the center of the S_2 sound was found by searching for the maximum peak in the Hilbert envelope of the PCG signal specifying search window around the end-T-Wave. The mean duration of S_2 is again according to 92ms with a standard deviation of 28ms, consequently the longest expected duration of S_2 is $\max(S_2 \pm \sigma_{S_2}) = \max(92 + 28, 92 - 28) = 120$.

Chapter 3

Statistical model

1 Markov Renewal Chain

A renewal process is an idealized stochastic model for "events" that occur randomly in time (generically called renewals or arrivals). The basic mathematical assumption is that the times between successive arrivals are independent and identically distributed. We will need the renewal process in discrete time, which we refer it as renewal chain. Let us now present the Markov renewal chain. A Markov renewal chain is a random process that generalizes the notion of Markov chains, distinguished from the fact that the holding time of each state is geometrically distributed.

Definition 3.1. *Consider a state space E of a Markov chain. Consider a set of variables (Y_n, S_n), where S_n are the jump times and $\tau_n = S_n - S_{n-1}$ is the inter-arrival times of the states. Then the sequence (Y_n, S_n) is called a Markov renewal process if*

$$Pr(\tau_{n+1} \leq t, Y_{n+1} = j|(Y_0, S_0), (Y_1, S_1), ..., (Y_n = i, S_n)) = Pr(\tau_{n+1} \leq t, Y_{n+1}|Y_n = i)$$

$\forall n \geq 1, t \geq 0, i, j \in S$

As we can see the next state of a Markov renewal process only depends on the current state.

2 Semi-Markov chains

Before we define the Hidden semi-Markov models we have to introduce the semi-Markov chains[8]. First of all, it's worthwhile to refer that a semi-Markov chain is a stochastic process which generalizes a Markov chain in the sense that the "memoryless property". As the well-known memoryless property is still present, not globally, but only at the specific time instants where we have change from one state to another state. Now, we consider a random system with finite state space $E = \{1, \ldots, s\}$, whose evolution in time is governed by a stochastic process $Z = (Z_k)_{k \in \mathbb{N}}$. Let us denote by

$S = (S_n)_{n\in\mathbb{N}}$ the successive time point when state changes in $(Z_k)_{k\in\mathbb{N}}$ occur and $Y = (Y_n)_{n\in\mathbb{N}}$ the successively visited states at these time points. Set also $X = (X_n)_{n\in\mathbb{N}^*}$ for the successive sojourn time in the visited states, thus $X_n = S_n - S_{n-1}, n \in \mathbb{N}^*$. The relation between process Z and process Y of the successively visited states is given by $Z_k = Y_{N(k)}$ where,

$N(k) = \max\{n \in \mathbb{N} | S_n \leq k\}$ is the discrete-time counting process of the number of jumps in $[1, k] \subset \mathbb{N}$.

So we have

$$Z_k = Y_{N(k)}, k \in [S_{n-1}, S_n)$$

where $Y_{N(k)}$ is assumed to be a Markov chain and X_n is given by $X = 1 + \sum_{k=1}^{n} S_k$ we start the discrete-time process from $t = 1$ instead of $t = 0$ in order to comply with the formalism of the application context of this thesis. The process $(Z_k)_{k\geq 1}$ is called **discrete-time semi-Markov chain** and this is the type of semi-Markov chain that we deal with in this thesis.

Definition 3.2. *A matrix-valued function $q = (q_{ij}(u)) \in \mathcal{M}_E(\mathbb{N})$ is said to be a discrete-time semi-Markov kernel if it satisfies the following three properties:*

1. $0 \leq q_{ij}(u),\ i, j \in E, u \in \mathbb{N}$
2. $q_{ij}(0) = 0,\ i, j \in E$
3. $\sum_{u\in\mathbb{N}} \sum_{j\in E} q_{ij}(u) = 1, i \in E$

Definition 3.3. *(Markov renewal chain)*
The chain (Y_n, S_n) is a Markov renewal chain if $\forall n \in \mathbb{N}$, $\forall i, j \in E$ and $\forall k \in \mathbb{N}$, the following equation is almost surely satisfied :

$$P(Y_{n+1} = j, S_{n+1} - S_n = u | Y_1, Y_2, \ldots, Y_n, S_1, ..., S_n) = P(Y_{n+1} = j, S_{n+1} - S_n = u | Y_n)$$

In the case, where the above equation is independent of n, then (Y_n, S_n) is homogeneous with a discrete-time semi-Markov kernel q, which is defined by

$$q_{ij}(u) = P(Y_{n+1} = j, X_{n+1} = u | Y_n = i)$$

where $X_{n+1} = S_{n+1} - S_n$.

As for the dynamics of z_k are driven by the semi-Markov kernel q above. Let us assume that the system enters in state i at time t_1 and remains there until t_2 where $(t_1 \leq t_2)$ and leaves for state j,

$(i, j \in S)$. Then $Z_{t_1-1} \neq i$, $Z_{t_2} = i$, $Z_{t_2+1} = j$. In fact, the semi-Markov kernel plays in semi-Markov chains the role that Markov transition matrix plays in Markov chains.

Remark. *If (Y, S) is a (homogeneous) Markov renewal chain, we can see that $(Y_n)_{n\in\mathbb{N}}$ is (homogeneous) Markov chain. The transition matrix of Y_n, $p = (p_{ij})_{i,j\in E} \in \mathcal{M}_E$ is defined by*

$$p_{ij} = P(Y_{n+1} = j|Y_n = i), i, j \in S, n \in \mathbb{N}$$

Remark. *Another way to express p_{ij} is $p_{ij} = \sum_{u\in\mathbb{N}} q_{ij}(u)$ where we used the semi-Markov kernel.*

Remark. *The process $(Y_n, X_n)_{n\geq 1}$ is also a Markov chain with transition probabilities*

$$\begin{aligned} p_{(i,u')(j,u)} &= P(Y_{n+1} = j, X_{n+1} = u|Y_n = i, X_n = u') \\ &= P(Y_{n+1} = j, X_{n+1} = u|Y_n = i) \\ &= q_{ij}(u) \end{aligned}$$

There is no dependence on u'.

The researchers who used Markov Renewal chains, in order to analyze data and to make estimation, are interested in two types of holding time distributions in a given state and the conditional distributions depending on the next state to be visited.

Definition 3.4. $\forall i, j \in E$

1. *$f_{ij}()$, the conditional distribution of X_{n+1}, $n \in \mathbb{N}$*

$$f_{ij}(u) = P(X_{n+1}|Z_n = i, Z_{n+1} = j), u \in \mathbb{N}$$

2. *$F_{ij}()$, the conditional cumulative distribution of X_{n+1}, $n \in \mathbb{N}$*

$$F_{ij}(u) = P(X_{n+1} \leq k|Z_n = i, Z_{n+1} = j) = \sum_{l=0}^{u} f(l), u \in \mathbb{N}$$

Definition 3.5. *The sojourn time distribution in state i is defined as:*

$$d_i(u) = P(X_{n+1} = u|Y_n = i) = \sum_{j\in E} q_{ij}(u), u \in \mathbb{N}$$

Remark. *In this work we assume that*

$$f_{ij}(u) = d_i(u) \qquad \forall j \in E$$

Now we can go ahead and present the Hidden semi-Markov model.

3 Hidden semi-Markov model

In statistics, when we have a set of data, we want to make inference and predictions about them and that we can do it choosing "useful" statistical models. In our case we have chosen the Hidden semi-Markov models[1][35]. Hidden Markov Models (HMMs) are a class of models in which the distribution that generates an observation depends on the state of a underlying but unobserved Markov process.It's worthwhile to refer that HMMs have been applied in the many fields like as signal-processing, speech recognition and many others. In addition, a Hidden Markov Model is a statistical model can be used to describe the evolution of observable events that depend on internal factors, which are not directly observable. In the simplest case, the underlying stochastic process is a discrete-time finite-state homogeneous Markov chain, which influences another stochastic process that produces a sequence of observations. However, the fact that the non-zero probability of self-transition of a non-absorbing state, the state duration of an HMM is implicity a geometric distribution. This makes the HMM has limitation in some application like as in our. As a result have been used a Hidden semi-Markov model (HSMM). A HSMM is an extension of HMM. The main difference between them is the fact that in the HMMs the underlying sequence is a Markov chain while in the HSMMs the underlying sequence is a Semi-Markov chain as described above. Moreover, the state durations in HSMMs are not geometrically distributed. Hence, in HSMMs each state has a variable duration, which is associated with the number of observations produced while the Markov process is in specific state.

Definition 3.6. *A Hidden semi-Markov Model is a doubly discrete time stochastic process $(Z_k, Y_k)_{k\geq 1}$ where*

- (Z_k) *is an unobservable Hidden semi-Markov chain and*
- (Y_k) *is an observable sequence of conditionally independent random variables such that the conditional distribution of* Y_k *given* Z_k *depends only on* Z_k.

The output process Y_k is related to the semi-Markov chain Z_k by the observation (or emission) probabilities

$$b_i(y_k) = P(Y_k = y_k | Z_k = i)$$

where $\sum_{y_k} b_i(y_k) = 1$, as we are in the case of Y_k is continuous $\sum_{x_t}$ has to be replaced by $\int_{x_t}$.

Also the observation process is characterized by the conditional independence property as the output at time k depends only on the state of the underlying semi-Markov chain at time k

$$P(Y_{1:k} = y_{1:k} | Z_{1:k} = z_{1:k}) = \prod_{i=1}^{k} P(Y_k = y_k | Z_k = z_k)$$

where for convienience, we denote by $Y_{1:k}, Z_{1:k}$ the state sequence $Y_1, ...,Y_k$ and $Z_1, ..., Z_k$ respectively.

Remark. *Let us now introduce, an important notation:*

- $Z_{k_1:k_2} = i$, *represents that the system stays in state i from* t_1 *to* t_2*, analytically* $Z_{k_1} = i$, $S_{k_1+1} = i \dots S_{k_2} = i$ *and* S_{k_1-1} *and* S_{k_2+1} *may or may not be i.*
- $Z_{[k_1,k_2} = i$, *represents that* Z_k *enters at state i at time* k_1 *and remains there at least at time* k_2 *but we don't know the state at time* $k_2 + 1$.
- $Z_{k_1:k_2]} = i$, *represents that the system stays in i during the period from* k_1 *to* k_2 *but we know that* $Z_{k_2+1} \neq i$*, means that at time* k_2 *the state will end and transit to some other state at time* $t_2 + 1$.
- $Z_{[k_1:k_2]} = i$, *represents that i starts at time* k_1 *and ends at* k_2 *with duration* $d = t_2 - t_1 + 1$. *This implies that the previous state* Z_{t_1-1} *and the next* Z_{t_2+1} *must not be i.*

In addition, state duration is a random variable, let k be the duration of a state where $k \in \Delta$ $= \{1, 2, \dots, T_{max}\}$ and T_{max} is the maximum duration allowed in a state.

So we can also write

$$q_{ij}(u) = P(Z_{[k+1:k+u]} = j | Z_{k]} = i)$$

and

$$d_i(u) = P(Z_{k+1:k+u} = j | Z_{[t+1} = i)$$

which is the sojourn time distribution of each non-absorbing state and represents the state transition probability from state i having duration u to a state $j \neq i$, and the probability that the state i has duration u.

It is important the fact that in the standard formulation from the classical HSMM the end of the sequence of observations always coincides with the exit from a state. This very specific assumption does not seem to be realistic in most application like as in our application.

So, we will use the survivor function in order to estimate the duration of the last visited state.

Definition 3.7. *The survivor function of the sojourn time in a state i is defined as*

$$D_i(u) = \sum_{v \leq u} d_i(v)$$

and represents the marginal sojourn time of u by summing over all admitted sojourn time $v \geq u$, i.e limited by the upper bound T_{max}.

Note that the classical definition of the survivor function uses strict inequality in previous definition, but we adopt this convention to comply with the application of the thesis.

Remark. *We have to make attention to the fact that the first visited state does not coincide with a transition for one state to another. However the initial distribution of the semi-Markov chain should take into account the entrance time. So, we denote by $\pi_{j,u}$ the probability that the initial state is j and the time elapsed from entrance to this state is u, that is*

$$\pi_{j,u} = P(S_{[t-u+1:t]} = j), \qquad t \geq 0$$

It's worthwhile to refer that in our application the cardiac structure the state sequence follows the cyclic pattern

$$S_1 \; sound \rightarrow Systole \rightarrow S_2 \; sound \rightarrow Diastole \rightarrow S_1 \; sound \; \ldots$$

as a result the transitions are deterministic. So we can use conventional models which are simpler than the general HSMM. This leads us in a problem with fewer parameters and lower computational complexity.

Moreover the HSMM, we consider that do not allow self-transitions in the states of the semi-Markov chain and consequently

$$q_{ii}(u) = 0, \forall i \in S, u \in \Delta$$

In addition, since $f_{ij}(u) = d_i(u)$, we obtain the

$$q_{ij}(u) = p_{ij} d_i(u).$$

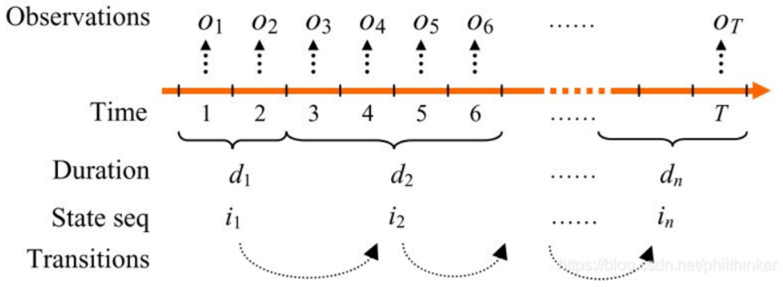

FIGURE 3.1: Graphical representation of an HSMM [35]

4 Common problems

Having set the HSMM $\lambda = (\pi, \mathrm{A}, \mathrm{B}, D)$ we have to deal with the basic problems, which are the same as in HMM. These are :

1. Evaluation(Classification): We have the observation sequence $Y_{1:T} = y_1 y_2 \dots y_T$ and the HSMM λ and we want to calculate the quantity $P(Y_{1:T}|\lambda)$, which is the probability of the observation sequence given the model.

2. Decoding (Recognition): Like the previous case we have the observation sequence $Y_{1:T} = y_1 y_2 \dots y_T$ and the HSMM λ, we want to find the sequence of hidden states.

3. Training: We desire to adjust the model parameters $\lambda = (\pi, \mathrm{A}, \mathrm{B}, D)$ to maximize $P(Y_{1:T}|\lambda)$

Different algorithms have been developed for above three problems. The most straightforward way of solving the evaluation problem is enumerating every possible state sequence of length T (the number of observations). However, the computation burden for this exhaustive enumeration is prohibitively high. Fortunately, there is a more efficient algorithm that is based on dynamic programming, called forward-backward procedure. The goal for decoding problem is to find the optimal state sequence associated with the given observation sequence. The most widely used optimality criterion is to find the single best state sequence, which is based on dynamic programming methods. For training problem, there is no known way to obtain analytical solution. However, we can adjust the model parameter $\lambda = (\pi, \mathrm{A}, \mathrm{B}, D)$ such that $P(Y_{1:T}|\lambda)$ is locally maximized using an iterative procedure, such as the Baum-Welch method or equivalently the EM (Expectation-Maximization algorithm).

5 Forward-Backward

For the observation sequence $Y_{1:T}$, the likelihood function of an HSMM for given parameters λ is given by

$$P(Y_{1:T}|\lambda) = \sum_{Z_{1:T}} P(Z_{1:T}, Y_{1:T}|\lambda)$$

As we have mentioned in order to compute the $P[Y_{1:T}|\lambda]$ we use the Forward- Backward, a efficient algorithm which based on dynamic programming. Now we define the forward variables.

The forward variables for HSMM are defined by

$$a_{j,k}(t) = P(Z_{[t-k+1:t]} = j, Y_{1:t}|\lambda)$$

and the backward variables by

$$\beta_{j,k}(t) = P(Y_{t+1:T}|Z_{[t-k+1:t]=j}, \lambda)$$

and the forward-backward algorithm for general HSMM:

$$\begin{aligned}
\alpha_{j,k}(t) &= \sum_{i\neq j,k'} P(Z_{[t-k-k'+1:t-k]} = i, Z_{[t-d+1:t]} = j, Y_{1:t}|\lambda) \\
&= \sum_{i\neq j,k'} \alpha_{i,k'}(t-k)P(Z_{[t-k+1:t]=j}, Y_{t-d+1:t}|Z_{[t-d-k'+1:t-d]}) \\
&= \sum_{i\neq j,k'} \alpha_{i,k'}(t-k)q_{ij}(k)P(Y_{t-d+1:t}|Z_{[t-k+1:t]} = j, \lambda) \\
&= \sum_{i\neq j,k'} \alpha_{t-k}(i,k')q_{ij}(k)b_{j,k}(Y_{t-k+1:t})
\end{aligned}$$

for $t > 0, k \in D, j \in S$ and

$$\begin{aligned}
\beta_{j,k}(t) &= \sum_{i\neq j,k'} P(Z_{[t+1:t+k']=i}, Y_{t+1:T}|Z_{[t-d+1:t]=j,\lambda}) \\
&= \sum_{i\neq j,k'} q_{ij}(u')P(Y_{t+1:T}|Z_{[t+1:t+k']} = i, \lambda) \\
&= \sum_{i\neq j,k'} q_{ji}(k')b_{i,k'}(Y_{t+1:t+k'})P(Y_{t+k'+1}|Z_{[t+1:t+k']} = i, \lambda) \\
&= \sum_{i\neq j,k'} h_{ji}(k')b_{i,k'}(Y_{t+1:t+k'})\beta_{i,k'}(t+1)
\end{aligned}$$

It's worthwhile that for the calculation of forward-backward variables have been used the following equations

$$P(Y_{t-k+1:t}|Z_{[t-k-k'+1:t-k]} = i, Z_{[t-k+1:t]} = j, \lambda) = P(Y_{t-d+1:t}|Z_{[t-k+1:t]} = j, \lambda)$$

and

$$P(Y_{[t+1:T]}|Z_{[t-k+1:t]} = j, Z_{[t+1:t+k']} = i, \lambda) = P(Y_{t+1:T}|Z_{[t+1:t+k']} = i, \lambda)$$

that means the current /future observation are dependent on the current state based on Markov property.

Also, current/future observation are independent of the previous observation,for example

$$P(Z_{[t-k+1:t]} = j, Y_{t-k+1:t}|Z_{[t-k-k'+1:t-k]} = i, Y_{1:t-k}, \lambda) =$$
$$P(Z_{[t-k+1:t]} = j, Y_{t-d+1:t}|Z_{[t-k-k'+1:t-d)=i}, \lambda)$$

and

$$P(Y_{t+k+1:T}|Z_{[t+1:t+k]} = i, Y_{t+1:t+k}, \lambda) = P(Y_{t+k+1:T}|S_{[t+1:t+k]} = i, \lambda)$$

Now it is important to set the initial conditions. Let assume that the first state must start at $t = 1$, let $\alpha_{j,k}(t) = 0$, for $t < 0$, otherwise if it starts at $t < 1$ $\alpha_{j,k}(t) = \pi_{j,k}$ for $t < 0$. Respectively, if we consider that the last state must end at time $t = T$, then $\beta_{j,k}(t) = 0$ for $t > T$, otherwise if it ends at time $t > T$ $\beta_{j,k}(t) = 1$ for $t > T$.

In order to simplify the definition of the forward-backward variables for the HSMM, let

$$\alpha_j(t) = P(Z_{t]} = j, Y_{1:t}|\lambda) = \sum_{\kappa \in \Delta} \alpha_{j,\kappa}(t)$$

which represents the joint probability that state j ends at time t and the partial observation sequence is $y_{1:t}$, and,

$$\beta_j(t) = P(Y_{t+1:T}|Z_{t]} = j, \lambda)$$

which represents the conditional probability that given that state i ends at time t, the future observation sequence is $y_{t+1:T}$. Hence, we can easily obtain the respective simplified recursive formulas for the HSMM.

6 Viterbi Algorithm

As for the decoding problem, one of the most frequently used methodologies, in recent times is the Viterbi algorithm. This is a dynamic algorithm that computes the most likely sequence of states. So we desire to find a state path to maximize $P(Z_{1:T}|Y_{1:T}, \lambda)$.

The Viterbi algorithm for a HSMM is defined by

$$\delta_t(j,k) = \max_{Z_{1:t-k}} P(S_{1:t-k}, Z_{[t-k+1:t]} = j, Y_{1:t}|\lambda) =$$

$$\max_{i\neq j} \max_{k'} \max_{Z_{1:t-k-k'}} P(S_{1:t-k-k'}, S_{[t-k-k'+1:t-k]} = i,$$

$$S_{[t-k+1:t]} = j, Y_{1:t}|\lambda) =$$

$$\max_{i\neq j} \max_{k'} [\delta_{t-k}(i,k') q_{ij}(k) b_{j,k}(Y_{t-k+1:t})]$$

for $1 \leq t \leq T, j \in S, k \in D$, $\delta_t(j,d)$ represents the most likely partial state sequence that ends at t in state j of duration k. In order to find the optimal path we should also record the previous state. So let $\psi(t, S_n, k) = (t-k, S_{n-1}, k')$ be a function where

$$Z_{n-1} = \text{previous state survived}$$

$$k' = \text{duration of the previous state}$$

$$t-k = \text{ending time of the previous state}$$

Determinating ψ(t,j,d) by letting

$$(Z_{n-1}, k') = \arg\max_{i\in Z'_j} \max_{k'\in D} \delta_{t-d}(i,k') q_{ij} b_{j',k}(Y_{t-d+1:t})$$

After we can determine the state sequence by finding the last state that maximizes the likelihood. Assuming that the last state ends at time $t = T$, then
$t_1 = T$
$(j_1, k_1) = \arg\max_{j\in Z} \max_{u\in\Delta} \delta_T(j,k)$
otherwise,

$$(t_1, Z_1, k'_1) = \arg\max_{T\leq t\leq t+T_{max}-1} \max_{j\in Z} \max_{t-T+1\leq u\leq T_{max}} \delta_T(j,k)$$

Hence, for n=2,3,. . . , we can trace back the state sequence by letting

$$(t_n, Z_n, k'_n) = \psi(t_{n-1}, Z_{n-1}, k_{n-1})$$

7 EM ALGORITHM

As for the Training problem in order to deal with it and to maximize the $P(Y|\lambda)$ adjusting model parameters have been used the EM algorithm. The Expectation-Maximization (EM) algorithm [13] is a broadly applicable approach to compute maximum likelihood (ML) estimates iteratively. This use in a variety of incomplete-data problems, where algorithms such as the Newton-Raphson method may fall through. On each iteration of the EM algorithm there are two steps-called the expectation step of the E-step and the maximization step or the M-step. The situations where the EM algorithm is profitably applied can be described as incomplete-data problems, where ML estimation is made difficult by the absence of some part of data in a more familiar and simpler data structure. The EM algorithm is closely related to the ad hoc approach to estimation with missing data. The latter are then updated by their predicted values, using these initial parameter estimates. The parameters are then reestimated, and so on, proceeding iteratively until convergence. But also EM algorithm can be used in the situations where the incompleteness of the data is not all that natural or evident. These include statistical models such as random effects, mixtures, convolutions, log linear models, and latent class and latent variables structures. In our case we would like to maximize the likelihood function but we cannot do it so using adding latent variables smooth the likelihood. We maximize $logL(Y|\lambda) = l_y$ instead because it is analytically easier.

In order to use EM, first of all we must have some observed data y, a parametric density $p(y|\lambda)$, a description of some complete data z that you wish you had, and the parametric density $p(z|\lambda)$. We assume that the complete data can be modeled as a continuous random vector Z with density $p(z|\lambda)$, where $\lambda \in \Omega$ for some set Ω. Because of the fact that we don't observe Z directly instead, we observe a realization y of the random vector Y that depends on Z. Given that you only have y, the goal here is to find the maximum likelihood estimate (MLE) of λ

$$\hat{\lambda}_{MLE} = \arg\max_{\lambda \in \Omega} p(y|\lambda) \tag{3.1}$$

Most of times it is much easier to calculate the λ that maximizes the log-likelihood of y because log is a monotonically increasing function, the solution (3.1) will be the same as the solution to

$$\hat{\lambda}_{MLE} = \arg\max_{\lambda \in \Omega} \log p(y|\lambda) \tag{3.2}$$

However, when we cannot solve either (3.1) neither (3.2) then we can try EM making a guess about the complete data Z and solving for the λ that maximizes the (expected) log-likelihood of Z. And once we have an estimate for λ we can make a better guess about the complete data Z, and iterate. EM is usually described as two steps (the E-step and M-step), but let us first break it down into five steps:

1. Let $m = 0$ and make an initial estimate $\lambda^{(m)}$ for λ.

2. Given the observed data y and preteding for the moment that your current guess $\lambda^{(m)}$ is correct formulate the conditional probability distribution $p(z|y, \lambda^{(m)})$ for the complete data z.

3. Using the conditional probability distribution $p(z|y, \lambda^{(m)})$ calculated in Step 2, form the conditional expected log likelihood, which is called the Q-function:

$$Q(\lambda|\lambda^{(m)}) = \int_{Z(y)} \log p(z|\lambda) p(z|y, \lambda^{(m)}) dz E_{Z|y,\lambda^m}[\log p(z|\theta)] \tag{3.3}$$

   where the integral is over the set $Z(y)$, which is the closure of the set $z|p(z|y, \lambda) > 0$, and we assume that $Z(y)$ does not depend on λ.

   Note that λ is a free variable in (3.3), so the Q-function is a function of λ, but also depends on current guess $\lambda^{(m)}$ implicity though the $p(z|y, \lambda^{(m)})$ calculated in Step 2.

4. Find the λ that maximizes the Q-function (3.3), the result is your new estimate $\lambda^{(m+1)}$

5. Let $m = m + 1$ and go back to Step 2. The EM algorithm does not specify a stopping criterion; standard criteria are to iterate until the estimate stops changing: $||\lambda^{(m+1)} - \lambda^{(m)}|| < \epsilon$ for some $\epsilon > 0$, or to iterate until the log-likelihood $l(\lambda) = \log p(y|\lambda)$ stops changing: $|l(\lambda^{(m+1)}) - l(\lambda^{(m)})| < \epsilon$ for some $\epsilon > 0$.

It is important that the EM estimate is only guaranteed to never get worse. Usually, it will find a peak in the likelihood $p(y|\lambda)$, but if the likelihood function $p(y|\lambda)$ has multiple peaks, EM will not necessarily find the global maximum of the likelihood. In practise, it is common to start EM from multiple random initial guesses and choose the one with the largest likelihood as the final guess for λ.

The traditional description of the EM algorithm consists of only two steps. The above Steps 2 and 3 combined are called the E-step for expectation,and Step 4 is called the M-step for maximization:

E-step: Given the estimate from the previous iteration $\lambda^{(m)}$ compute the conditional expectation $Q(\lambda|\lambda^{(m)})$ given in (3.3).

M-step: The $(m + 1)^{th}$ guess of λ is

$$\lambda^{(m+1)} = \arg\max_{\lambda \in \Omega} Q(\lambda|\lambda^{(m)}). \tag{3.4}$$

Since the E-step is just to compute the Q-function which is used in the M-step given by (3.4). When applying EM to a particular problem, this is usually the best way to think about EM because then one does not waste time computing parts of the Q-function these do not depend on λ.

7.1 Convergence of EM

Here is what can be proved without extra conditions as the EM algorithm iterates, the $(m+1)^{th}$ guess $\lambda^{(m+1)}$ will never be less likely then the m^{th} guess $\lambda^{(m)}$. This property is called the monotonicity of the EM algorithm, and results from the following theorem, which states that improving the Q-function will at least not make the log-likelihood $l(\lambda)$ worse.

Theorem 3.1. *Let random variables Z and Y have parametric densities with parameter $\lambda \in \Omega$. Suppose the support of Z does not depend on λ, and the Markov relationship $\lambda \to Z \to Y$, that is*

$$p(y|z,\theta) = p(y|z)$$

holds for all $\lambda \in \Omega$, $z \in \mathcal{Z}$ and $y \in \mathcal{Y}$. Then for $\lambda \in \Omega$ and any $y \in \mathcal{Y}$ with $\mathcal{Z}(y) \neq \emptyset$, $l(\lambda) \geq l(\lambda^{(m)})$ if $Q(\lambda^{(m)}) \geq Q(\lambda^{(m)})$.

For the EM algorithm, the M-step ensures that

$$\lambda^{(m+1)} = \underset{\theta \in \Omega}{\arg\max}\, Q(\lambda|\lambda^{(m)})$$

and hence it must be that $Q(\lambda^{(m+1)}|\lambda^{(m)}) \geq Q(\lambda^{(m)}|\lambda^{(m)})$. Therefore, we can apply Theorem 3.1 and conclude that $l(\lambda^{(m+1)}) \geq l(\lambda^{(m)})$.

$$
\begin{aligned}
l(\lambda) &= \log p(y|\lambda) \qquad \text{(by definition)} \\
&= \log \int_{Z(y)} p(z,y|\lambda) dz \qquad \text{(by the law of total probability)} \\
&= \log \int_{Z(y)} \frac{p(z,y|\lambda)}{p(z|y,\lambda^{(m)})} p(z|y,\lambda^{(m)}) dz \qquad \text{(multiply the top and bottom by the same factor)} \\
&= \log E_{Z|y,\lambda^{(m)}} [\, \frac{p(Z,y|\lambda)}{p(Z|y,\lambda^{(m)})}] \qquad \text{(rewrite the integral as an expectation)} \\
&\geq E_{Z|y,\lambda^{(m)}} [\log \frac{P(Z,y|\lambda)}{P(Z|y,\lambda^{(m)})}] \qquad \text{(by Jensen's inequality)} \\
&= E_{Z|y,\lambda^{(m)}} [\log \frac{p(Z|\lambda)p(y|\lambda)}{p(Z|\lambda^{(m)})p(y|Z)}] \qquad \text{(by Bayes' rule and the assumed Markov relationship)} \\
&= E_{Z|y,\lambda^{(m)}} [\log \frac{p(z|\lambda)p(y|\lambda^{(m)})}{p(x|\lambda^{(m)})}] \\
&= E_{Z|y,\lambda^{(m)}} [\log p(z|\lambda)] - E_{Z|y,\lambda^{((m)}} [\log p(Z|\lambda^{(m)})] + \log p(y|\lambda^{(m)}) \\
&= Q(\lambda|\lambda^{(m)}) - Q(\lambda^{(m)}|\lambda^{(m)}) + l(\lambda^{(m)})
\end{aligned}
$$

where the Q-function is defined in (3.3). Note that because of the assumption that the support of Z does not depend on λ, combined with assumed Markov relationship, we can easily conclude that $\mathcal{Z}(y)$ does not depend on λ, (2.2) can lead to $\frac{0}{0}$ and the rest of the proof won't follow.

We can conclude the first part of the proof by restarting (2.3) as a lower bound on the log-likelihood

$$l(\lambda) \geq l(\lambda^{(m)}) + Q(\lambda|\lambda^{(m)}) - Q(\lambda^{(m)}|\lambda^{(m)})$$

Notice that in the above lower bound, $Q(\lambda|\lambda^{(m)})$ is the only term that depends on λ.

Next since we assume that $Q(\lambda|\lambda^{(m)}) \geq Q(\lambda^{(m)}|\lambda^{(m)})$, we can simply conclude that

$$l(\lambda) \geq l(\lambda^{(m)}) + Q(\lambda|\lambda^{(m)}) - Q(\lambda^{(m)}|\lambda^{(m)}) \geq l(\lambda^{(m)})$$

which completes the proof.

The monotonicity of the EM algorithm guarantees that as EM iterates, its guesses won't get worse in terms of their likelihood, but the monotonicity alone cannot guarantee the convergence of the sequence $\lambda^{(m)}$. Indeed, there is no general convergence theorem for the EM algorithm the convergence of the sequence $\lambda^{(m)}$ depends on the characteristics of $l(\lambda)$ and $Q(\lambda|\lambda')$, and also the starting point $\lambda^{(0)}$.

Under certain regularity conditions, we can prove that $\lambda^{(m)}$ converges to a stationary point (for example, a local maximum or saddle point) of $l(\lambda)$. However, this convergence is only linear instead of using EM algorithm one could (locally) maximize the likelihood using Newton-Raphson updates, which requires calculating the inverse of, the Hessian matrix, but has quadratic convergence. Superlinear convergence could instead be achieved using conjugate gradient methods or quasi-Newton updates such as the Broyden-Fletcher-Goldferb-Shamon (BFGS) update, which only require computing the gradient of the log-likelihood. The Newton-Raphson method can be expected to home in on λ^* fast once $\lambda^{(m)}$ is close, but EM may be more effective given a poor initial guess in part because the Hessian matrix for the Newton-Raphson method may not be positive definite and hence makes the inversion unstable.

Chapter 4

Results

In the last chapter of this thesis, we present some results that we obtained by applying the methodology that we presented related to the problem of segmentation of the cardiac cycle, including a statistical analysis on a specific dataset. Moreover, we outline the metrics that we used in order to measure the performance of the proposed algorithm, as well as the final results of each model.

1 Pre-processing and Feature extraction

A broad dataset of heart sounds was published in 2016 by PhysioNet. All the data were collected from various research groups and obtained in different environments both clinical and non-clinical. In this assignment, we use recordings of 792 heart sounds from 135 patients taken from the Physionet dataset, where multiple recordings per patient are generally available in order to include sounds from several spots over the chest to account for corruptions due to different sources and noise levels. Out of 792 recordings, 386 sounds correspond to healthy patients with no observed heart abnormalities, while the rest 406 sounds are aggregated from patients with pathological heart lesions, namely mitral valve prolapse. The sampling frequency of these recordings is 1000 Hz and their duration varies from 1 to 35,5 seconds. Together with the heart sound recordings, the PCG-based annotations are also provided in order to label each state later. Given this dataset we continue to the pre-processing method. Initially, all the signals are filtered using a Butterworth bandpass filter (see section 2.2) of order 4, with cutoff frequencies at 25 Hz and 400 Hz, since higher frequencies are not of clinical significance for analysis. Additionally, we use a filter to eliminate unwanted spikes created by the presence of abnormalities. Now, using the filtered signal we extract three different features: the homomorphic envelogram (see section 2.3), the hilbert envelogram (see section 2.4) and the power spectral density envelope (see section 2.5). After each feature is extracted, it is normalized and then downsampled to 50 Hz in order to improve the speed of computations. Moreover we use PCG-based annotations in order to label each observation with the respective state. To do that not only we use

PCG-based annotations but also we use the counts of Schmidt. Specifically, the S_1 sound is considered to have average duration of 0,122 sec and a standard deviation of 0,022 sec, while the S_2 sound an average duration of 0,092 sec and a standard deviation of 0,022 sec. In contrast, the average duration of the systolic and diastolic period are varying for each individual since the duration of the heart cycle is highly variable from subject to subject. As a result, an adapted procedure is needed to tune better counts corresponding to the sojourn time of the remaining states. To achieve this goal Schmidt use an auto-correlation analysis preferring to use the homomorphic envelope, as it is smoother than the original signal. So, Schmidt deduces that

$$\bar{d}(Sys) = \arg\{k \in \mathbb{N} : \rho_{\kappa}\} - \bar{d}, (S_1)$$

with a standard deviation of 0,025 sec. The average diastolic duration is inferred from the heart cycle and the duration of the other states, i.e

$$\bar{d}(Dias) = dHR - \left|\bar{d}(S_1) + \bar{d}(Sys) + \bar{d}(S_2)\right|,$$

where dHR is the estimated duration of the diastolic duration in the recordings was partly correlated with $\bar{d}(Dias)$. Therefore the relation between the diastolic duration and the standard deviation of the diastolic duration was determined by the formula $sd(Dias) = 0,07 < \bar{d}(Dias) + 0,006$, as established by Schmidt. For the implementation of the proposed methodology we used the R statistical software.

2 Parameter estimation and Decoding

In this section, we present a method for estimating the parameters of the emission distributions and sojourn time distributions of the different hidden states. We adopt an HSMM in which each state of the semi-Markov chain corresponds to a specific heart sound $\in \{S_1, Systole, S_2, Diastole\}$, as it is assumed that the signal characteristics in each state are homogeneous. To begin with the training phase as well as the initialization process that we selected which is in accordance with the method applied in.

2.1 Training phase

First of all, we randomly split the dataset into a training and a test set. After the training set has been determined as explained above, it is used in the training phase to fit each of the corresponding HSMMs. We remind that the probability (or density) of observing y_t conditioned on being in state j called emission probability and we denoted by $b_j(y_t)$. We have assumed that $b_j(y_t)$ corresponds to a density of a Normal distribution with mean and variance the empirical mean and variance of the training set which differs for each state j. As for the distribution of the sojourn time we have two different modeling assumptions. The former is the Poisson and the latter is the Gamma.

Normal emission distributions

If the emission distribution is the Normal then:

$$b_j(y_t) = \frac{1}{\sigma_j \sqrt{2\pi}} \exp\left(-\frac{1}{2} \frac{(y_t - \mu_j)^2}{\sigma_j^2} \right)$$

where μ_j is the mean of the distribution and σ_j is its standard deviation.

Poisson sojourn distribution

If the distribution of the sojourn time is based on the Poisson distribution then:

$$d_j(u|\lambda_j) = \exp\{-\lambda_j\} \frac{\lambda_j^u}{u!}$$

where λ_j is the expected sojourn time in state j.

Gamma sojourn distribution

Also if the distribution of the sojourn time is based on the Gamma distribution, then the density is given by:

$$d_j(u|k_j, \theta_j) = \frac{1}{\Gamma(\kappa_j)\theta_j^{k_j}} u^{k_j - 1} e^{\frac{-u}{\theta_j}}, \qquad u > 0$$

where $k_j > 0, \theta_j > 0$ are the shape and scale parameters respectively.

Remark. *If $X \sim Gamma(\kappa, \theta)$ is a random variable and the shape parameter κ is large relative to the scale parameter θ, then X can be well approximated by a normal random variable with the same mean and variance.*

To initialize the parameters of sojourn time distributions and emission distributions we use the empirical mean and variance of each state. The Normal emission distribution the μ_j and σ_j parameters is initialized with empirical mean and variance of observed data for each state. As for Gamma sojourn distribution

$$m_j = \kappa_j \theta_j, \qquad s_j^2 = \kappa_j \theta_j^2$$

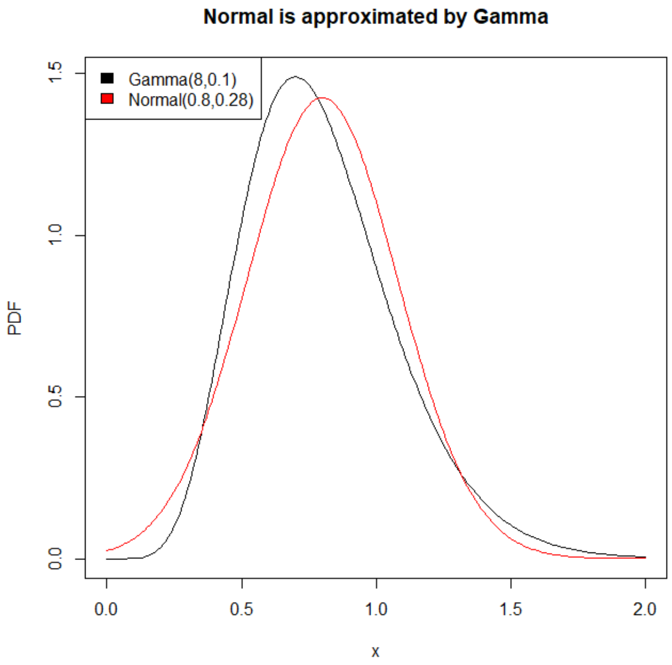

FIGURE 4.1

that is

$$k_j = \frac{m_j^2}{s_j^2}, \qquad \theta_j = \frac{s_j^2}{m_j}$$

where m_j, s_j^2 are empirical mean and variance of sojourn time in each state. Finally, as for Poisson sojourn distribution the λ_j parameter is initialized with the empirical mean of sojourn time in each state.

2.2 Application of the EM Algorithm

Finishing the data pre-processing and feature extraction, we continue to determine the set of parameters that maximize the incomplete likelihood of a given sequence of observations y of heart sound signal under test and that we can do it using the EM algorithm. As we have mentioned in (3.2.3), the EM algorithm contains the computations and the maximization of the auxiliary function (see equation 3.12). So EM algorithm maximizes $L(\lambda)$ by iteratively maximizing $Q(\lambda|\lambda^{(m)})$ (auxiliary function) over λ. The next value $\lambda^{(m+1)}$

$$\lambda^{(m+1)} = \arg\max\{Q(\lambda|\lambda^{(m)})\}$$

As referred each iteration of the EM algorithm increases $L(\lambda)$ and, generally, the sequence of reestimated parameters $\lambda^{(m)}$ converge to a local maximum of $L(\lambda)$. As for $Q(\lambda|\lambda^{(m)})$ it is important that given the set of parameters λ this can be rewritten as sum of terms, where each term depending on a given subset of parameters

$$\begin{aligned} Q(\lambda|\lambda^{m}) = {} & Q_{\pi}(\{\pi_j\}_{j=1}^{J}|\lambda^{(m)}) + \sum_{i=1}^{J} Q_p(\{p_{ij}\}_{j=1}^{J}|\lambda^{(m)}) + \sum_{j=1}^{J} Q_d(\{d_j(u)\}|\lambda^{(m)})\mathbb{1}(p_{jj}=0) \\ & + \sum_{j=1}^{J} Q_b(\{b_j(y_t)\}_{t=1}^{Y}|\lambda^{(m)}) \end{aligned} \tag{4.1}$$

with

$$Q_{\pi}(\{\pi_j\}_{j=1}^{J}|\lambda^{(m)}) = \sum_{j} P(Z_0 = j|Y_{[1:T]} = y_{[1:T]}\,;\lambda^{(m)}) \log \pi_j \tag{4.2}$$

$$Q_p(\{p_{ij}\}_{j=1}^{J}|\lambda^{(m)}) = \sum_{j\neq i}\sum_{t=0}^{T-1} P(Z_{t+1} = j, Z_t = i|Y_{[1:T]} = y_{[1:T]}\,;\lambda^{(m)}) \log p_{ij} \tag{4.3}$$

$$\begin{aligned} Q_d(\{d_j(u)\}|\lambda^{(m)}) = {} & \sum_{u}\{\sum_{t=0}^{T-1} P(Z_{t+u+1} \neq j, Z_{t+u-v} = j, v = 0, 1, \ldots, u-1, Z_t \neq j|Y_{[1:T]} = y_{[1:T]}\,;\lambda^{(m)}) \\ & + P(Z_u \neq j, Z_{u-v} = j, v = 1, \ldots, u|Y_{[1:T]} = y_{[1:T]}\,;\lambda^{(m)})\} \log d_j(u) \end{aligned} \tag{4.4}$$

and

$$Q_b(\{b_j(y_t)\}_{t=1}^{T}|\lambda^{(m)}) = \sum_{t=1}^{T} P(Y_t = y_t, Z_t = j|Y_{[1:T]} = y_{[1:T]}\,;\lambda^{(m)}) \log b_j(y_t) \tag{4.5}$$

where $\lambda^{(m)}$ are the estimated values of λ at the iteration m of the algorithm. As for our application, the terms that contain the initial and transition probabilities are not updated through the maximization step. As the initial distribution is considered to be fixed and transition probabilities are only 0 and 1, since state changes occur in a deterministic way. In contrast the terms that contains emission probabilities parameters and sojourn time parameters are these whose we have to maximize. So, our interest lies in the maximization of (4.4) and (4.5). But to maximize the above quantities it is necessary the computation of specific state posterior probabilities, i.e forward variable, backward variable as well as the following quantities.

$$L1_j(t) = P(Z_{t+1} \neq j, Z_t = j|Y_{[1:T]}), \qquad 1 \leq j \leq J$$

As a result, in the case of a hidden semi-Markov chain, the forward-backward equations can be decomposed as in [] as follows:

$$\begin{aligned} L1_j(t) &= P(Z_{t+1} \neq j, Z_t = j|Y_{[1:T]} = y_{[1:T]}) \\ &= \frac{P(Y_{[t+1:T]} = y_{[t+1:T]}|Z_{t+1} \neq j, Z_t = j)}{P(Y_{t+1:T} = y_{t+1:T}|Y_{[1:t]} = y_{[1:t]})} P(Z_{t+1} \neq j, Z_t = j|Y_{[1:T]} = y_{[1:T]}) \\ &= B_j(t)F_j(t) \end{aligned}$$

which expresses the conditional independence between the past and the future of the process at stage change times. The notations $B_j(t)$, $F_j(t)$ are used for the backward and forward variables respectively. More specifically, the forward recursion is given by:

$$\begin{aligned} F_j(t) &= P(Z_{t+1} \neq j, Z_t = j|Y_{[1:T]} = y_{[1:T]}) \\ &= \sum_{u=1}^{t} \sum_{i \neq j} P(Z_{t+1} \neq j, S_{t-v} = j, v = 0, \dots, u-1, S_{t-u} = i|Y_{[1:t]} = y_{[1:t]}) \\ &+ P(Z_{t+1} \neq j, Z_{t-v} = j, v = 0, \dots, t|Y_{[1:t]} = y_{1:t}) \\ &= \sum_{u=1}^{t} \frac{P(Y_{[t-u+1:t]} = y_{[t-u+1:t]}|Z_{t-v} = j, v = 0, \dots, u-1)}{P(Y_{[t-u+1:t]} = y_{[t-u+1:t]}|Y_{[1:t-u]} = y_{[1:t-u]})} \\ &\times P(Z_{t+1} \neq j, Z_{t-v} = j, v = 0, \dots, u-2|Z_{t-u+1} = j, Z_{t-u} \neq j) \\ &\times \sum_{i \neq j} P(Z_{t-u+1} = j|Z_{t-u+1} \neq i, Z_{t-u} = i) P(Z_{t-u+1} \neq i, Z_{t-u} = i|Y_{[1:t-u]} = y_{[1:t-u]}) \\ &+ \frac{P(Y_{[1:t]} = y_{[1:t]}|Z_{t-v} = j, v = 0, \dots t)}{P(Y_{[1:t]} = y_{[1:t]})} * P(S_{t+1} \neq j, S_{t-v} = j, v = 0, \dots, t) \\ &= \frac{b_j(y_t)}{N_t} [\sum_{u=1}^{t} \{\prod_{v=1}^{u-1} \frac{b_j(y_{t-v})}{N_{t-v}}\} d_j(u) \sum_{i \neq j} p_{ij} F_i(t-u) + \{\prod_{v=1}^{t} \frac{b_j(y_{t-v})}{N_{t-v}}\} d_j(t+1)\pi_j] \end{aligned} \tag{4.6}$$

for $t = 1, \dots, T-1, j = 1, \dots, J$. Concerning the censoring at time T of the sojourn time in the last visited state, we obtain for time $t = T$

$$\begin{aligned} F_j(T) &= P(Z_T = j|Y_{[1:T]} = y_{[1:T]}) \\ &= \frac{b_j(y_T)}{N_T} [\sum_{u=1}^{T} \{\prod_{v=1}^{u-1} \frac{b_j(y_{T-v})}{N_{T-v}}\} D_j(u) \sum_{i \neq j} p_{ij} F_i(T-u) \\ &+ \{\prod_{v-1}^{T} \frac{y_{T-v}}{N_{T-v}}\} D_j(u)\pi_j] \end{aligned} \tag{4.7}$$

The exact time spent in the last visited state is unknown, only the minimum time spent in this state is known. Therefore, the probability mass functions of the sojourn times in state j of the general forward recursion formula (4.7) are replaced by the corresponding survivor functions (see Definition 3.7).

The normalizing factor N_t is directly obtained during the forward recursion, that is

$$\begin{aligned}
N_t &= P(Y_t = y_t | Y_{[1:T]} = y_{[1:T]}) \\
&= \sum_j P(Z_j = j, Y_t = y_t | Y_{[1:T]} = y_{[1:T]}) \\
&= \sum_j b_j(y_t)[\sum_{u=1}^{t} \{\prod_{v=1}^{u-1} \frac{b_j(y_{t-v})}{N_{t-v}}\} D_j(u) \sum_{i \neq j} F_i(t-u) \{\prod_{v=1}^{t} \frac{b_j(y_{t-v})}{N_{t-v}}\} D_j(t+1)\pi_j]
\end{aligned} \tag{4.8}$$

The backward recursion consists of computing $L_j(t) = P(Z_t = j | Y_{[1:T]} = y_{[1:T]})$ for each state j backward from time T to time 0. The backward recursion is initialized for $t = T$ and $j = 1, \ldots, J$

$$L_j(T) = P(Z_T = j | Y_{[1:T]} = y_{[1:T]}) = F_j(T)$$

The key point here lies in the rewriting of $L_j(t)$ as three terms, $L1_j(t), L_j(t+1)$ computed previously and a third term which expresses the entrance into state j:

$$\begin{aligned}
L_j(t) &= P(Z_t = j | Y_{[1:T]} = y_{[1:T]}) \\
&= P(Z_{t+1} \neq j, Z_t = j | Y_{[1:T]} = y_{[1:T]}) + P(Z_{t+1} = j | Y_{[1:T]} = y_{[1:T]}) \\
&- P(Z_{t+1} = j, Z_t \neq j | Y_{[1:T]} = y_{[1:T]}) \\
&= L1_j(t) + L_j(t+1) - P(Z_{t+1} = j, Z_t \neq j | Y_{[1:T]} = y_{[1:T]})
\end{aligned} \tag{4.9}$$

Now, the backward recursion is based on $L1_j(t)$ for $t = T-1, \ldots, 1$ and $j = 1, \ldots, J$, given by :

$$\begin{aligned}
L_j(t) &= P(Z_t = j | Y_{[1:T]} = y_{[1:T]}) \\
&= \sum_{k \neq j} \sum_{u=1}^{T-1-t} P(Z_{t+u+1} \neq k, Z_{t+u-v} = k, v = 0, \ldots, u-1, Z_t = j | Y_{[1:T]} = y_{[1:T]}) \\
&+ P(Z_{T-v} = k, v = 0, \ldots, T-1-t, Z_t = j | Y_{[1:T]} = y_{[1:T]})
\end{aligned} \tag{4.10}$$

For general term in (4.9),we have the following decomposition

$$
\begin{aligned}
G &= P(Z_{t+u+1} \neq k, Z_{t+u-v} = k, v = 0, \dots, u-1, Z_t = j | Y_{[1:T]} = y_{[1:T]}) \\
&= \frac{P(Z_{t+u+1} \neq k, Z_{t+u-v} = k, v = 0, \dots, u-1, Z_t = j, Y_{[1:T]} = y_{[1:T]})}{P(Z_{t+u+1} \neq k, Z_{t+u} = k, Y_{[1:T]} = y_{[1:T]})} \\
&\times P(Z_{t+u+1} \neq k, Z_{t+u} = k | Y_{[1:T]} = y_{[1:T]}) \\
&= \frac{P(Y_{[t+u+1:T]} = y_{[t+u+1:T]} | Z_{t+u+1} \neq k, Z_{t+u} = k) P(Z_{t+u+1} \neq k, Z_{t+u} = k) | Y_{[1:T]} = y_{[1:T]}}{P(Y_{[t+u+1:T]} = y_{t+u+1:T} | Z_{t+u+1:T} \neq k, Z_{t+u} = k) P(Z_{t+u+1} \neq k, Z_{t+u} = k | Y_{[1:t+u]} = y_{[1:t+u]})} \\
&\times \frac{P(Y_{[t+1:t+u]} = y_{t+1:t+u} | Z_{t+u-v} = k, v = 0, \dots, u-1)}{P(Y_{[t+1:t+u]} = y_{t+1:t+u} | Y_{[1:t]} = y_{[1:t]})} \\
&\times P(Z_{t+u+1} \neq k, Z_{t+u-v} = k, v = 0, \dots, u-2 | Z_{t+1} = k, Z_t \neq k) \\
&\times P(Z_{t+1} = k | Z_{t+1} \neq j, Z_t = j) P(Z_{t+1} \neq j, Z_t = j | Y_{[1:t]} = y_{[1:t]}) \\
&= \frac{L1_k(t+u)}{F_k(t+u)} \{ \prod_{v=0}^{u-1} \frac{b_k(y_{t+u-v})}{N_{t+u-v}} \} d_k(u) p_{jk} F_j(t)
\end{aligned}
\tag{4.11}
$$

while for the second term in equation (4.10), we have respectively

$$P(Z_{T-v}=k, v=0,\dots,T-1-t, Z_t=j|Y_{[1:T]}=y_{[1:T]}) = \left\{\prod_{v=0}^{T-1-t}\frac{b_k(y_{T-v})}{N_{T-v}}\right\} D_k(T-t)p_{jk}F_j(t) \tag{4.12}$$

Lastly, based on the decomposition described in equations 4.11, 4.12 we obtain for $L1_j(t)$:

$$L1_j(t) = \left[\sum_{k\neq j}\left[\sum_{u=1}^{T-1-t}\frac{L1_k(t+u)}{F_k(t+u)}\left\{\prod_{v=0}^{u-1}\frac{b_k(y_{t+u-v})}{N_{t+u-v}}\right\}d_k(u) + \left\{\prod_{v=0}^{T-1-t}\frac{b_k(y_{T-v})}{N_{T-v}}\right\}D_k(T-t)\right] p_{jk}\right] F_j(t) \tag{4.13}$$

Following the equation 4.9, the third term is given by:

$$\begin{aligned}
P(Z_{t+1}=j, Z_t\neq j|Y_{[1:T]}=y_{[1:T]}) &= \sum_{u=1}^{T-1-t}\sum_{i\neq j}P(Z_{t+u+1}\neq j, Z_{t+u-v}=j, v=0,\dots,u-1, Z_t=i|Y_{[1:T]}=y_{[1:T]}) \\
&+\sum_{i\neq j}P(Z_{t-v}=j, v=0,\dots,T-1-t, Z_t=i|Y_{[1:T]}=y_{[1:T]}) \\
&=[\sum_{u=1}^{T-1-t}\frac{L1_j(t+u)}{F_j(t+u)}\left\{\prod_{v=0}^{u-1}\frac{b_j(y_{T-v})}{N_{T-v}}\right\}d_j(u) \\
&+\left\{\prod_{v=0}^{T-1-t}\frac{b_j(y_{T-v})}{N_{T-v}}\right\}D_j(T-t)]\sum_{i\neq j}p_{ij}F_i(t)
\end{aligned} \tag{4.14}$$

Immediately, computing $L_j(t)$ may seem more than complex, however the computation of $L1_j(t)$ in (4.13) and (4.14) can be simplified by introducing the above auxiliary quantities:

$$G_j(t+1,u) = \frac{L1_j(t+u)}{F_j(t+u)}\left\{\prod_{v=0}^{u-1}\frac{b_j(y_{t+u-v})}{N_{t+u-v}}\right\}d_j(u), u=1,\dots,T-1-t$$

$$G_j(t+1,T-t) = \left\{\prod_{v=0}^{T-1-t}\frac{b_j(y_{T-v})}{N_{T-v}}\right\}D_j(T-t)$$

and

$$\begin{aligned}
G_j(t+1) &= \frac{P(Y_{[t+1:T]}=y_{[t+1:T]}|Z_{t+1}=j, Z_t\neq j)}{P(Y_{[t+1:T]}=y_{[t+1:T]}|Y_{[1:t]}=y_{[1:t]})} \\
&= \sum_{u=1}^{T-t}G_j(t+1,u)
\end{aligned}$$

At each time t, these auxiliary quantities should be precomputed, then

$$L1_j(t) = \left\{ \sum_{k \neq j} G_k(t+1) p_{jk} F_j(t) \right\}$$

and

$$\begin{aligned} P(Z_{t+1} = j, Z_t \neq j | Y_{[1:T]} = y_{[1:T]}) &= \frac{P(Y_{[t+1:T]} = y_{[t+1:T]} | Z_{t+1} = j, Z_t \neq j)}{P(Y_{[t+1:T]} = y_{t+1:T} | Y_{[1:t]} = y_{[1:t]})} \\ &\times P(Z_{t+1} = j, Z_t \neq j | Y_{[1:t]} = y_{[1:t]}) \\ &= G_j(t+1) \sum_{i \neq j} p_{ij} F_i(t) \end{aligned}$$

Because for each t<T, $L1_j(t) = B_j(t)F_j(t)$, the backward recursion based on $B_j(t)$ is directly deduced from (4.13)

$$\begin{aligned} B_j(t) = \sum_{k \neq j} [\sum_{u=1}^{T-1-t} B_k(t+u) \left\{ \prod_{v=0}^{u-1} \frac{b_k(y_{t+u-v})}{N_{t+u-v}} \right\} d_k(u) \\ + \left\{ \prod_{v=0}^{T-1-t} \frac{b_k(y_{T-v})}{N_{T-v}} \right\} D_k(T-t)] p_{jk} \end{aligned}$$

and the third term in (4.9) can be written as:

$$\begin{aligned} P(Z_{t+1} = j, Z_t \neq j | Y_{[1:T]} = y_{[1:T]}) = [\sum_{u=1}^{T-1-t} B_j(t+u) \\ \times \left\{ \prod_{v=0}^{u-1} \frac{b_j(y_{t+u-v})}{N_{t+u-v}} \right\} d_j(u) + \left\{ \prod_{v=0}^{T-1-t} \frac{b_j(y_{T-v})}{N_{T-v}} \right\} D_j(T-t)] \sum_{i \neq j} p_{ij} F_i(t) \end{aligned}$$

So, the above quantities, namely $F_j(t), B_j(t), L1_j(t)$ have been computed, they will be used to the expectation step of the EM algorithm.

E-step

Recall that the EM algorithm alternates two steps, the E-step which consists in calculating $Q(\lambda|\lambda^{\kappa})$ and the M-step which consists in choosing the next parameter value $\lambda^{(\kappa+1)}$ that maximizes $Q(\lambda|\lambda^{\kappa})$ over λ. In this step we have to express the expectations of the model parameters needed. This can be accomplished by using the expressions $F_j(t), B_j(t), L1_j(t)$ that computed above. So, we need to compute the expected number of times $\eta_{i,k}$ that the model remains in state i for k time steps, i.e:

$$\begin{aligned}\eta_{i,k} &= P(Z_{k+1} \neq i, Z_{k+1-v} = i, v = 1, \ldots, k | Y, \lambda) \\ &+ \sum_{t=1}^{T-1-k} P(Z_{t+k+1} \neq i, Z_{t+k-v} = i, v = 0, \ldots, k-1, Z_t \neq i | Y, \lambda)\end{aligned} \tag{4.15}$$

Analyzing the above quantity (4.15), the first term can be expressed for $v \leq T$ as :

$$\frac{L1_j(v)}{F_j(v)} \left\{ \prod_{u=1}^{v-1} \frac{b_j(y_{v-u})}{N_{v-u}} \right\} D_j(v) \pi_j$$

while for $v > T$ as:

$$\left\{ \prod_{u=1}^{T-1} \frac{b_j(y_{T-u})}{N_{T-u}} \right\} D_j(v) \pi_j$$

Moreover, the general term in equation (4.15) can be expressed for $v \leq T-1-t$ as :

$$\frac{L1_j(t+1+v)}{F_j(t+1+v)} \left\{ \prod_{u=1}^{v} \frac{b_j(y_{t+1+v-u})}{N_{t+1+v-u}} \right\} p_j(v) \sum_{i \neq j} \gamma_{ij} F_i(t)$$

and for $v > T-1-t$ as

$$\left\{ \prod_{u=0}^{T-1-t} \frac{b_j(y_{T-u})}{N_{T-u}} \right\} D_j(v) \sum_{i \neq j} \gamma_{i,j} F_i(t)$$

Now, the expected number of times that the model will remain in the state i is given by:

$$\eta_i = \sum_{k=0}^{T_{max(i)}} \eta_{i,k}$$

where $T_{max(i)}$ is the maximum sojourn time allowed in state i.

Definition 4.1. *The digamma function is defined as the logarithmic derivative of the gamma function that is*

$$\zeta(x) = \frac{\partial}{\partial x} \ln \Gamma(x) = \frac{\Gamma'(x)}{\Gamma(x)}$$

M-step

The second part of the EM algorithm consists of a re-estimation procedure, the M-step. This step determines the likelihood-increasing next set of parameters $\lambda^{(\kappa+1)}$ by

$$\lambda^{(\kappa+1)} = \arg\max_{\lambda} Q(\lambda | \lambda^{(\kappa)})$$

In Equation (4.4) we showed that the Q-function $Q(\lambda|\lambda^{(\kappa)})$ of a HSMM can be decomposed into four different terms, each depending on a given subset of λ. Hence, the re-estimation formulate for the parameters can be derived by maximizing each of the different terms separately. In our application we need to derive the re-estimation formula for emission distribution parameters and sojourn time parameters by the maximizing the terms (4.4) and (4.5).

The term of $Q - function$ given equation (4.4) and (4.15) treating gamma sojourn time distribution

$$Q_d(\{d_j(u)|\lambda^{\kappa}\}) = \sum_u \eta^{\kappa}_{j,u} \log d_j(u) \tag{4.16}$$

$$d_j(u) = \frac{\theta_j^{k_j}}{\Gamma(k_j)} u^{k_j-1} e^{-\theta u} \tag{4.17}$$

From (4.16) using the (4.17) arrive at

$$Q_d(d_j(u)|\lambda^{(\kappa)}) = \sum_u \eta^{(\kappa)}_{j,u}[\, k_j \log(\theta_j) - \log(\Gamma(k_j)) + (k_j-1)\log(u) - \theta_j u]$$

$$\frac{\partial Q_d(d_j(u)|\lambda^{(\kappa)})}{\partial k} = \sum_u \eta^{(\kappa)}_{j,u}[\, \log\theta_j - \frac{\Gamma'(k_j)}{\Gamma(k_j)} + \log u] \;= 0 \tag{4.18}$$

$$\frac{\partial Q_d(d_j(u)|\lambda^{(\kappa)})}{\partial \theta} = \sum_u \eta^{(\kappa)}_{j,u}[\, \frac{k_j}{\theta_j} - u] \;= 0 \Rightarrow \theta_j = \frac{k_j}{u} \tag{4.19}$$

$$(4.18) \xRightarrow{(4.19)} \sum_u \eta^{(\kappa)}_{j,u} \left[\log k_j - \frac{\Gamma'(k_j)}{\Gamma(k_j)} \right] = 0$$

where $\frac{\Gamma'(k)}{\Gamma(k)}$ is the digamma function as defined in the Definition 4.1 and k_j denotes the shape parameter. So the parameters are calculated using numerical methods.

As for the poisson sojourn time we have (4.16) and we have the following probability function

$$d_j(u) = \frac{e^{\theta}\theta_j^{u-d}}{(u-d)!}$$

$$Q_d(d_j(u)|\lambda^{\kappa}) = \sum_{u=1}^{T} \eta^{\kappa}_{j,u}[\, \theta_j + (u-d)\log\theta_j - \log[\,(u-d)!]\,]$$

$$\frac{\partial Q_d(d_j(u)|\lambda^\kappa)}{\partial \theta} = \sum_{u=1}^{T}\left[1 + \frac{u-d}{\theta_j}\right] = 0 \Rightarrow \hat{\theta}_j = \sum_{u=1}^{T}(u-d)\eta_{j,u}$$

for all possible shift parameters $d = 1, \ldots, \min(u : \eta_{j,u} > 0)$ choosing the d which gives the maximum likelihood.

Now, we continue with the emission probabilities parameters we have

$$b_j(y_t) = \frac{1}{\sqrt{2\pi\sigma_j^2}} \exp\left(-\frac{1}{2\sigma_j^2}(y_t - \mu_j)^2\right)$$

$$\begin{aligned}
Q_b(\{b_j(y_{[1:T]})\}|\lambda^{(\kappa)}) &= \sum_{t=1}^{T} L_j(t)\left[-\log(2\pi)^{-\frac{1}{2}} - \log\sigma_j - \frac{(y_t-\mu_j)^2}{2\sigma^2}\right] \\
\frac{\partial Q_b(\{b_j(y_{[1:T]}|\lambda^{(\kappa)})\})}{\partial \mu_j} &= \sum_{t=1}^{T} L_j(t)\left(\frac{(y_t-\mu_j)^2}{2\sigma^2}\right) = 0 \Rightarrow \\
&\sum_{t=1}^{T} L_j(t) y_t - \sum_{t=1}^{T} L_j(t)\mu_j = 0 \Rightarrow \\
\mu_j &= \frac{\sum_{t=1}^{T} L_j(t) y_t}{\sum_{t=1}^{T} L_j(t)}
\end{aligned}$$

We set $\theta_{2j} = \sigma_j^2$

$$\begin{aligned}
\frac{\partial Q(\{b_j(y_{[1:T]}|\lambda^{(\kappa)})\})}{\partial \theta_{2j}} &= \sum_{t=1}^{T} L_j(t)\left[-\frac{1}{2\theta_{2j}} + \frac{(y_t-\mu_j)^2}{2\theta_{2j}^2}\right] = 0 \Rightarrow \\
\frac{\sum_{t=1}^{T} L_j(t)}{2\theta_{2j}} &= \frac{\sum_{t=1}^{T} L_j(t)(y_t-\mu_j)^2}{2\theta_{2j}^2} \xrightarrow{\theta_{2j}>0} \\
\theta_{2j} &= \frac{\sum_{t=1}^{T} L_j(t)(y_t-\mu_j)^2}{\sum_{t=1}^{T} L_j(t)}
\end{aligned}$$

After each iteration of the algorithm the auxiliary function (see 4.5) is computed with the updated parameters of the M-step above. The parameter set that maximizes the quantity in (4.5) is obtained and was chosen as the optimal parameter set.

2.3 Decoding

Now we move forward to decoding problem. In order to deal with this problem and to find the most likely state sequence we use a dynamic programming method known as Viterbi algorithm.

Due to the fact that the state process is a semi-Markov chain, we have for all t

$$\begin{aligned}\max_{z_1,\ldots,z_T} P(Z_{[1:T]} = z_{[1:T]}, Y_{[1:T]} = y_{[1:T]}) = \max_{z_t}(&\max_{z_{t+1},\ldots,z_T} P(Y_{[t+1:T]} = y_{[t+1:T]}, \\ &Z_{[t+1:T]} = z_{[t+1:T]} | Z_{t+1} \neq z_t, Z_t = z_t) \\ &\times \max_{z_1,\ldots,z_{t-1}} P(Z_{t+1} \neq z_t, Z_{[1:t]} = z_{[1:t]}, Y_{[1:t]} = y_{[1:t]})) \end{aligned} \tag{4.20}$$

Following, it have been defined

$$\alpha_j(t) = \max_{z_1,\ldots,z_{t-1}} P(Z_{t+1} \neq j, Z_t = j, Z_{[1:t-1]} = z_{[1:t-1]}, Y_{[1:t]} = y_{[1:t]}) \tag{4.21}$$

Hence,decomposition (4.21) can be rewritten as

$$\begin{aligned}\max_{z_1,\ldots,z_T} P(Z_{[1:T]} = z_{[1:T]}, Y_{[1:T]} = y_{[1:T]}) = \max_{j}(&\max_{z_{t+1},\ldots,z_T} P(Y_{[t+1:T]} = y_{[t+1:T]}, \\ &Z_{t+1:T} = z_{t+1:T} | Z_{t+1} \neq j, Z_t = j) \alpha_j(t)) \end{aligned} \tag{4.22}$$

So,using this decomposition can be built the following recursion $t = 1, \ldots, T$; $j = 1, \ldots, J$

$$\begin{aligned}\alpha_j(t) &= \max_{z_1,\ldots,z_T} P(Z_{t+1} \neq j, Z_t = j, Z_{[1:t-1]} = z_{[1:t-1]}, Y_{[1:t]} = y_{[1:t]]}) \\ &= b_j(y_t) \max \left(\max_{1 \leq u \leq t} \left(\left\{ \prod_{v=1}^{u-1} b_j(x_{t-v}) d_j(u) \right\} \max_{i \neq j} \{ p_{ij} \alpha_i(t-u) \} \right) \left\{ \prod_{v=1}^{t} b_j(x_{t-v}) \right\} d_j(t+1) \right) \end{aligned} \tag{4.23}$$

The right-censoring of the sojourn time in the last visited state makes particular the case $t = T$, $j = 1, \ldots, 4$

$$\begin{aligned}\alpha_j(T) &= \max_{z_1,\ldots,z_{T-1}} P(Z_t = j, Z_{[1:T-1]} = z_{[1:T-1]}, Y_{[1:T]} = y_{[1:T]}) \\ &= b_j(y_T) \max[\max_{1 \leq u \leq T} [\left\{ \prod_{v=1}^{u-1} b_j(y_{T-v}) \right\} \\ &\times D_j(u) \max_{i \neq j} \{ p_{ij} \alpha_i(T-u) \}], \left\{ \prod_{v=1}^{T} b_j(x_{T-v}) \right\} D_j(T+1) \pi_j] \end{aligned} \tag{4.24}$$

The likelihood of the optimal state sequence associated with the observed sequence $y_{[1:T]}$ is $\max_j \{\alpha_j(T)\}$. The Viterbi recursion is the equivalent in terms of dynamic programming of the forward recursion (summation in (4.6) and (4.7)) replaced by maximization in (4.23) and (4.24).

Therefore, the proposals made for an efficient implementation of the forward recursion can be directly transposed to the Viterbi algorithm. For each time t and each state j, two backpointers can be recorded, the first giving the optimal preceding state and the second the optimal preceding time of transition from this preceding state. These backpointers can be used in a second stage-often referred to as "backtracking"-to retrieve the optimal state sequence. The backtracking procedure consists in tracing backward along the couple of backpointers from the optimal final state (at time T) to the optimal initial state (at time 1).

3 Metrics of performance

The evaluation of segmentation methods is based on their ability to accurately identify the fundamental heart sound S_1 and S_2. The labeling of these sounds was made as described in section 2.7. The performance of the segmentation algorithms were evaluated using the F_1 score, which is defined as

$$F_1 = \frac{2 \times P_+ \times S_e}{P_+ + S_e}$$

where S_e is sensitivity or recall and P_+ is positive predictive value (PPV) or precision. In order to explain each of these measures we should first define the following notions:

- TP (True Positive) is an outcome where the model correctly identifies S_1 and S_2 sounds. An S_1 sound was labeled as correctly allocated if the start of the predicted S_1 sound was found to be within 100ms (at sampling frequency 1000Hz) of the R-peak of the ECG. Correspondingly, an S_2 sound was found to be within 100ms (at sampling frequency 1000Hz) of the respective end-T-wave. This tolerance was necessary since it appears in ECG R-peak detection.

- TN (True Negative) is an outcome where the model correctly predicts the non-existence of the S_1 (or S_2) sound.

- FP(False Positive) is an error where the model incorrectly indicates the existence of S_1 (or S_2) sound.

- FN (False Negative) is an error where the model does not indicate S_1 (or S_2) sound when in reality it is.

As a result, the precision an the sensitivity can be respectively defined by:

$$P_+ = \frac{TP}{TP + FP}$$

$$S_e = \frac{TP}{TP + FN}$$

but considered as insufficient since no true negatives are included. However, it is used for comparison purposes. The process of halving the data into training and test set and extracting the results was reproduced 100 times in order to increase the reliability of the model and the final results are the averages over all iterations. The proposed procedure is represented in figure(flowchart).

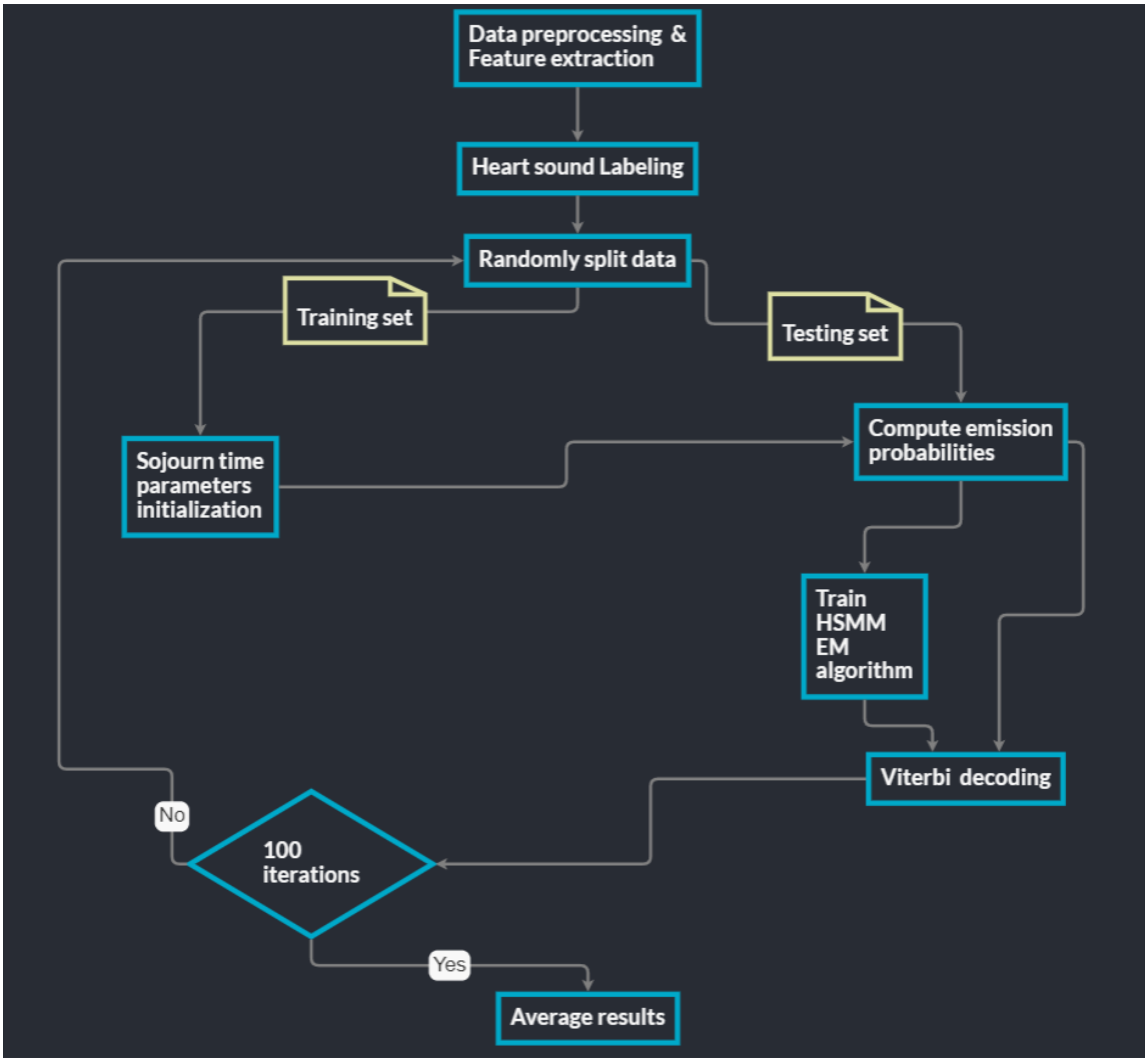

FIGURE 4.2: Flowchart of the proposed procedure.

4 Results

Inputs		Metrics of perfomance			
$d_j(u)$	Signal	Prec	Rec	Acc	Fscore
Poisson	Hom	87,20±0,52	93,97±0,54	84,27±0,51	91,67±0,33
	Hilb	85,20±0,52	93,09±0,53	81,69±0,57	90,01±0,33
	Psd	80,85±0,37	91,61±0,40	76,47±0,36	86,12±0,45
	Orig	83,87±0,55	91,15±0,59	79,29±0,61	88,59±0,35
Gamma	Hom	84,25±0,56	91,38±0,60	80,26±0,63	88,79±0,44
	Hilb	79,55±0,54	89,75±0,66	74,80±0,63	84,68±0,48
	Psd	79,14±0,52	88,52±0,43	75,49±0,55	85,43±0,37
	Orig	84,67±0,49	90,26±0,38	78,81±0,49	87,43±0,36

The gross performance results of the algorithms under consideration on both the training and test sets, using the different features, are presented in Table(). This table illustrates the scores for combined S_1 and S_2 sounds. These gross scores were calculated on a per patient basis, summing the total number of sounds for each patient in the training and testing set and calculating the different metrics for each patient. The results over the 100 iterations were then averaged over patients in both the training and testing sets. The standard deviation of the results over the 100 evaluation iterations is also shown. As we can see, the highest F-score (last column) is achieved by using an input signal the Homomorphic envelope results in a smooth envelope that helps enable easy peak detection; additionally, it efficiently removes the effects of murmurs. Peak conditioning was performed to remove peaks that dis not correspond to S_1 to S_2. Also the results show that sojourn time distribution Poisson achieves better results than Gamma sojourn time distribution. Finally, we can see that the other features, not only they cannot accomplish higher scores than Homomorphic envelope but also they fail to improve the performance of the original signal. Moreover, the small values of the standard deviations indicate that there is relatively small variability of the F-scores with respect to the choice of the training and the test set, which is induced by the random split of the data. And last but not least, the qualitative interpretation of the results related to the performance of each feature and the selected sojourn time distributions given above for F-score, remains unchanged for the other metrics of performance. However, the results indicate that the Homomorphic envelope performs the best in the metric of sensitivity , then in the Precision and lastly at the Accuracy.

Bibliography

[1] Shun-Zheng Yu. Hidden Semi-Markov models: theory, algorithms and applications. Morgan Kaufmann, 2015.

[2] Wang, Ning; Sun, Shu-dong; Cai, Zhi-qiang; Zhang, Shuai; Saygin, Can (2014). A Hidden Semi-Markov Model with Duration-Dependent State Transition Probabilities for Prognostics. Mathematical Problems in Engineering, 2014(), 1–10. doi:10.1155/2014/632702

[3] Clemens Valens. A really friendly guide to wavelets. ed. Clemens Valens, 1999.

[4] John Van Der Hoek, Robert J. Elliott.Introduction to Hidden Semi-Markov Models, 2018.

[5] Jan Bulla. Application of Hidden Markov Models and Hidden Semi-Markov Models to Financial Time Series,pages 57-105,2006.

[6] Lawrence R Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. Proceedings of the IEEE, 77(2):257–286, 1989.

[7] Serafim Batzoglou,Lecture 6: Hidden Markov Models Continued,pages 1-4.

[8] Vlad Stefan Barbu,Nikolaos Limnios.Semi-Markov Chains and Hidden Semi-Markov Models toward Applications:Their Use in Reliability and DNA Analysis,pages 43-73, 2008.

[9] Vlad Stefan Barbu, Jan Bulla, Antonello Maruotti. Estimation of the stationary distribution of a semi-Markov chain. Journal of Reliability and Statistical Studies; ISSN (Print): 0974-8024, (Online):2229-5666 Vol. 5, Issue Special (2012): 15-26.

[10] Geoffrey J. McLachlan, Thriyambakam Krishnan. The EM Algorithm and Extensions , second edition, 2008.

[11] Leon Gu,EM and HMM.

[12] Maya R. Gupta, Yihua Chen. Theory and Use of the EM Algorithm,Vol. 4, No. 3 (2010) 223–296, 2011.

[13] Neal, R. M., & Hinton, G. E. (1998). A View of the Em Algorithm that Justifies Incremental, Sparse, and other Variants. Learning in Graphical Models, 355–368.

[14] Ajit Singh. The EM Algorithm, pages 1-4. 2005.

[15] Cardiovascular Disease,`https://my.clevelandclinic.org/health/diseases/21493-cardiovascular-disease`

[16] Anatomy of the Human Heart,`https://www.physio-pedia.com/Anatomy_of_the_Human_Heart`

[17] Cardiac cycle, `https://en.wikipedia.org/wiki/Cardiac_cycle`

[18] Heart, `https://www.britannica.com/science/heart`

[19] Gill, D.; Gavrieli, N.; Intrator, N. (2005). [IEEE Computers in Cardiology, 2005 - Lyon, France (2005.09.25-2005.09.28)] Computers in Cardiology, 2005 - Detection and identification of heart sounds using homomorphic envelogram and self-organizing probabilistic model. , (), 957–960.

[20] Ming Dong; David He (2007). A segmental hidden semi-Markov model (HSMM)-based diagnostics and prognostics framework and methodology. , 21(5), 2248–2266.

[21] Joachim Behar, Julien Oster, Qiao Li, and Gari D Clifford. Ecg signal quality during arrhythmia and its application to false alarm reduction. IEEE transactions on biomedical engineering, 60(6):1660–1666, 2013.

[22] YJ Chung. Pattern recognition and image analysis, iberian conference. In ch. Classification of Continuous Heart Sound Signals Using the Ergodic Hidden Markov Model, pages 563–570. Springer Berlin Heidelberg, 2007.

[23] Seyedeh Zahra Fatemian. A wavelet-based approach to electrocardiogram (ECG) and phonocardiogram (PCG) subject recognition. PhD thesis, 2009.

[24] Daniel Gill, Noam Gavrieli, and Nathan Intrator. Detection and identification of heart sounds using homomorphic envelogram and self-organizing probabilistic model. In Computers in Cardiology, 2005, pages 957–960. IEEE, 2005.

[25] Ricardo Gutierrez-Osuna. Introduction to speech processing. CSE@ TAMU, 2016.

[26] Isabelle Guyon and André Elisseeff. An introduction to feature extraction. In Feature extraction, pages 1–25. Springer, 2006.

[27] Nicholas Peter Hughes and H Term. Probabilistic models for automated ECG interval analysis. PhD thesis, University of Oxford, 2006.

[28] Jared O'Connell, Søren Højsgaard, et al. Hidden semi markov models for multiple observation sequences: The mhsmm package for r. Journal of Statistical Software, 39(4):1–22, 2011.

[29] W Phanphaisarn, A Roeksabutr, P Wardkein, J Koseeyaporn, & PP Yupapin. Heart detection and diagnosis based on ecg and epcg relationships. Medical devices (Auckland, NZ), 4:133, 2011.

[30] Iead Rezek and Stephen J Roberts. Envelope extraction via complex homomorphic filtering. Technical Report TR-98-9 Technical report, 1998.

[31] David B Springer, Lionel Tarassenko, and Gari D Clifford. Support vector machine hidden semi-markov model-based heart sound segmentation. In Computing in Cardiology 2014, pages 625–628. IEEE, 2014.

[32] David B Springer, Lionel Tarassenko, and Gari D Clifford. Logistic regression-hsmm-based heart sound segmentation. IEEE Transactions on Biomedical Engineering, 63(4):822–832, 2015.

[33] Shuping Sun, Zhongwei Jiang, Haibin Wang, and Yu Fang. 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):219–230, 2014.

[34] Harun Uğuz, Ahmet Arslan, and İbrahim Türkoğlu. A biomedical system based on hidden markov model for diagnosis of the heart valve diseases. Pattern recognition letters, 28(4):395–404, 2007.

[35] Shun-Zheng Yu. Hidden semi-markov models. Artificial intelligence, 174(2):215–243, 2010.

[36] Github, https://github.com/davidspringer/Springer-Segmentation-Code?fbclid=IwAR0BVmDZiSX2aWAdw5BAwAnYH4iCFE5pPV3zJAkerQOu5NiYuFC9lx1crdw

[37] National Library of Medicine, https://www.ncbi.nlm.nih.gov/books/NBK535419/?fbclid=IwAR23JqlSJOmWa5O199uguiB2vzB6QyAQtK2hFmrkoQzn7AaKdB7bEwG8FY

[38] Mubarak, Q.-A., Akram, M. U., Shaukat, A., Hussain, F., Khawaja, S. G., and 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, 143–157.

[39] Hassani, K., Bajelani, K., Navidbakhsh, M., Doyle, D., and Taherian, F. (2014). Heart sound segmentation based on homomorphic filtering. Perfusion, 29(4), 351–359.

[40] Fear Cat, https://blog.fearcat.in/a?ID=00650-f9f096b0-31b6-4dcc-8313-f619b9e087b2

[41] https://en.wikipedia.org/wiki/Feature_extraction

[42] Reality AI `https://reality.ai/it-is-all-about-the-features/`

[43] Oliveira, J. H., Renna, F., Mantadelis, T., & Coimbra, M. T. (2018). Adaptive Sojourn Time HSMM for Heart Sound Segmentation. IEEE Journal of Biomedical and Health Informatics, 1–1.

[44] Gill, D., Gavrieli, N., & Intrator, N. (2005). Detection and identification of heart sounds using homomorphic envelogram and self-organizing probabilistic model. Computers in Cardiology, 2005.

[45] Henrici Peter, Applied and computational complex analysis, Volume 1, John Wiley & Sons, Inc., New York, 1988.

[46] Xu J, Durand LG, Pibarot P. Nonlinear transient chirp signal modeling of the aortic and pulmonary components of the second heart sound, IEEE Trans. Biomed. Eng. 2000; 47:1328-1335.