Universidad del País Vasco Euskal Herriko Unibertsitatea
BILBOKO INGENIARITZA ESKOLA
ESCUELA DE INGENIERÍA DE BILBAO

BACHELOR DEGREE IN TELECOMUNICATION ENGINEERING

BACHELOR'S FINAL DEGREE PROJECT

AN ALGORITHM TO DETECT ATRIAL FIBRILATION USING SHORT ECG SEGMENTS

Student: Asurmendi, Perez, Iratxe

Manager: Irusta, Zarandona, Unai

Academic year: 2018-2019

Date: Eibar, 22, July, 2019

Universidad del País Vasco Euskal Herriko Unibertsitatea | BILBOKO INGENIARITZA ESKOLA ESCUELA DE INGENIERÍA DE BILBAO

Table of Contents

Universidad del País Vasco Euskal Herriko Unibertsitatea

BILBOKO INGENIARITZA ESKOLA
ESCUELA DE INGENIERÍA DE BILBAO

Trilingual Summary

Summary

Atrial fibrillation is a difficult disease to detect until it manifests in a serious way. However, if it is detected early enough, treatment can be applied so that symptoms are controlled and do not become deadly.

The objective of this project is to develop software that allows the detection of atrial fibrillations in short-term electrocardiograms. This would allow early detection in electrocardiograms performed in health centers and hospitals, without having to make a long study of cardiac activity.

To carry out this work we will work with the concept of Deep Learning. Before that, an automatic heartbeat detector is implemented, which must be very accurate in detecting the peak of the R wave in the QRS complex. Then, a neural network is created that will be responsible for differentiating between atrial fibrillations and sinus rhythms. Finally, the results are evaluated and considered the various benefits that will have its implementation.

Resumen

La fibrilación atrial es una enfermedad difícil de detectar hasta que se manifiesta de forma seria. Sin embargo, si se detecta con suficiente tiempo se puede aplicar un tratamiento para que sus síntomas estén controlados y no llegue a ser mortal.

El objetivo de este proyecto es desarrollar un software que permita la detección de fibrilaciones atriales en electrocardiogramas de corta duración. Esto posibilitaría una temprana detección en electrocardiogramas realizados en centros de salud y hospitales, sin tener que hacer un estudio largo de la actividad cardiaca.

Para llevar a cabo este trabajo se trabajara con el concepto de Deep Learning. Antes de ello, se implementa un detector de latidos automático, el cual debe ser muy preciso en la detección del pico de la onda R en el complejo QRS. A continuación se crea una red neuronal que será la encargada de diferenciar entre fibrilaciones atriales y ritmos sinusales. Por último, se evalúan los resultados obtenidos y se plantean los diferentes beneficios que supondría su implementación.

Laburpena

Fibrilazio aurikularra gaixotasun zaila da hautemateko, modu larrian agertzen den arte. Hala ere, denbora nahikorekin hautematen bada, sintomak kontrolatzeko tratamendua jar daiteke eta ez da hilgarri izango.

Proiektu honen helburua epe laburreko elektrokardiogrametan fibrilazio aurikularrak detektatzeko softwarea garatzea da. Modu honetan osasun zentroetan eta ospitaleetan burutu diren elektrokardiogrametan fibrilazio aurikularrak goiz detektatzea ahalbidetuko litzateke, bihotz-jardueraren azterketa luzea egin beharik gabe.

Lan hau burutzeko Deep Learning kontzeptua lantzen dugu. Honen aurretik, bihotz taupaden detektore automatiko bat jarri da. Detektorea, oso zehatza izan behar du QRS konplexuko R uhinaren gailurra hautemateko. Ondoren, fibrilazio atrialen eta erritmo normalen artean bereizteko ardura duen neurona-sarea sortuko da. Azkenik, lortutako emaitzak ebaluatzen dira eta metodoaren aplikazioak dituen onura ezberdinak aztertuko dira.

Table of Figures

Table of tables

1. Introduction

This is a project developed within the Bioengineering and Resuscitation (BioRes) research group of the Superior Technical School of Engineering of Bilbao by an undergraduate student in order to get to know the work done in the mentioned research group. The BioRes group's research is focused on the application of digital signal processing techniques in biomedical signals, with strong applications to treatment of cardiac arrest. One of the fields of expertise of the group is the detection of cardiac arrhythmia.

In particular, the work to be carried out has been part of the challenge launched in 2017 by the computing in cardiology conference. The computing in cardiology conference is an IEEE sponsored conference focused on the application of computer methods to the field of cardiology.

The basis of this project is the processing of biomedical signals, and in particular of the electrocardiogram (ECG) signal, in order to detect atrial fibrillation in short ECGs.

Atrial fibrillation is the most common cardiac arrhythmia, and its prevalence increases with age. Although atrial fibrillation is not an immediately life threatening arrhythmia, it can cause cloths that can produce heart attacks or strokes. It is therefore a serious medical condition. Despite not being a life threatening arrhythmia itself, it is essential to have control of it, as it can be asymptomatic - and therefore difficult to detect - and lead to serious medical problems.

Therefore, this project will focus on developing a low cost algorithm for the detection of atrial fibrillation using the ECG, so that its detection is simpler and more efficient (both in time and in resources).

2. Context

2.1 Medical relevance of atrial fibrillation

Atrial fibrillation is the most common sustained cardiac arrhythmia in the general population. In Spain, the latest data suggest that the prevalence in people aged over 40 years could be greater than 4%. Both the prevalence and incidence of atrial fibrillation have increased substantially over time. Possible reasons for this rise are the aging of the population, an increase in the number of cardiovascular risk factors and better cardiovascular disease survival. In addition, the increased availability of improved diagnostic tools for detecting atrial fibrillation could have contributed [1].

In the decade of 2000-2010, the number of hospitalizations associated with atrial fibrillation in the United States increased by 14-23%, with approximately 3 million hospitalized patients, which translates into an increase in the costs of hospital care. . Strikingly, the largest increase occurred in the group of patients between 35 and 49 years old, with a relative increase of 16.6%. In this same period of time, in Colombia there was an increase of 10.4% in the number of new cases of atrial fibrillation, for a total of 1,995 attributable deaths [2].

For patients who are hospitalized, mortality is close to 1%, but it can be up to 8% in patients with concomitant heart failure and 9.4% in patients with haemorrhagic cerebrovascular attack [2].

The costs of atrial fibrillation are not distributed uniformly, but suffer changes throughout the patient's life: in the month following the diagnosis and in the month preceding the death, most of the costs are related with hospital care, while in the chronic stage the costs are related to medical visits, consultations for emergencies and hospitalizations for acute decompensation [2].

Similarly, there are patient characteristics that predict costs: in the chronic phase, the presence of comorbidities (diabetes, heart failure, kidney or liver failure) and advanced age predict a higher cost of care; in the month preceding death, advanced age decreases the total cost of care. In general terms, with the increase in hospitalization rates, there was also a relative increase in hospitalization costs of 24%, which represent the greatest expense in the management of patients with atrial fibrillation [2].

Thus, the reduction of these cases can occur with early detection of the disease, with a simple electrocardiogram provided by a low-cost software tool such as the one proposed in this paper.

2.2 What is atrial fibrillation.

Atrial fibrillation is an irregular and often rapid heart rate that occurs when the two upper chambers of the heart, the atria, contract chaotically and irregularly [3]. Atrial activity is then decoupled from the activity of the lower chambers of the heart, the ventricles, producing an irregular ECG before the actual heartbeat. AF is defined as a “tachyarrhythmia characterized by predominantly uncoordinated atrial activation with consequent deterioration of atrial mechanical function” by the American College of Cardiology (ACC), the American Heart Association (AHA) and the European Society of Cardiology (ESC) [4].

The sinoatrial (SA) node is a group of cells located in the right atrium, and it is the natural pacemaker of the heat. It produces the impulse that normally starts each heartbeat [3]. Normally, the impulse travels first through the atria and then through a connecting pathway between the upper and lower chambers of your heart called the atrioventricular (AV) node. As the signal passes from the sinus node through the atria, they contract, pumping blood from your atria into the ventricles below. As the signal passes through the AV node to the ventricles, it signals the ventricles to contract, pumping blood out to your body [3].

In atrial fibrillation, there are many electrical re-entry pathways in the atria that result in the impulse not being conducted from the SA node to the AV node.. As a result, they quiver. The AV node is bombarded with impulses trying to get through to the ventricles. . This produces an irregular electrical activity before the heartbeats, and an irregular heart rate since it is hard to predict when an atrial impulse will be conducted to the ventricles. The ventricles also beat rapidly, but not as rapidly as the atria, as not all the impulses get through. The reason is that the AV node is like a highway on-ramp — only so many vehicles can get on at one time [3]. The differences in electrical conduction between a normal rhythm and atrial fibrillation are shown in Figure 1.

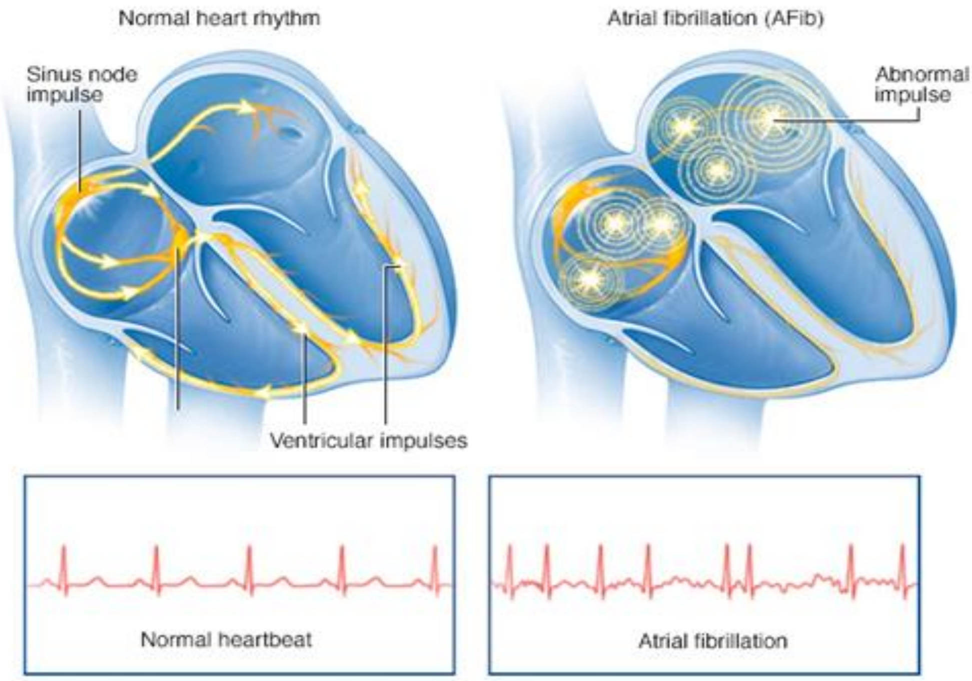


Figure 1. The differences between a normal rhythm and atrial fibrillation.

The result is a fast and irregular heart rhythm. The heart rate in atrial fibrillation may range from 100 to 175 beats per minute. The normal range for a heart rate is 60 to 100 beats per minute [3].

Episodes of atrial fibrillation can come and go, or you may develop atrial fibrillation that doesn't go away and may require treatment. Although atrial fibrillation itself usually isn't life-threatening, it is a serious medical condition that sometimes requires emergency treatment [3]. Some people with atrial fibrillation have no symptoms and are unaware of their condition until it's discovered during a physical examination.

Possible causes of atrial fibrillation

There are many possible causes of atrial fibrillation. The most common are abnormalities or damage to the heart's structure. Other possible causes include: high blood pressure, coronary artery disease, abnormalities in heart valves, congenital defects, exposure to stimulants, sick sinus syndrome or malfunction of the SA node, etc....

Patients who suffer atrial fibrillation symptoms may experience signs and symptoms such as: palpitations, weakness, reduced cardiac output, fatigue, dizziness, chest pain, softness of breath.

Atrial fibrillation may be:

- **Occasional**. In this case it's called paroxysmal (par-ok-SIZ-mul) atrial fibrillation. You may have symptoms that come and go, lasting for a few minutes to hours and then stopping on their own.
- **Persistent**. With this type of atrial fibrillation, your heart rhythm doesn't go back to normal on its own. If you have persistent atrial fibrillation, you'll need treatment such as an electrical shock or medications in order to restore your heart rhythm.
- **Long-standing persistent**. This type of atrial fibrillation is continuous and lasts longer than 12 months.
- **Permanent.** In this type of atrial fibrillation, the abnormal heart rhythm can't be restored. You'll have atrial fibrillation permanently, and you'll often require medications to control your heart rate.

Complications

Although it is not immediately life-threatening, atrial fibrillation may lead to several medical complications. Atrial fibrillation may produce blood clots forming in the heart that may circulate to other organs and lead to blocked blood flow (ischemia). The most severe complications of atrial fibrillation are:

- **Stroke.** In atrial fibrillation, the chaotic rhythm may cause blood to pool in your heart's upper chambers (atria) and form clots. If a blood clot forms, it could dislodge from your heart and travel to your brain. There it might block blood flow, causing a stroke. The risk of a stroke in atrial fibrillation increases with age, high blood pressure, diabetes, a history of heart failure or previous stroke, and other factors. Certain medications, such as blood thinners, can greatly lower your risk of a stroke or the damage to other organs caused by blood clots.

- **Heart failure.** Atrial fibrillation, especially if not controlled, may weaken the heart and lead to heart failure — a condition in which your heart can't circulate enough blood to meet your body's needs.

However, some people who have atrial fibrillation don't have any heart defects or damage, a condition called lone atrial fibrillation. In lone atrial fibrillation, the cause is often unclear, and serious complications are rare.

Low cost diagnostic tools

Since atrial fibrillation is much extended and does not need to be immediately treated, a low cost general purpose tool for its detection would be of great use. That is why many methods have focused on the use of the ECG to detect atrial fibrillation, since the ECG can be very simply acquired using two leads placed on the surface of the patient. As shown in Figure 2 the ECGs during atrial fibrillation and normal rhythms are very different.

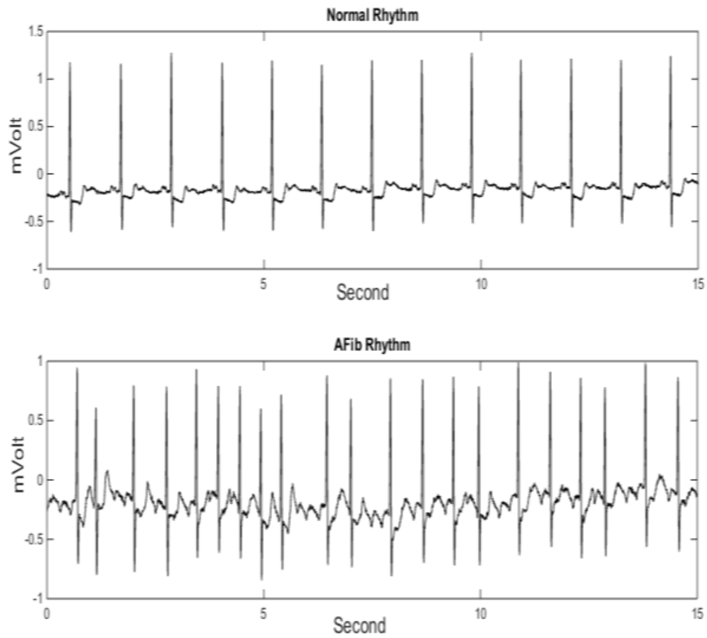


Figure 2. ECGs during atrial fibrillation and normal rhythms.

Consequently, in this project we will focus on the development of ECG based low cost tools for the detection of atrial fibrillation.

3. Scope/Reach

The main aim of this project is to create an automatic algorithm to identify atrial fibrillation using the ECG recorded between two electrodes in the body of the subject.To fulfil this objective, several intermediate objectives need to be completed, namely:

1. **Definition of the database**
   The database contains 8,528 single-lead ECG recordings lasting from 9s to just over 60s. ECG recordings were sampled as 300 Hz.

2. **Development of a tool to visualize**
   In order to carry out the project properly, it is necessary to develop GUIs in order to visualize all the signals with which it's been working, their annotations, intermediate results and final results. The support of these graphical user interfaces will be fundamental at the time of finalizing the project successfully.

3. **Integration of ECG processing tools. This includes filtering to remove interferences and signal processing algorithms to detect heartbeats.**
   A pre-processing of the ECG signal is necessary, where it is especially important to delete interferences such as baseline or low frequency oscillations.

4. **Development of an automatic classification algorithm**
   Based on the Machine Learning concept, we parameterize the different electrocardiograms to obtain their parameters of the different types of signals and, later, based on these parameters, we can classify them. Based on the parameters we have obtained with this algorithm, we can compare the parameters of a new signal with those already learned and thus be able to classify what type of electrocardiogram it is.

5. **Characterization of the algorithm**
   Once the algorithm has learned to classify the different signals, the effectiveness of the algorithm is checked. It is done with other signals that have been used to learn. For this, in our case, we check the parameters Sp and Se.

4. Benefits of the project

An electrocardiogram is a very common test in different clinical scenarios. Although the detection of atrial fibrillation is not something that is usually sought in the realization of an ECG, having a tool that detects it without having to look for it, could help medical teams. Therefore, with the development of this algorithm, the project will have different benefits: technical, economic and social.

4.1 Technical benefits

The completion of this project will produce several important technical benefits. First, the BioRes group will start work on a very active and demanding research field, namely ECG based detection and characterization of atrial fibrillation. Second, a simple algorithm based on machine learning will be developed deepening the knowledge of BioRes in machine learning. Third, a low cost easily deployable solution will be developed that can be integrated into current ECG monitoring equipment adding new functionality to automatically detect atrial fibrillation.

4.2 Economic benefits

The development of this project will result in economic benefits for public health. On the one hand, the low cost of the techniques used for the detection of AF from the ECG signal, that is, it is a software solution that does not require additional investment. In addition, and despite being a low-cost solution, it offers an improvement in technical performance, since it will be saved in subsequent medical tests.

4.3 Social benefits

Atrial fibrillation is a health compromising condition that affects large parts of the population. The development of automatic algorithms for its early detection and diagnosis can improve its treatment and lead to important health benefits to the patients. Those benefits include a better quality of life and the prevention of possible serious or life threating conditions like heart attacks or stroke.

5. State of art

5.1 The Electrocardiogram (ECG)

Low cost non-invasive diagnostic techniques are key to generalize treatment. The electrocardiogram (ECG) is the most important non-invasive technique to monitor the heart and its condition. In most surgical procedures, an ECG is usually requested previously [5].

The ECG is the electrical activity of the heart recorded by two electrodes in the surface of the body. The electrical activity of the heart controls its mechanical activity and thus pumping of the blood. It is used routinely in the initial diagnosis of many cardiac diseases [6]. The primary function of the heart is to pump oxygen-rich blood throughout the body. The heart is composed of muscle cells, which produce the contraction; these cells are connected into a network (conduction system) that allows an electrical impulse spread throughout the heart. The cardiac cycle starts in the atria and goes down throw the ventricles, so the impulse is triggered in the atria and it precedes the heart contraction [1]. Each cardiac cycle is composed of two stages, contraction and relaxation, called in electrical terms depolarization and repolarization [7].

The electrical activity of the heart in the ECG is observed in the form of a path that presents different deflections (waves of the ECG) that correspond to the path of the electrical impulses through the different structures of the heart. To understand the basic principles that explain the oscillations it is convenient to know the fundamentals by which the movement of the heart takes place [5]. The heartbeat produces a series of waves with a time-variant morphology. These waves are caused by voltage variations of the cardiac cells [7].

P Wave

The P wave is the first wave of the cardiac cycle. It represents the depolarization of the atria. It is composed of the superposition of the electrical activity of both atria. Its initial part corresponds to the depolarization of the right atrium and its final part to that of the left atrium. When it is generated by the sinus node, it is positive in all derivations. In atrial growths, the P wave may increase in height or duration, and is absent in atrial fibrillation.

QRS complex

It is formed by a set of waves that represent the depolarization of the ventricles. It takes several morphologies depending on the derivation.

- Q Wave: if the first wave of the QRS complex is negative, it is called the Q wave.
- R Wave: is the first positive wave of the QRS complex, it can be preceded by a negative wave (Q wave) or not. If there is another positive wave in the QRS complex, it is called R '.
- S Wave: is the negative wave that appears after the R wave.

T Wave

It represents the repolarization of the ventricles. It is generally of smaller amplitude than the QRS that precedes it. In a normal electrocardiogram, it is positive in all leads. The normal T wave is asymmetric, with the ascending portion slower than the descending one. There are multiple pathologies that cause changes in the T wave.

A representation of a sinusal (normal) ECG can be seen in Figure 3.

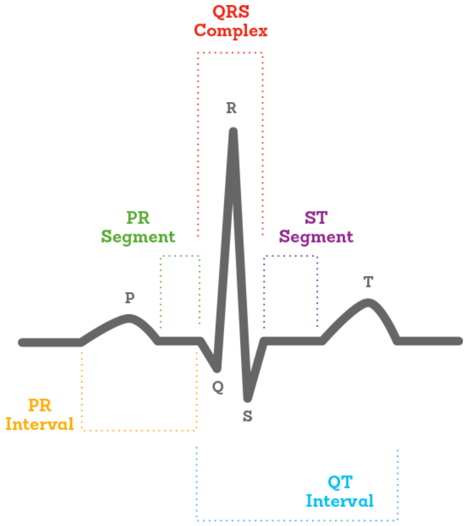


Figure 3. Normal ECG graph

It is a simple test, available, quick, that does not produce any discomfort (it is painless) and does not have any risk for the patient (no electricity is sent through the body, it only detects the electrical activity that is generated in the patient's own heart) [8]. Performing an ECG is a simple procedure. You need an electrocardiograph and ECG patches. The electrocardiogram sensors on the skin, behaving as if they were electrodes, and a system of wires that transmit the microcurrents collected by the patches to the electrocardiograph, which will be responsible for amplifying them [5].

To record an ECG, it is necessary to place a series of electrodes on the patient's skin, which will be attached to the electrocardiograph by wires [8]. With a total of 10 patches (electrodes) it will normally be sufficient [5]. The points where the electrodes are placed are: ankles, wrists and chest (precordial region). The same electrical impulse is collected from different positions [8]. Thus forming the six named derivations of the members. An electrocardiographic derivation is constituted by the union of two electrodes. Each one allows obtaining a different electrocardiographic view, representing 12 different windows or observation points. An anomaly that may not be noticed from one derivation (window) and from another. Once the patient is lying down and with the 10 wires that connect the ECG with its corresponding patch, the ECG record can be started [5]. Wires are placed as in Figure 4.

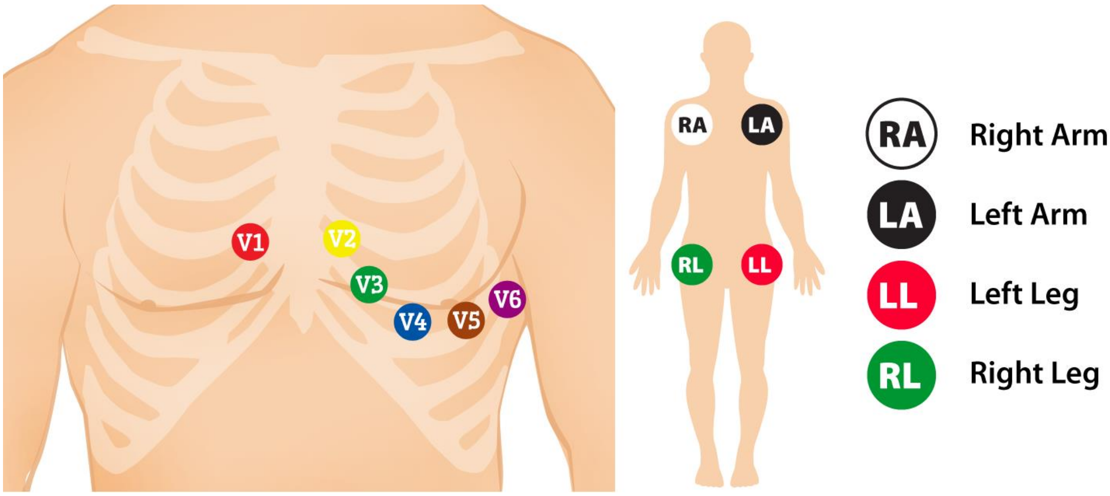


Figure 4. Wire placing for an electrocardiogram.

By using 4 limb electrodes, you get 6 frontal leads that provide information about the heart's vertical plane:

- Lead I
- Lead II
- Lead III
- Augmented Vector Right (aVR)
- Augmented Vector Left (aVL)
- Augmented vector foot (aVF)

Leads I, II, and III require a negative and positive electrode (bipolarity) for monitoring. On the other hand, the augmented leads-aVR, aVL, and aVF-are unipolar and requires only a positive electrode for monitoring. Shown in Figure 5.

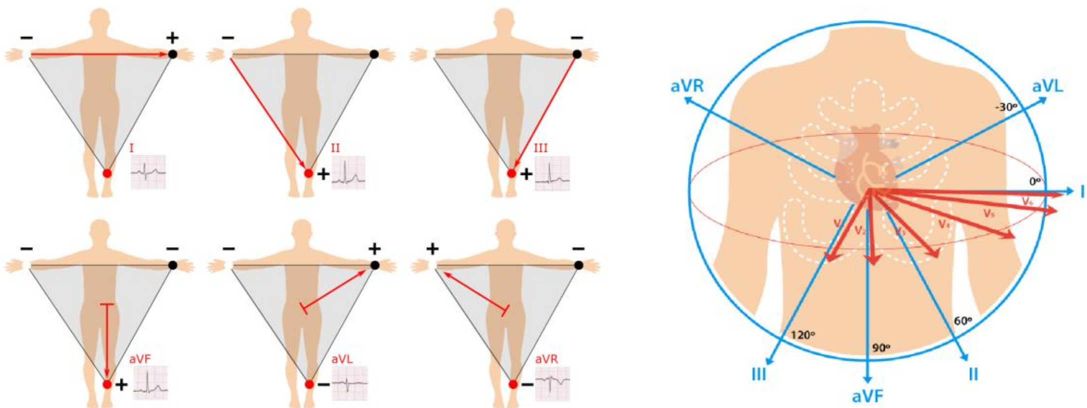


Figure 5. Heart's vertical plane

There are other ways to perform an ECG apart from the conventional at rest. Fundamentally there are two: the effort ECG and the Holter-ECG. The effort ECG consists of walking on an endless treadmill or pedaling on a specially adapted bicycle, while the doctor assesses the ECG performed during the exercise, as well as if the patient presents some type of discomfort or pain during the test. It is used primarily for the diagnosis and monitoring of coronary disease (obstruction of the coronary arteries). In the Holter-ECG, the patient's ECG is recorded through a specially designed recording system for an approximate time of 24 hours; subsequently, it is analyzed by specific software. It is mainly used for the study of arrhythmias [5].

The electrocardiogram provides important and relevant information about the heart state. It is being used to diagnose cardiac diseases, which are the main cause of mortality in our society. Thus, using the power of computers, we can detect heart diseases or anomalies that otherwise could only be detected by experts physicians [7]. The ECG of a healthy person has a characteristic pattern and the changes that occur in the pattern of normality of the ECG are usually associated with heart diseases [5].

The ECG is fundamentally used to detect heart rhythm disorders (arrhythmias) and in the diagnosis of situations that present with an insufficient blood supply to the heart (myocardial infarction and angina pectoris). The ECG makes it possible to differentiate the normal rhythm of the heart (sinus rhythm) from any type of tachycardia -rhythms in which the heart beats at an abnormally fast rate (100-300 beats per minute). In the opposite direction, it is the simplest method to objectify the slow rhythms, in which the heart rate decreases below a lower limit considered as normal, which is accepted between 55-60 beats per minute. Below this frequency we speak of bradycardia. Likewise, the ECG is the method of choice in the diagnosis of cardiac blocks. However, the most frequent pathological arrhythmia is atrial fibrillation. In this situation, the atria beat acceleratedly more than three hundred times in a minute and lose their effectiveness as priming pumps of the ventricles. When this occurs in a heart that presents a certain degree of insufficiency, a severe arrhythmia can result. It is present in 10% of people over 65 years of age and is also easily identified in the ECG [5].

5.2 Detection of heartbeats, QRS detection.

In ECG processing, it is very important to detect very accurately heartbeats, because it is the base for further analysis. The QRS complex (ventricular depolarization) is the most visible waveform in the ECG, and thus detection of heartbeats consists in QRS detection. An accurate QRS detector is essential for ECG analysis [7]. The QRS complex is the graphic representation of the depolarization of the ventricles of the heart forming a peaked structure in the electrocardiogram. The QRS complex appears after the P wave and, because the ventricles have more mass than the cardiac atria, the QRS complex is larger than the P wave. In addition, it coordinates the depolarization of the ventricles at a very high speed. Elevated and, consequently, QRS complex waves tend to be very narrow and peak-shaped, rather than rounded [9]. In order to analyze the ECG signal is important to know waves that form the heartbeat (Figure 6).

Each beat is divided in three stages, atrial depolarization (P wave), ventricular depolarization (QRS) and finally ventricular repolarization (T wave). These three stages are continuously repeated in the ECG signal, representing heartbeats over the time [7].

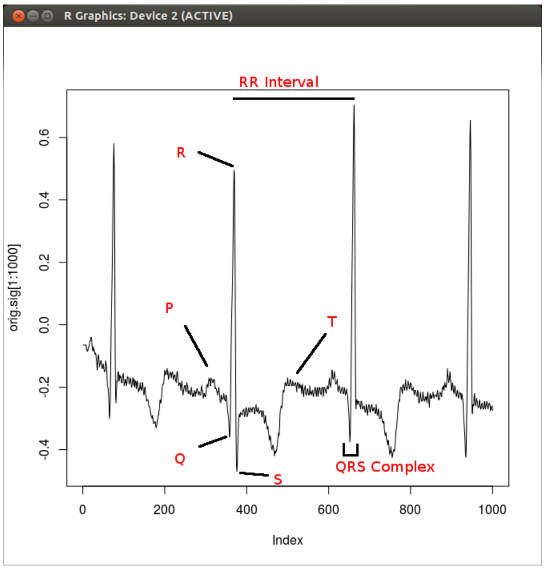


Figure 6. The QRS complex in the ECG signal.

QRS detection is difficult, because the signal varies along the time and different types of noise can be present in it. From the eighties, software QRS detection has been a research topic. Whereas in the early years its computational load and complexity, nowadays the detection performance determined the performances of the algorithms is the major objective [7].

In most QRS detection algorithms there are two differentiated stages: pre- processing and decision [10]. In the pre-processing stage, different techniques are applied, such as linear and nonlinear filtering or smoothing, to attenuate P and T waves as well as the noise. While in the decision stage the most important task is the determination of thresholds and in some cases the use of techniques to discriminate T waves. Some algorithms include another decision stage to reduce false positives [7].

Pahlm and Sörnmo [10] noticed that most of QRS detectors are divided in two sections: pre-processing stage and decision stage. Almost all algorithms use a filter stage before the detection, in order to remove the noise and reduce the amplitude of P and T waves to facilitate the subsequent detection [11]. Some algorithms apply a bank of high-pass and low-pass filters, which is known as a band-pass filter, while others do it separately. Once the signal is pre-processed, the QRS complex is detected by thresholding the signal, where thresholds can be fixed or adaptive. Finally, most algorithms use another decision stage, where decision rules are applied in order to reduce false positives [7].

During the last 30 years there have been proposed a lot of algorithms for QRS detection [7]:

- Derivate-based algorithms: algorithms based in filters and derivative [12]. They often use a high-pass filter and derivative is used to determine the maximum slope, which corresponds to QRS complex.
- Algorithms based on digital filters: other algorithms with use more sophis- ticated filters [13] [14]. Two different filters process the ECG, low-pass and high pass ones, with different cut-off frequencies, forming the band-pass filtered signal [15] [16].
- Wavelets: wavelet based approaches discompose the signal into different scale components to analyze the signal in different frequency bands [17] [18] [19].
- Neural Networks: neural networks are used to predict current signal values from the past ones, and therefore apply suitable filters to attenuate the noise [19] [20].
- Genetic algorithms: they intend to get optimal polynomial filters, for pre- processing stage, and parameters for decision stage [21].

Algorithm	Database	Sensitivity	Pos. Predictivity
N. Arzeno *2008* [12]	MIT-BIH	99.68%	99.63%
V. Afonso *1999* [8]	MIT-BIH	99.59%	99.56%
J. Pan *1985* [15]	MIT-BIH	99.3%	-
P. Hamilton *1986* [16]	MIT-BIH	99.69%	99.77%
J. Martinez *2004* [17]	MIT-BIH, QT, ST-T, CSE	99.66%	99.56%
C. Li *1995* [18]	MIT-BIH	99.8%	-
B. Abibullaev *2011* [19]	MIT-BIH	97.2%	98.52%
R. Poli *1995* [21]	MIT-BIH	99.6%	99.51%

Table 1: Performance of the most important QRS detection algorithms.

5.3 ECG to detect AF.

AF is a type of arrhythmia of increasing prevalence and it is associated with a reducible risk of having a stroke. There are some useful screening tests such as pulse palpation, single lead ECG… All of them with an acceptable sensitivity. On the one hand, there is pulse palpation test which is the cheapest method even though the test has a large number of false positives. On the other hand, there is the 12-lead ECG method, which is more cost effective in order to screen people randomly rather than to offer the test to everybody. Furthermore, definitive diagnosis of AF should be done by 12-lead test and interpreted by someone with the appropriate expertise. Up to now, the computer software is not sensitive enough all alone to find the diagnosis of AF. Primary care practitioners may not accurately detect AF on ECG, but consistently high accuracy can be achieved by healthcare professionals with adequate training. In patients post-stroke, a single ECG will miss cases of Paroxysmal Atrial Fibrillation (PAF) which can be detected by longer duration monitoring such as serial ECGs [22].

There are several ways to detect the features of an ECG and thus, discriminate whether it is AF or not. Methods are based on RR intervals or P wave method; but they have some limitations. As an example, if the ECG changes quickly or AF takes place, RR interval methods fail in accurate detection; or P wave is difficult to detect due to the small amplitude.

Although most of the conventional pattern recognition techniques have been previously applied successfully to the ECG arrhythmia detection tasks, in general, extracting highly representative features from the data in hand has the most significant impact on the performance of computerized classification/recognition systems.

5.4 Detection of AF, into deep learning.

Though it was not until recently it became part of daily life thanks to advances in big data availability and affordable high computing power. AI works at its best by combining large amounts of data sets with fast, iterative processing and intelligent algorithms. This allows the AI software to learn automatically from patterns or features in that vast data sets [23].

Machine learning refears to a broad family of techniques and algorithms (computer programs) that are able to learn from data. As data availability has become cheap, solutions based on machine learning techniques have proliferated.

For the purposes of our problem the techniques needed are known as supervised classification problems. Classification because our goal is to replicate a diagnoses, which has a finite number of possible outputs. In our case the patient either has AF or not. And supervised because the algorithm learns from labeled data, that is we train the algorithm by feeding it data consisting of ECGs and the diagnoses (ground truth) made by clinicians.

The classical supervised methods had two stages: first, an output label associated with each instance in the dataset is needed. This output can be discrete/categorical or real-valued. Data has known labels as output. It involves a supervisor that is more knowledgeable than the neural network itself. Then, the supervisor feeds some example data about which the supervisor already knows the answers. The supervisor guides the system by tagging the output. The algorithm would be first trained with available input data set that is already tagged with this classification to help the machine learning system learn the characteristics or parameters and distinguish them. Techniques such as linear or logistic regressions and decision tree classification fall under this category of learning.

More recently, deep learning methods that are able to automatically learn the best features and classify have been developed. These methods are based on neural networks. The difference between Machine Learning (ML) and Deep Learning (DL) can graphically be seen in the Figure 7.

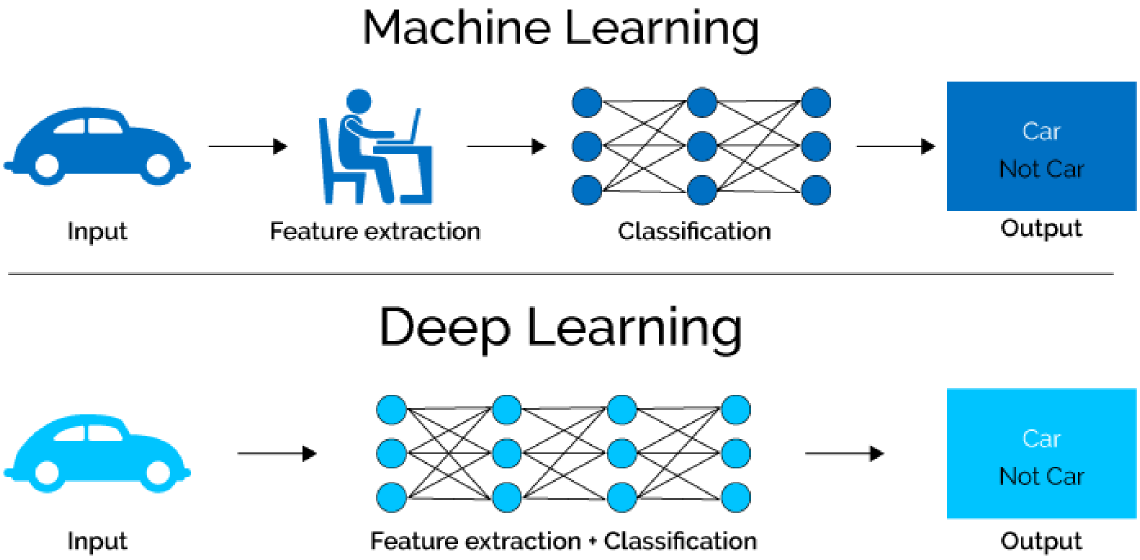


Figure 7. The difference between Machine Learning (ML) and Deep Learning (DL)

Deep learning inspired from human brain.

We have a biological neural network that is connected to our nervous systems. Our brains are very complex networks with about 10 billion neurons each connected to 10 thousand other neurons [24].

Each of these neurons receives electro-chemical signals and passes these messages to other neurons. Actually, we do not completely know how our brain neurons work. We do not know enough about neuroscience and the deeper functions of the brain to be able to correctly model how the brain works. DL is only inspired by the functionality of neurons which lead to the concept of artificial neural networks (ANN). ANN is modeled using layers of artificial neurons to receive input and apply an activation function along with a human set threshold. Deep learning has already achieved near or better than human level image classification, speech/hand writing recognition and of course the autonomous driving [24].

Deep-learning networks are distinguished from the more general single-hidden-layer neural networks by their depth.

Convolutional neural networks (CNN): one of most popular applied DL cases. They are great for image/video processing or computer vision applications. CNNs are deep artificial neural networks that are used primarily to classify images, cluster them by similarity, and perform object recognition within scenes. These are algorithms that can identify faces, individuals, street signs, tumors, flowers and many other aspects of visual data. The most popular applied corporate cases are probably optical character recognition (OCR) [25].

Technically, deep learning CNN receives images to pass through a series of convolution layers with filters. Of course initially these filters don't know where to look for features and the weights are random numbers. The model first makes a forward pass, calculates the initial weights, makes a prediction of the outcome label and compares it with the truth that is the existing training set labels. Because this is a training set we already know the outcome labels thus depending on the success of the prediction, a loss function is calculated and the network makes a back pass while updating its weights. The way the computer is able to adjust its weights to decrease the loss is through a method called back propagation. Now the model performs a backward pass through the network, which is determining which weights contributed most to the loss and finding ways to fine tune these weights so that the loss decreases through consecutive passes [25].

Initially the calculated loss is expected to be very high and it is expected to decrease to a minimum after many (but fixed) times of forward/backward passes. At the end hopefully the network should be trained well enough so that the weights of the layers are tuned correctly [25].

Then we run testing to be able to see whether our CNN model works. We compare the outputs to the testing set to see if and how well our network works. Naturally the more datayou have the better your model could be tuned through training and testing. That is why big data enables deep learning. After we have a good enough model, it is ready to be used for real life scenarios [25].

For example, a deep learning framework previously trained is transferred to carry out automatic ECG arrhythmia diagnostics by classifying patient ECG's into corresponding cardiac conditions. Studies are done to implement a simple, reliable and easily applicable deep learning technique for the classification of this disease. Latest results demonstrate that applying this method a high recognition rate is obtained. But not only a high recognition rate, 98,51%, also a test accuracy of around 92% [26].

5.5 Evaluation of a binary diagnostic test

A diagnostic test is used by physicians to help diagnose an illness, injury, disease or any other type of medical condition. In a typical binary diagnostic test, a positive or negative diagnosis is made for each individual patient, subject, or unit and the diagnosis is compared to the known true condition. When this is done there are four possible outcomes: true positive (A), false positive (B), true negative (C) and false negative (D) [27], these are shown in Figure 8.

		Test Result: Positive	Test Result: Negative
True Condition	Positive	True Positive (A)	False Negative (C)
	Negative	False Positive (B)	True Negative (D)

Figure 8. Confusion matrix of a binary diagnostic test in which the true contition (determined by clinicians) is compared to the result of the test (or detection algorithm).

In the terminology *true/false positive/negative*, *true* or *false* refers to the assigned classification being correct or incorrect, while *positive* or *negative* refers to assignment to the positive or the negative category [28].

A diagnostic test should be able to differentiate between those that have the disease or condition and those that do not. The most common measures of diagnostic test accuracy are sensitivity (true positive rate) and specificity (true negative rate). Stated differently, the sensitivity of a diagnostic test is the proportion of those that have the condition for which the diagnostic test is positive, and the specificity of a diagnostic test is the proportion of those that do not have the condition for which the diagnostic test is negative [27]. The formulas for computing sensitivity and specificity from a sample of diagnostic test results are

- **Sensitivity = True Positive Rate (TPR) = A/(A+C)**
- **Specificity = True Negative Rate (TNR) = D/(B+D)**

Often in diagnostic medicine it is important to compare the accuracy of two or more diagnostic tests used in a variety of applications. This can be done by comparing sensitivity and/or specificity using a statistical test [27], Figure 9 shows an example.

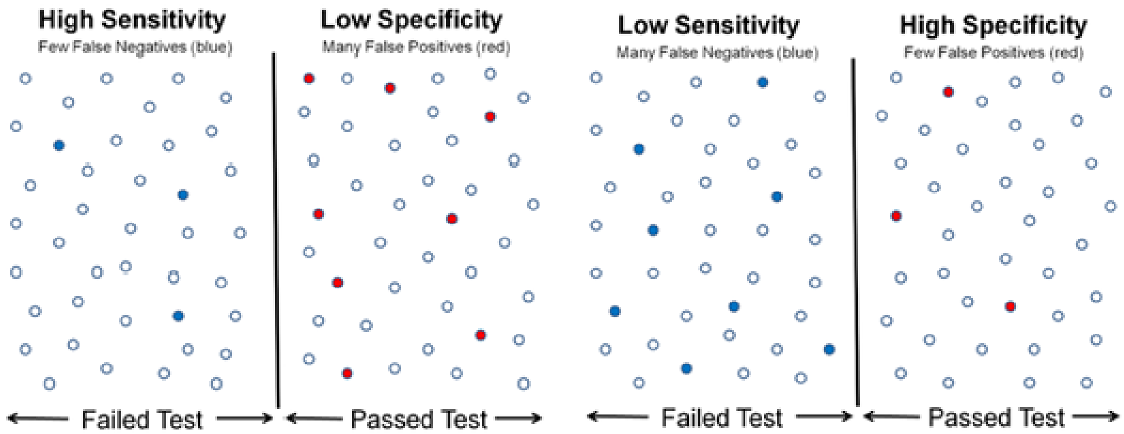


Figure 9. Example.

The terms "sensitivity" and "specificity" were introduced by the American biostatistician Jacob Yerushalmy in 1947 [28]. In our problem the algorithm will try to detect AF. So positives will be the ECGs with AF and negatives normal ECGs. The sensitivity will therefore be the proportion of detected AF ECGs, and the specificity the proportion of detected normal ECGs.

Avoiding overfitting when training a deep learning model is one of the aims of any model. To find out if the model is overfitting, that is, data fits the training set but isn't able to make an accurate prediction, cross-validation is used. Measures in the training set lets us see if the model is progressing in terms of training. With this in mind, loss and accuracy are measures of loss and accuracy on the training set, while validation loss and validation accuracy are measures of loss and accuracy on the validation set. For example, if a model has an accuracy of 86% on the training and 84% on the validation. Performing this model a 84% accuracy in new data can be expected.

5.6 The Atrial Fibrilation (AF) detection challenge

In 2017, PhysioNet/CinC launched a Challenge to encourage searcher to develop algorithms to classify single lead short ECG recordings and tell if those recordings show normal sinus rhythm, atrial fibrillation (AF), and some others we are not taking into account. Challenge participants must classify one lead ECG recordings as normal, AF, and some others we are not taking into account.The classification must discriminate between Normal or AF rhythms. In the real challenge, participants were asked to discriminate between some extra others as Noise and Others. The recordings were done using the AliveCor device, and were donated to the Challenge by them. The database contains 8,528 single lead ECG recordings which last from 9 to over 60 seconds. Al the recordings sampling frequency is 300 Hz and the AliveCor device provides them band pass filtered and in MATLAB format (a .mat). No additional software is needed, but MATLAB or GNU Octave must be use to do the Challenge. MATLAB programming language may be used. To participate in the challenge, software must be created. This software must be able to read the data provided and to output a result without the user interaction [29].

6.Alternatives analysis

To fulfill the project objectives efficiently, in this section we will analyze the different alternatives that have been considered to carry out the project.

First of all, I choose an adequate software platform to develop the project following the challenge guidelines that we are solving. Then, we will discuss what method we are going to use to discriminate between the different rhythms. Finally, we will analyze the methods of automatic learning that we have considered once the method to be followed had been decided.

6.1 Software Platform

Regarding the software platform, we have considered three options. The two proposed in the AF detection challenge: MATLAB and GNU Octave; and one extra. Below are explained the characteristics and differences of each of them.

6.1.1 MATLAB

The MATLAB (Matrix Laboratory) software platform is a mathematical software tool that offers an integrated development environment (IDE), which has its own programming language (M language). As an interpreted language, it offers a wide range of facilities to the user, even if he is not an expert. However, this also affects the compiling speed, which is somewhat slower.

Matlab offers additional tools very useful for the user, such as toolboxes, where you can find applications and functions already developed. Additionally, it has a very complete help guide and a technical support website. (http://es.mathworks.com/products/matlab/).

The main disadvantage of this software platform is its price, since it is quite expensive. The price of the license is €800 per year, without taking into account the additional toolboxes that are needed. The price of these is approximately €500 each.

6.1.2 GNU Octave

Octave is an interpreted language similar to MATLAB that uses a programming language of its own. The biggest advantage of this software platform is that it is free code software. Although the options offered by the Octave platform are similar to those of MATLAB, there are great differences between them.

As an example, graphical user interfaces (GUI) are not as developed in Octave as they are in MATLAB. Developing a GUI is an essential tool in this project, Octave loses points when it comes to being chosen as development software.

In addition, Octave does not have as many toolboxes as MATLAB, the guide is less concrete and the technical support is not as complete.

6.1.3 Python

Although the Challenge itself does not consider this option, as Python is a very useful tool when working with neural networks, we will also take it into account.

Python is an interpreted programming language whose philosophy promotes the syntax of a readable code.

It is a multi-paradigm programming language, which means that it supports object-oriented programming, imperative programming and, less extensively, functional programming. It is an interpreted, simple and multiplatform language.

It is open source software, provided by the Python Software Foundation. Keep in mind that not all versions are retro compatible.

6.1.4 Criteria for selecting the software platform

Facility
Software complexity is established as an important feature.

Weight: 2/10

Algorithm development time
The importance of being able to develop the algorithms developed in the project in an easy and efficient way must be taken into account. In addition, the possibility of using functions already developed.

Weight: 3/10

Organization and visualization of data
In this project we work with a large amount of data from a database and, therefore, the ability to handle data with ease and also, the visualization of this is of good quality.

Weight: 2/10

Execution time
Another important criterion to take into account is the effectiveness of the software platform when executing the code developed, related to the execution time.

Weight: 2/10

Price

The price of the program is another criterion that must be taken into account.
Weight: 1/10

To determine the most appropriate software platform, the following table will be used:

Criteria	Weight	MATLAB	GNU Octave	Python
Facility	2/10	2/10	2/10	2/10
Algorithm development time	3/10	3/10	1.5/10	1.5/10
Organization and visualization of data	2/10	2/10	0.8/10	2/10
Execution time	2/10	1.5/10	1.5/10	1.8/10
Price	1/10	0.4/10	1/10	1/10
Total	10/10	**8.9/10**	6.8	8.3/10

Table 2: Software selection.

6.2 Methods

Regarding the methods for the detection of different rhythms, 3 alternatives have been considered.

6.2.1 Classical AF processing (f waves)

The interpretation of an ECG easily establishes the diagnosis in persistent atrial fibrillation but it is not so simple for the paroxismal atrial fibrillation. This test must be done when the patient has symptoms.

The diagnosis of atrial fibrillation relapses in the demonstration in an electrocardiogram of the absence of P waves (normally with the presence of f waves). In addition, in a patient without symptoms, abnormalities such as the aforementioned alterations in P waves that can favour the appearance of the disease can easily be seen.

These waves f are irregular oscillations in configuration, amplitude and frequency. As with P waves, f waves are best evaluated in leads DII and V1.

Therefore, in order to make a timely diagnosis, first a conventional ECG is performed in which if any alteration is seen, then another ECG, a DII or V1 derivation is requested, and then a doctor or more specifically a cardiologist, will arrive at a diagnosis [30].

6.2.2 Machine Learning

As mentioned earlier in this document, Machine Learning is a process by which a programmer introduces the characteristics of what is to be detected and then, computationally, a solution is arrived at working with the characteristics that have been previously introduced.

The disadvantage of this process is that there must be someone, prior to the computational part, who introduces the characteristics and, therefore, that knows them.

6.2.3 Deep Learning

Deep learning is an evolved process of Machine Learning itself. The advantage of this method with respect to ML, is that the method itself extracts the characteristics of the input, that is, it is not necessary for a programmer to enter them manually.

This advantage lowers costs, since it is dispensed with the person who knows the characteristics of what is being worked on.

6.2.4 Criteria for selecting the methods

Facility
The ease of application
Weight: 2/10

Time

The time it takes from suspicion to diagnosis.

Weight: 3/10

Personal and knowledge requirement

The staff and knowledge that these people require to reach a diagnosis

Weight: 2.5/10

Price

The price of the program is another criterion that must be taken into account.
Weight: 2.5/10

To determine the most appropriate method, the following table will be used:

Criteria	Weight	Classical process	Machine Learning	Deep Learning
Facility	2/10	1/10	2/10	2/10
Time	3/10	1/10	2/10	3/10
Personal and knowledge requirement	2.5/10	1/10	1.8/10	2.5/10
Price	2.5/10	1.5/10	2.5/10	2.5/10
Total	10/10	4.5/10	8.3/10	**10/10**

Table 3: Method selection.

6.3 Learning methods

Regarding the learning methods, I have considered the three options once the methods of automatic learning had been decided.

6.3.1 Convolutional Neural Network

A convolutional neuronal network is a type of artificial neural network where neurons correspond to receptive fields in a manner very similar to neurons in the primary visual cortex of a brain.

They are a really good option for image processing applications. That is why they are one of most popular applied DL cases. CNNs are deep artificial neural networks that are used to classify images and recognize different scenes.

6.3.2 Recurrent Neural Network

The neural network includes feedback loops, which allow the information to be maintained during the different steps or epochs of training through the output. This data is "embedded" (embedding) in the entry.

It is a network of multiple copies in which the output is inserted back into the input (feedback).

The connections between nodes form a graph directed along a time sequence.

The training of the neural network stretches over time at every step, which is why it is very long in time and consumes a lot of RAM.

To simplify, you can 'unroll' the networks in the different time steps as if it were a non-recurrent network. The more 'unrolled' the network is, the more the process will be accelerated.

The longer it is, temporarily speaking, the greater the number of layers to be unrolled, which can cause problems. This is solved using adjacent layers of type LSTM or GRU that allow a simple feedback.

6.3.3 Long short-term memory

It is a deep learning tool. Specifically, an artificial recurrent neuronal network (RNN) architecture.

The difference of LSTM with a standard direct-feed neural network is that LSTM has a feedback connection. It can process from images to voice or video.

A usual LSTM it is made up by a cell, an entrance door, an exit door and a forgotten door. The cell remembers values in for some time and the doors regulate the flow of information that enters and leaves the cell.

LSTM are a good option to classify process and make predictions coming from external time series data. One of the disadvantages of this method is that there might be variant duration delays. The LSTMs were developed to deal with the explosion and disappearance gradient problems when training traditional RNNs.

6.3.4 Criteria for selecting the methods

Complexity
The ease of application

Weight: 3/10

Extra needs
The need for adjacent structures.

Weight: 3/10

Resource consumption
The amount of resources consumes both time and memory

Weight: 4/10

To determine the most appropriate learning method, the following table will be used:

Criteria	Weight	CNN	RNN	LSTM
Complexity	3/10	2/10	1/10	1.5/10
Time	3/10	2/10	1.5/10	1.5/10
Resource consumption	4/10	3/10	2/10	2/10
Total	10/10	**7/10**	4.5/10	5/10

Table 4: Learning method selection.

7.Description of the solution

To execute this project efficiently, agilely and visually, in addition to a simple and economical way; its development has been divided into different phases. These phases go from the initial project plan to its realization, including its main results and conclusions. To understand this in a simple way before going on to tell what all the steps have been, a brief summary has been made to have a global vision of it.

First, the conversion of the original database into a MatLab format suitable for our processing needs. Some of the original information fields were removed, and the existing data reorganized. We even add fields that are useful to us throughout the execution of the work and we adapt it to make it easier to work with it. Then, the creation of a data visualization tool, that is, a graphical user interface (GUI). Once we have the data and we are able to visualize them efficiently, we proceed to prepare the data to be able to do what our work really looks for, which is the correct detection of atrial fibrillation using the ECG. Therefore, the data is prepared in order to be introduced into a deep neural network, the most important and delicate part of the project. This preparation consists of two phases, which is the preparation of the data itself. Data preparation included the selection of the records with which we want to work and their adaptation to the work method, and its subsequent separation in training and testing. To train the network and check that it works as we wish. Then, we create a convolutional neural network (CNN) so that it learns from the data that we provide; and finally, we check the results obtained and we come to a conclusion.

7.1 Conversion of the database.

Before starting work we consider if the information we have is useful and if it is, if we have it in the format with which we like to work with. The database that is provided to us has very valuable information but in a format that does not suit the needs of the project, as I mentioned before there are fields that do not interest us that are there, and we prefer them elsewhere or we also have the need to add fields with which undoubtedly we will work later to achieve the detection of atrial fibrillations in a simple and economical way. For that reason it is decided to make some modification in this as it can be the elimination of some fields, so that at the time of working with it, it is easier to execute, to find, thus achieving the objective that it is easier to handle.

To see a little about the change that has been made in it, we will first explain the fields it originally contained by explaining them one by one. Later, we will explain which fields have been replaced or modified and finally the fields that have been added.

The original database contained 8528 records and each of them contained the following fields: 'name', 's_ecg', 'r_type', 'fs', and 'info'.

- name: this field contained the record name. This name was given as 'A00000'. All records started with the letter A in uppercase and followed by a 5-digit number. As an example, the first record is 'A00001' and the last one 'A08528'. This field has remained unchanged throughout the entire work. And it has been useful to name the signals that have been modified.

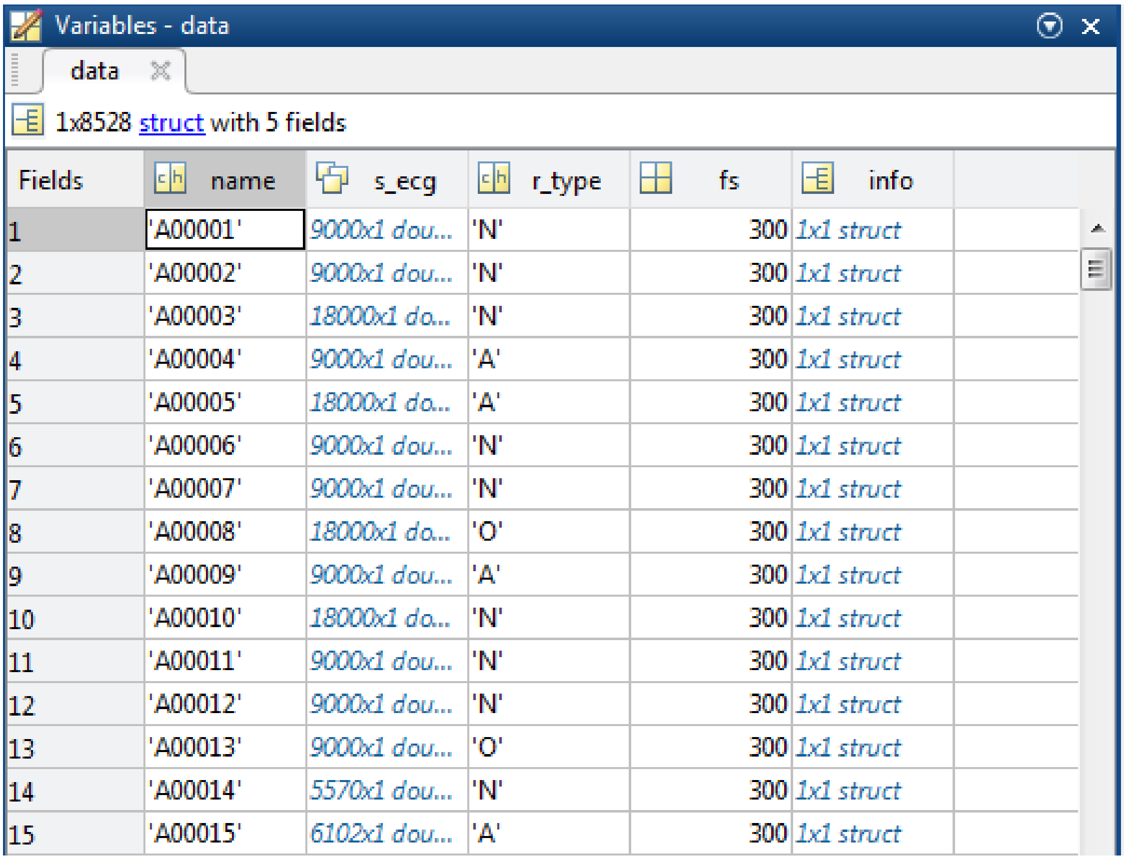
- s_ecg: this field contained the numerical data with which to represent the electrocardiogram of a patient. These records had different lengths, such as 9000x1 records or 18000x1 records among others. This field, however, is one of the main causes of having to remodel the database because, since there is so much data, it makes the database very heavy and, therefore, not very manageable.

- r_type: the content of this field is shown as: 'N', 'A', '~' and 'O'. With 'N' reference was made to the registers of normal cardiac rhythms, with 'A' to those corresponding to atrial fibrillations, with '~' those records that were impossible to recognize since they were mostly noise; and finally, with 'O', to those records denominated as others. This field on the other hand is essential because later, when constructing the neural network, we will have to give information on what type of record is treated so that it can learn correctly.

- fs: the content of this field is simply the sampling frequency of the records. In all cases it is 300Hz. It is important not to make changes to this field since it is the frequency with which the ECGs have been sampled and it is essential to know it for the correct functioning of the software that we will develope.

- info: in these fields, additional information is found, such as the units of measure, the gain applied in each record or the description that is made of the record. This field is both useful and useless. That is, once it has been read and verified that the information is global and correct, it could be eliminated.

In Figure 10, we can see how the original database in MatLab environment works.

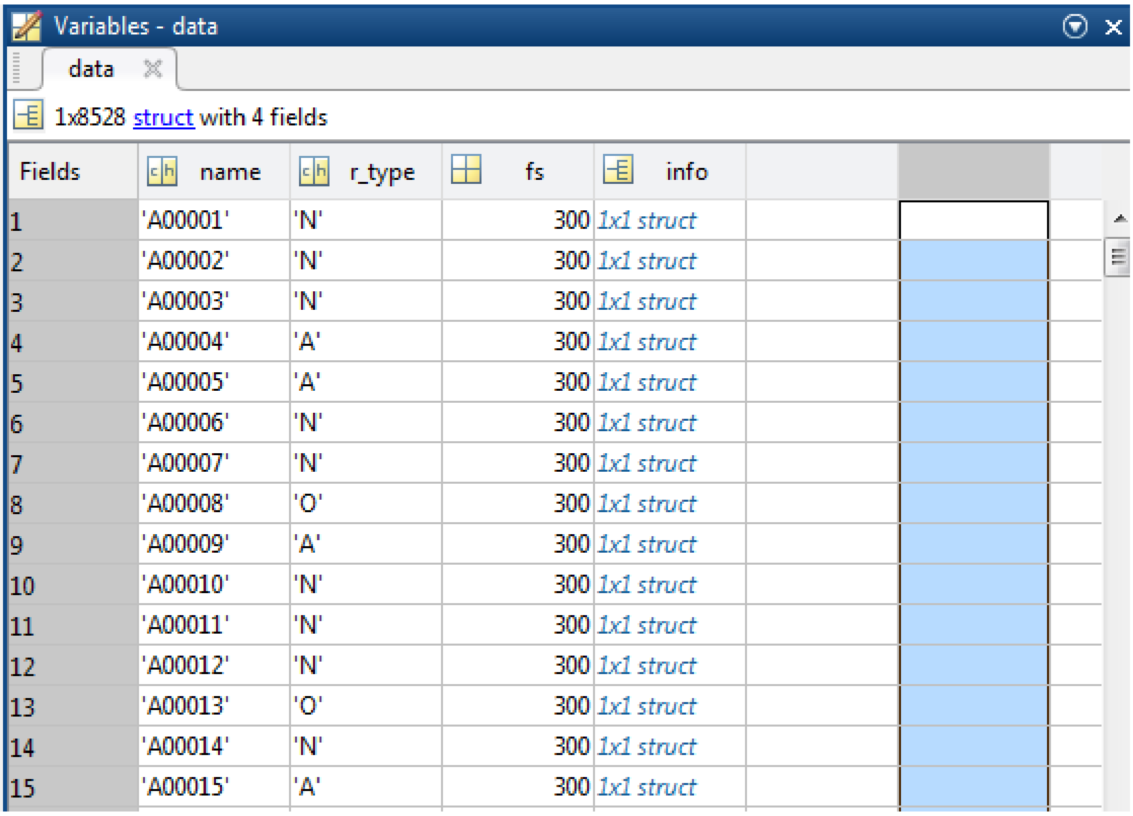


Fields	name	s_ecg	r_type	fs	info
1	'A00001'	9000x1 dou...	'N'	300	1x1 struct
2	'A00002'	9000x1 dou...	'N'	300	1x1 struct
3	'A00003'	18000x1 do...	'N'	300	1x1 struct
4	'A00004'	9000x1 dou...	'A'	300	1x1 struct
5	'A00005'	18000x1 do...	'A'	300	1x1 struct
6	'A00006'	9000x1 dou...	'N'	300	1x1 struct
7	'A00007'	9000x1 dou...	'N'	300	1x1 struct
8	'A00008'	18000x1 do...	'O'	300	1x1 struct
9	'A00009'	9000x1 dou...	'A'	300	1x1 struct
10	'A00010'	18000x1 do...	'N'	300	1x1 struct
11	'A00011'	9000x1 dou...	'N'	300	1x1 struct
12	'A00012'	9000x1 dou...	'N'	300	1x1 struct
13	'A00013'	9000x1 dou...	'O'	300	1x1 struct
14	'A00014'	5570x1 dou...	'N'	300	1x1 struct
15	'A00015'	6102x1 dou...	'A'	300	1x1 struct

Figure 10. Original database.

Even though everything in the database is useful, having the records (the s_ecg field) makes searching in the database slow. This slowness is caused by the fact that, as we mentioned before, a single file to store all the signals is too large. That is, a single record can be made up of 9000 samples or 18,000 samples, but if this amount is multiplied by the 8528 records that make up the total database, it becomes a number of samples so large that it slows down the process too much. Therefore, it was decided to create an external folder to the original database called SIGNALS in which each record was saved individually, with the name that is provided in the original database and only the s_ecg field.

This is how it is now, as can be seen in Figure 11, the database once this field has been modified.


Variables - data

data

1x8528 struct with 4 fields

Fields	name	r_type	fs	info
1	'A00001'	'N'	300	1x1 struct
2	'A00002'	'N'	300	1x1 struct
3	'A00003'	'N'	300	1x1 struct
4	'A00004'	'A'	300	1x1 struct
5	'A00005'	'A'	300	1x1 struct
6	'A00006'	'N'	300	1x1 struct
7	'A00007'	'N'	300	1x1 struct
8	'A00008'	'O'	300	1x1 struct
9	'A00009'	'A'	300	1x1 struct
10	'A00010'	'N'	300	1x1 struct
11	'A00011'	'N'	300	1x1 struct
12	'A00012'	'N'	300	1x1 struct
13	'A00013'	'O'	300	1x1 struct
14	'A00014'	'N'	300	1x1 struct
15	'A00015'	'A'	300	1x1 struct

Figure 11. First modification of the database.

Being able to access the signals in an external folder makes data access much more dynamic since it is not necessary to load the entire database when all you want to do is access the signal, for example, to visualize it.

To load one of the signals, it is as simple as accessing the folder.

This would correspond to the elimination or modification part but it has also been said that in order to obtain the database with which we want to work, it is possible that some other field has to be added. For this reason, one field has also been added: 'pQRS'. This field was added when the detection of the QRS complex was made, so we will describe it later when the QRS complex detection process is explained.

7.2 GUI of visualization

Once we have the information to our liking in the database we proceed to create a graphical user interface, a GUI.

The GUI (Figure 12) has been created to be able to select between the different types of cardiac rhythms that we have from the database and of these rhythms, the different registers. It should also be mentioned that not all types of rhythms have the same number of records, being the normal heart rate, much higher in number than the others. In this tool, besides being able to visualize the different registers, extras have been added such as filters to eliminate some interferences, like power line interferences (50Hz in Europe, 60Hz in the US) or the elimination of low frequencies. Making it more accurate in the process due to the elimination of noise.

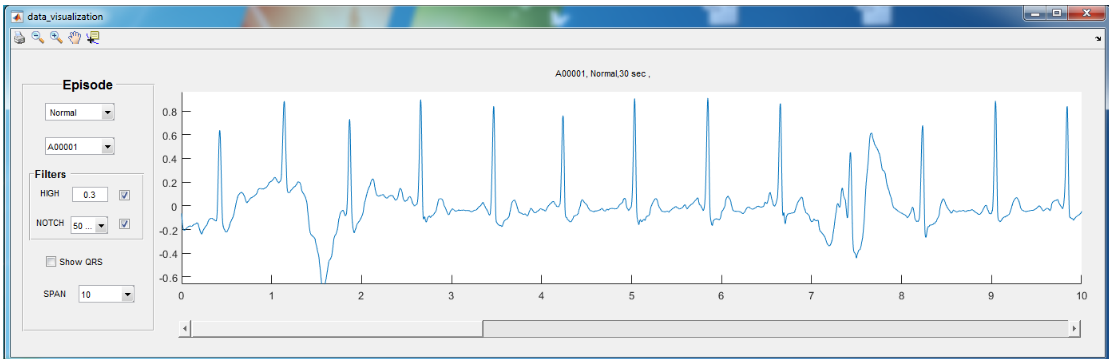


Figure 12. GUI tool.

This tool, in addition to the above mentioned, provides other tools such as the choice of Span or the Slider. With the Span we can make cuts or visualizations of the seconds that it indicates. And with the Slider, if the signal is longer than the value of Span we can move through it. Figure 12 shows an example. We have a Span selected for 10 seconds, so the first 10 seconds of the signal are shown. But as you can see, the Slider (the slide bar below the signal) is still allowing us to move through the signal. Although an anomaly may not appear in the first 10 seconds, it may appear later and this tool allows us to see the entire signal more clearly and slide through it to see the possible changes that may have occurred.

After the realization of this tool, a standard QRS detector was integrated that allows the user to visualize the R peaks of a QRS complex.

First, before proceeding to the detection of the R-peaks in the QRS complex, the necessary filters are applied. These filters are provided by the GUI and explained a little further in this section. Once the signals with which we want to work with are filtered and ready to be used, the detection of the QRS complexes is carried out by means of an automatic QRS detector. In our case, the detector Hamilton-Tompkins II detector [16] has been used. In Figure 13 it can be seen how the GUI is with the QRS complex detector activated.

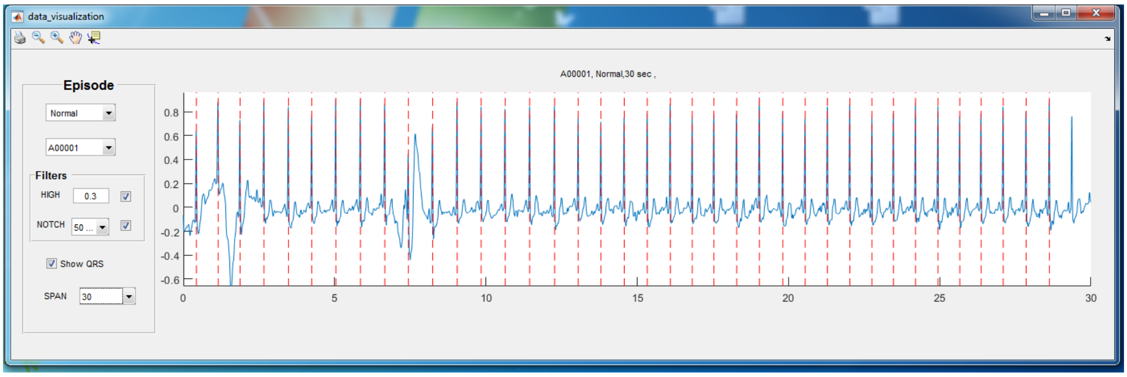


Figure 13. QRS detector in the GUI.

Before, we have said that once we arrived at this point, we would explain the addition of the 'pQRS' field to the database. Well, this field is added here, so that the detection of this QRS complex is done only once, that is, when it is being programmed. Having at hand the points in which a QRS complex is detected and being able to visualize them or not to our liking, simply by pressing a button, and not having to do the process once again, makes the process much more faster to the user. As it is thought that it is a health care provider who does it, this makes it a quick, simple and inexpensive process in time, as if it could be having to do it from scratch. It has been given the name pQRS and the only thing that the name of this field means is the position (p) of the QRS-pQRS- complex. It is possible, also, to add the tQRS field, that is, the time instant in which these pQRS points are found, but it is not absolutely necessary since it corresponds only to the arithmetic operation of dividing the position of these points by the sampling frequency.

Once the pQRS field was added the database contains additional information on the locations of the beats, as shown in Figure 14.

Variables - data

data

1x8528 struct with 5 fields

Fields	name	r_type	fs	info	pQRS
1	'A00001'	'N'	300	1x1 struct	1x38 double
2	'A00002'	'N'	300	1x1 struct	1x36 double
3	'A00003'	'N'	300	1x1 struct	1x82 double
4	'A00004'	'A'	300	1x1 struct	1x32 double
5	'A00005'	'A'	300	1x1 struct	1x109 double
6	'A00006'	'N'	300	1x1 struct	1x29 double
7	'A00007'	'N'	300	1x1 struct	1x30 double
8	'A00008'	'O'	300	1x1 struct	1x70 double
9	'A00009'	'A'	300	1x1 struct	1x47 double
10	'A00010'	'N'	300	1x1 struct	1x66 double
11	'A00011'	'N'	300	1x1 struct	1x35 double
12	'A00012'	'N'	300	1x1 struct	1x32 double
13	'A00013'	'O'	300	1x1 struct	1x46 double
14	'A00014'	'N'	300	1x1 struct	1x17 double
15	'A00015'	'A'	300	1x1 struct	1x29 double

Figure 14. Final database.

7.3 Data preparation

Once we have the data to our liking, it is easy and convenient to access them, and we are able to visualize them efficiently; we proceed to the first step of preparing the data that we will then introduce into the neural network.

The first step of this data preparation is to filter the signals with the filters that have been designed for the GUI, such as the electric signal frequency elimination filter and the low frequency of choice elimination filter. But before this, we select the signals with which we will work. Although the challenge is intended to differentiate atrial fibrillation from the 3 remaining types of rhythms (normal rhythm, others and noise), in this work we will differentiate only between a normal rhythm and atrial fibrillation. Once the 5788 records have been selected, they are filtered. Of these 5788 records 738 correspond to records of atrial fibrillation and the remaining 5050 with records of normal heart rhythm. When the signals have already been filtered, we select the first 5 seconds of all the signals we have in the database. These signals are stored in a new matrix that we call matrix X. This matrix X will be one of the important steps to take before the CNN, because it is a fundamental part to be able to develop this network. These signals are saved in a concrete way, which is the following: A 4D matrix is created, that is, in four dimensions. This means that we will have four fields, in which the fields are the following: (1, Y, 1, i). In the place where Y is, we put the signal itself, that is, the one corresponding to the 5 seconds to which we have previously reduced our electrocardiogram signal; and 'i' is an index to the current ECG. The value of 'i' goes from 1 to 5788. They are saved in this way since convolutional neural networks are designed to work with images or videos and they have 4 fields. The fields correspond to the unit vectors (h, i, j, k), but as in our case we do not complete all of them since our signal is only in two dimensions, so we simply put a 1 to indicate that this field, throughout the execution of it, it will not vary.

At the same time that we obtain this matrix X, which from here on we will call tensor, we obtain the matrix Y. This matrix is a 1x5788 matrix, in which we indicate if the record with which we are working is of the normal type or of the atrial fibrillation type. This is done by creating this matrix in a binary way, that is, it is or it is not. When the record corresponds to an atrial fibrillation, at the position corresponding to said record in the matrix Y will appear a 1. However, if the record corresponds to a normal heart rate, a 0 will appear. This will be one of the most important matrices when programming the convolutional neuronal network (CNN) because this will be the one that indicates the learning process. If it is of the type that has to learn or on the contrary, it is a normal heart rate. It will also serve to see if the prediction that the neural network makes when it is executed is correct, or on the contrary, the learning process has failed. Since this matrix is composed only of records to which we have assigned binary values, that is, they discriminate between one rhythm and another.

7.4 Partitions

As mentioned earlier in this section, the preparation of the data has two phases and once, the first part, that of the selection, filters, etc. has been completed, we can move on to the next one. In this part we separate what will be the training part and what will be the testing part or in other words, the part of the database that we will use to make the CNN learn the characteristics that we want it to know and the other part will be used to check if what it has learned is correct or not, and if it is, how reliable a prediction of this algorithm can be.

This division is an important step since it is necessary to divide the data that we have in two, in order to dedicate a part of them to the training of the CNN and the other part to the verification of the CNN in unseen data. That is, we dedicate a part of the records to learn and the other part to verify that what has been learned has been learned well.

The learning process is that the CNN is provided with characteristics of what we want it to detect and this, seeing these characteristics, learns how they are and it is able to detect a similar record of a new entry. In our case, Deep Learning, we, as programmers, do not directly introduce the features but we tell it which records meet the condition it must learn and the process learns them; however, in Machine Learning, it is the expert programmer in the field who must enter these characteristics manually. In our case, the programmer should not only be an expert in the programming of this type of algorithm, but also be aware of what it entails, an atrial fibrillation. This last field of knowledge is usually, in general, related to clinical experts in the field of cardiology.

To do this we are going to dedicate 70% of the records to training and the rest, the remaining 30%, to testing.

Therefore, 70% of the 5788 records are randomly selected. Doing it only once would not be logical, so it is done 50 times. That is, we selected 50 different times, randomly 70% of the data. It is important to keep in mind that 70% of the total is not selected, but 70% of the normal rhythms and 70% of the rhythms that represent atrial fibrillation. Otherwise, it could be unbalanced. And so, we make sure both rates are proportional. It is important that this is provided, if not, from one of the 50 times, the process will learn a lot from one type and little from another and vice versa; and that in the general calculation can be detrimental to the objective we have.

On the other hand we do the same with the remaining 30%, which will be used to check that once the convolutional neural network has learned the characteristics of what we have introduced, what it has learned is well learned. In case that in the 30% test the result was not the expected one, we should check the parameters that we have given the network to learn, check if they are correct. If they are correct we could consider two things: that the learning registers are not enough and therefore have not been able to learn enough or that there are parameters that are incorrect.

In the first case, this could happen when we have a small database or we dedicate a low proportion to learning compared to the test. In our case it is about 70% -30%. It is a good proportion for learning, which is what we are interested in. However, other options such as 30% -70% or 40% -60%, are somewhat more delicate and can lead to errors. In the second case, the only possible option is to change the parameters with which we configure the convolutional neural network and to vary them until obtaining the desired result.

7.5 Architecture of the CNN developed neural network.

The CNN as we have explained in a previous point, it is a type of artificial neural network very similar to how neurons work in the human brain. These networks, like human's brains, learn from examples or situations in which an event is similar.

For example, a child does not differentiate colors. That is, your brain has not assimilated that concept yet. He has not learned it. However, after a series of external stimuli, such as parents or teachers, who already have that acquired knowledge and are able to explain to the child how to differentiate them, he learns them until he is completely able to distinguish them by himself.

Well, neural networks work in the same way. At first they do not know anything. After a series of stimuli, which in our case are the 5788 records and a stimulant, which in our case the matrix Y that tells us if it is what we are looking for or not.

Knowing that both a stimulus and a stimulant are necessary, these are the steps that have been followed to realize the architecture of CNN:

In the first place, what we have called stimuli and stimulant are loaded, that is, the tensors or matrix X and matrix Y. We change to categorical the tensors corresponding to the matrix Y. This change is due to a requirement of the software that we are using. Once these, the tensors are available, two types of matrices are created. Some with the training data and some others with the test data that we are going to use.

Subsequently, the architecture of the CNN is defined. With CNN architecture, we refer to how the different steps have been structured and why they have been built. This architecture contains 5 CNN Layers of different characteristics. One of the differences is in the filter coefficient or how many different filters are available. On the other hand the number when making maxpooling also varies, that is, from a specified number of samples, to take only one. This is because in the early stages we make the filter wider and we reduce it so that it can be more precise. This structure is called a funnel structure. Once the layers have been arranged like this, it ends with a classification stage, based on a shallow neural network.

Thus, the larger the number of the filter, the less selective it will be. On the other hand, it is not advisable to put an excessively strict filter to begin with; otherwise it would not make much sense to put the subsequent filters. For this reason, we have followed the funnel structure and the first filter is made of an order twice greater than the fourth, and fifth.

Despite having said that the structure of a funnel must be made, the first three layers have the same number of filters. This is because not only the number of filters is important, but also the number of samples that are taken. In our first two layers, from seven samples available only one is taken, and in the third from every six samples only one is taken.

On the other hand, the same process has been followed with those layers in which the filter number was lower. In this case, the proportion of samples taken was one of seven possible.

In the Figure 15 the result obtained in this project is shown.

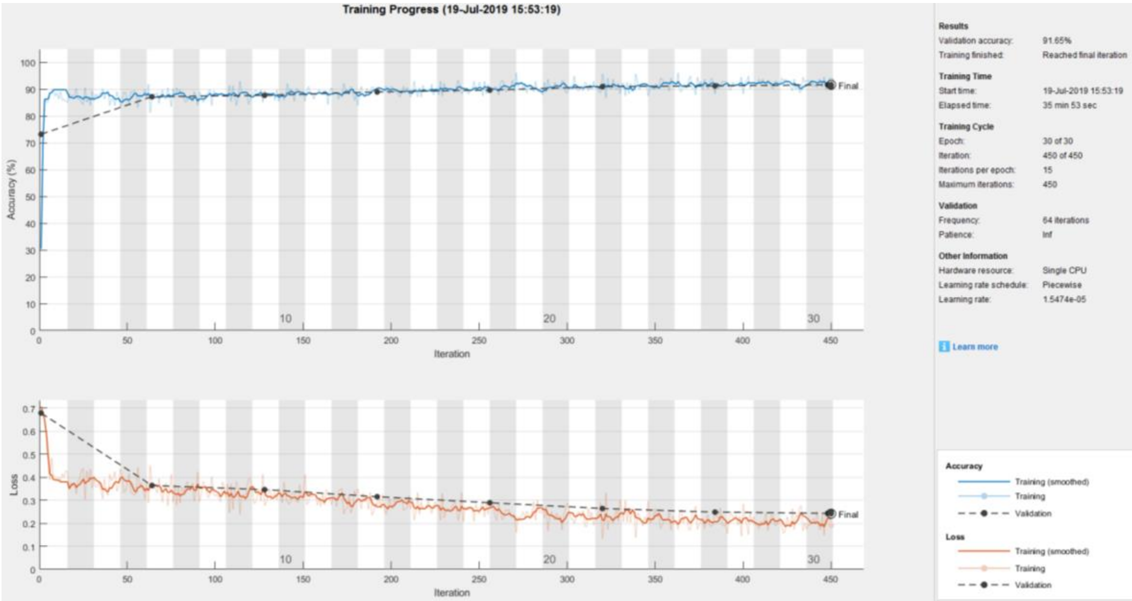


Figure 15. Results.

7.6 Summary of results.

The results obtained are quite good for what was predicted at the beginning. Our project ends with a validation accuracy of 91.65%. That is, 91.65% of the cases in which the software was used, would give a correct result. This would have a great impact on the lives of people because it would detect most of the cases of this pathology. As it has already been said, it is not usually fatal, but it can end up causing heart diseases that are. And having tools like these, that makes these pathologies easier and faster to detect, can lead to an early diagnosis and this pathology can be treated with enough time.

Although the result we have obtained is high, we started with a not so encouraging result. The first time the result was obtained was 86.06%. Normally, in order to start considering a good result, a value higher than 90% is expected, so this value was insufficient.

As we have indicated above, if the result was not the desired one, we had two options. In our case, the option to expand the percentage of learning was discarded since with 70% it is enough for the convolutional neural network to learn correctly. So the error or what we needed to improve was in the architecture of the neural network. More specifically the values we had given to that architecture.

Throughout the different proposals for the architecture of the neural network the following results have been obtained: 86.06%, 86.81%, 86.98%, 87.10%, 87.15%, 87.21% , 87.27%, 88.94%, 89.00%, 89.52%, 90.96% and finally 91.65%.

8. Methodology

Before beginning the development of any project, it is essential to specify the tasks and responsibilities of all those involved in the project. In addition, it is essential to define the limits and workload of each of the participants to carry out the project in an efficient and controlled manner. Therefore, in this section of the document, on the one hand, the work team will be presented and, on the other hand, the work packages made by the team participants. Additionally, the milestones and the concrete deliverables of the project are explained.

8.1 Work team

In this first section, the work team responsible for the development of the project will be briefly described, defining the tasks and responsibilities of each person. The following table shows it clearly and concisely:

Code	Responsability	Name and surname	Task
C1	Proyect director	Unai Irusta Zarandona	He proposes the project, indicates the necessary steps to follow and he is in charge of the correction and supervision of the document.
C2	Junior Engineer	Iratxe Asurmendi Perez	It is responsible for the development and drafting of the project. Author of the project.

Table 5. Project work team

8.2 Work packages

In the tables that are shown below, the workpackages that have been defined in the project is presented. In each work package the description of that phase and the subtasks are explained, specifying the duration and the start and end dates of each one.

First phase of the project.

WP1	Start date	End date	Duration (days)
PROJECT MANAGEMENT: Monitoring and administration carried out to verify the proper development of the project.	10/22/2018	07/22/2019	273
WP1.1: Management, monitoring and supervision of work: Coordination, supervision and administration work throughout the project until its closure.	10/22/2018	07/22/2019	273

Table 6. First work package

Second phase of the project

WP2	Start date	End date	Duration (days)
PREPARATION OF THE PROJECT: Acquisition of the necessary knowledge before specifying the course of the project and development.	09/03/2018	07/08/2019	308
WP2.1: Previous training: Study of the basic concepts that have been necessary and of help before the development of the project: concepts of cardiac rhythms, GUI environment concepts, etc.	09/10/2018	10/05/2018	25
WP2.2: Concretion of the project: Specification of the project line and the work plan.	10/03/2018	10/05/2018	2
WP2.3: Deepen in the environment of the subject: Search for the information and studies necessary for the development of the project, either on the subject itself or on the resources used.	10/22/2018	07/08/2019	259

Table 7. Second work package

Third phase of the project

WP3	Start date	End date	Duration (days)
PROJECT DEVELOPMENT: Different sections that have been carried out for the development of the project.	10/22/2018	07/07/2019	258
WP3.1: Development of the visualization GUI: In order to work with the signals of the database already converted, a GUI of visualization is developed, which will also serve to show the detector of the QRS complex.	10/22/2018	04/03/2019	163
WP3.2: Development of algorithms: **3.2.1 Automatic QRS detector:** Detection of QRS complexes by means of an automatic QRS detector. In this case, a Hamilton-Tompkins II is used.	04/03/2019	06/05/2019	63
3.2.2 Neural network: Creation of a neural network for the discrimination between atrial fibrillations and normal cardiac rhythms.	06/05/2019	07/07/2019	32

Table 8. Third work package

Fourth phase of the project

WP4	Start date	End date	Duration (days)
DOCUMENTATION AND PRESENTATION OF THE PROJECT: Drafting of the project and oral presentation.	06/10/2019	09/03/2019	85
WP4.1: Project documentation: Development of the document that summarizes the context of the project, the objectives, the scope, the benefits, the description of the solution and the conclusions.	06/10/2019	07/22/2019	42
WP4.2: Oral presentation of the project: Oral presentation of the project before the jury.	07/22/2019	09/03/2019	43

Table 9. Fourth work package

Description of the work units:

Unit	Description
Project	52 weeks
Week	7 days
Day	3 hours

Table 10. Work units description

8.2.1 Milestones and deliverables

In this section, the milestones and deliverables that have been fulfilled during the execution of the project are defined.

ID	Milestone	Date
M1	Beginning of the project	09/03/2018
M2	Definition of the project	10/05/2018
M3	Completion of the visualization GUI	03/04/2019
M4	Completion of the automatic QRS detector	06/05/2019
M5	Completion of the neural network	07/07/2019
M6	Completion of documentation development	07/22/2019
M7	Completion of the presentation	09/03/2019
M8	Project Completion	09/03/2019

Table 11. Milestones of the project

ID	Milestone	Date
D1	Visualization GUI	03/04/2019
D2	QRS detector	06/05/2019
D3	Algorithms	07/07/2019
D4	Documentation of project memory	07/22/2019
D5	Presentation	09/03/2019

Table 12. Deliverables of the project

8.3 Gantt diagram

In Figure 16 we can see the Gantt diagram of the project.

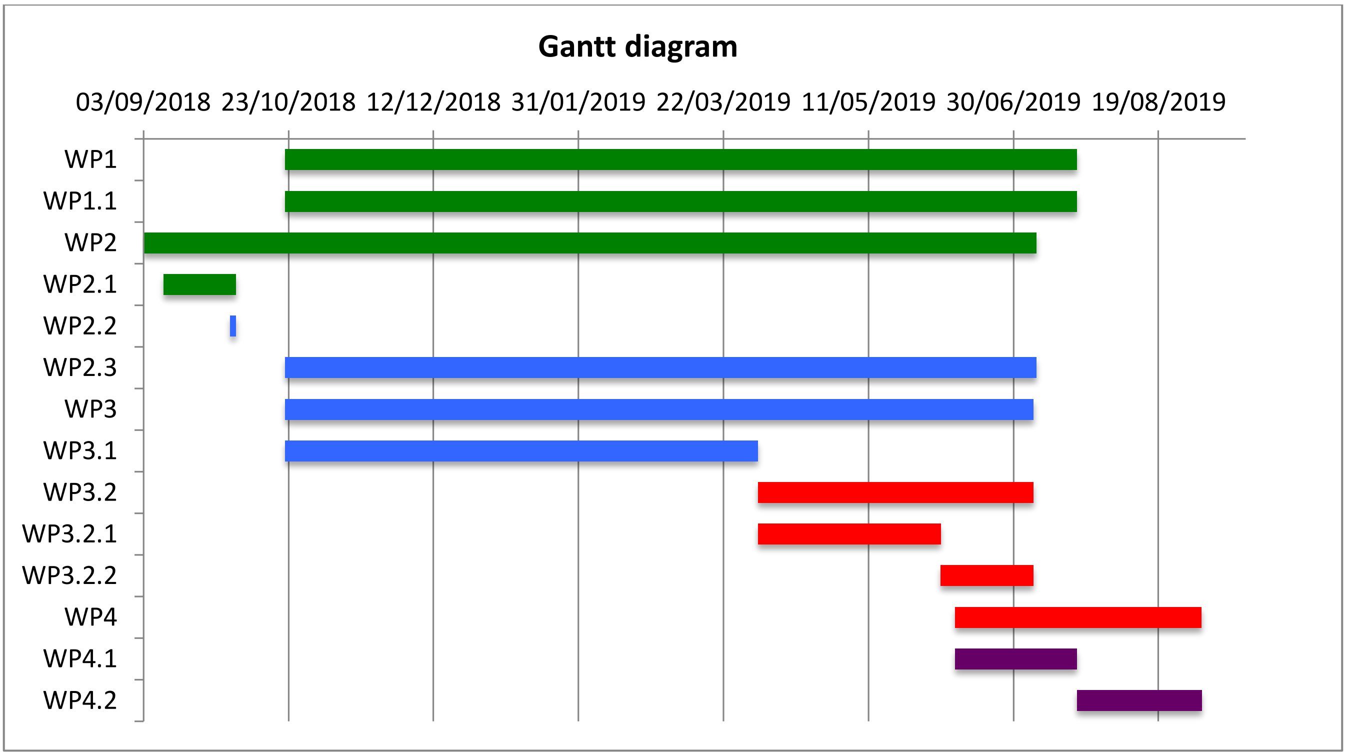


Figure 16. Gantt diagram

Universidad del País Vasco Euskal Herriko Unibertsitatea | BILBOKO INGENIARITZA ESKOLA ESCUELA DE INGENIERÍA DE BILBAO

9. Breakdown of expenses

In this section the breakdown of project expenses is presented. On the one hand, the cost of human resources is calculated and on the other, the subject of the project.

9.1 Human Resources

The salary of each one of the positions of the project is the following one:

Code	Position	Salary (€/h)
C1	Proyect director	60
C2	Junior Engineer	30

Table 13. Hourly salary of the members of the work team.

In the 11th table an economic balance of the human resources of the project in its entire time is presented:

Work package	C1		C2		TOTAL	
	Work hours	Cost (€)	Work hours	Cost (€)	Work hours	Cost (€)
WP1.1	20	1200	20	600	40	1800
WP2.1	15	900	30	900	45	1800
WP2.2	15	900	10	300	25	1200
WP2.3	10	600	20	600	30	1200
WP3.1	15	900	30	900	45	1800
WP3.2.1	10	600	30	900	40	1500
WP3.2.2	20	1200	30	900	50	2100
WP4.1	20	1200	60	1800	80	3000
WP4.2	10	600	20	600	30	1200
		TOTAL			385	15600

Table 14. Economic balance of the human resources.

9.2 Material resources

In the following tables the prices of the necessary materials are specified:

Material	Units	Inicial cost (€)	Lifespan (months)	Use (months)	Cost (€)
Personal desktop computer (with windows 7 system)	1	1200	84	12	171,43
Printer	1	400	60	0,5	3,33
Individual license of Matlab 2018a and toolboxes.	1	2300	12	9	1.725
Microsoft office word 2013	1	69	12	2	11,50
				Subtotal	1.911,26

Table 15. Material expenses.

In addition to the academic resources, we have had other expenses:

ID	Material	Cost (€)
MT1	Office supplies	50
MT2	Light	30
MT3	32 GB pendrive	15
	Subtotal	95

Table 16.Additional expenses.

9.3 Total Summary Data

The summary of the total expenditure is included in the following table, where human resources are included (working hours) and the costs of materials and material costs:

Concept	Cost (€)
Working hours	15600
Cost of materials	1.911,26
Material costs	95
Subtotal	17606,26

Table 17. Total expenses.

Taking into account all the expenses, the total cost of the project has been of **seventeen thousand six hundred and six with twenty-six** euros. Most of the total cost belongs to the working hours.

10. Conclusions

As mentioned throughout the document, the objective of this project was to examine and test an algorithm for the detection of atrial fibrillations in short electrocardiograms.

Throughout the development of the project, the points established in the scope of the project have been completed. Beginning with the development of the QRS automatic detector, the detection algorithm has been developed and implemented.

Once the first results have been obtained, a correction has been made to establish the correct parameters for the algorithm. In addition, with the parameters selected as optimal, the algorithm has a very good result.

The precision provided by the algorithm in the detection of atrial fibrillation makes it a software that, in addition to being easily implantable both in health centres and hospitals, provides great precision when detecting atrial fibrillation.

With the development of this project it is intended to obtain a series of technical, economic and social benefits. For example, the main technical contributions of this work are:

- A low cost easily deployable solution has been developed
- This solution can be integrated into current ECG monitoring equipment adding a new functionality.
- It supposes a competitive advantage with respect to other manufacturers.

Also, this project will result in economic benefits for public health. These benefits are:

- It is a software solution that does not require additional investment.
- It will make the diagnosis faster and more economical, both in material resources and in time.

Finally, a daily action, medically speaking, as doing an ECG can detect the appearance of pathology that if not treated, in the long term it can become serious is good news. In addition to simple and cheap, it will make a large part of these cases to be treated on time.

References

[1] Revista Española de la Cardiología, “https://www.revespcardiol.org/es-epidemiologia-fibrilacion-auricular-articulo-S1131358716300073,,” January 2016. [Online].

[2] Revista Colombiana de Cardiología, “https://www.elsevier.es/en-revista-revista-colombiana-cardiologia-203-articulo-costos-asociados-fibrilacion-auricular-S012056331630184X,” December 2016. [Online].

[3] Mayo Clinic, “https://www.mayoclinic.org/diseases-conditions/atrial-fibrillation/symptoms-causes/syc-20350624,” December 2018. [Online].

[4] V. Fuster, L. E. Ryden, R. W. Asinger, D. S. Cannom, H. J. Crijns, R. L. Frye, J. L. Halperin, G. N. Kay, W. W. Klein, S. Levy, R. L. McNamara, E. N. Prystowsky, L. S. Wann and D. G. Wyse, “ACC/AHA/ESC guidelines for the management of patients with atrial fibrillation: executive summary,” 2001, pp. 1231-1265.

[5] D. L. A. -, “https://www.fbbva.es/microsites/salud_cardio/mult/fbbva_libroCorazon_cap4.pdf,” Hospital Clínico San Carlos, Madrid. [Online].

[6] Clínica Universidad de Navarra, “https://www.cun.es/enfermedades-tratamientos/pruebas-diagnosticas/electrocardiograma,” [Online].

[7] R. A. Alvarez, “http://www.esei.uvigo.es/fileadmin/docs/estudiantes/Traballos_fin_de_carreira/TFM_SSIA_2012-13/A_comparison_of_three_QRS_detection_algoritms_over_a_public_database.pdf,” 13 May 2013. [Online].

[8] Fundación Española del Corazón, “https://fundaciondelcorazon.com/informacion-para-pacientes/metodos-diagnosticos/electrocardiograma.html,” [Online].

[9] C.-L. Chen and C.-T. Chuang, “https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5621148/,” A QRS Detection and R Point Recognition Method for Wearable Single-Lead ECG Devices, 6 August 2017. [Online].

[10] O.Pahlm and L.Sörnmo, “Software qrs detection in ambulatory monitoring,” in *Medical and Biological Engineerinf and Computing*, 1984, pp. 289-297.

[11] G. Friesen, T. Jannett, M. Jadallah, S. Yates, S. Quint and H. Nagle, "G. Friesen, T. Jannett, M. Jadallah, S. Yates, S. Quint, and H. Nagle, "A comparison of the noise sensitivity of nine qrs detection algorithms," Biomedical Engineering, IEEE Transactions on, vol. 37, no. 1,," 1990, p. 85–98.

[12] N. Arzeno, Z.-D. Deng and C.-S. Poon, ""Analysis of first-derivative based qrs detection algorithms," Biomedical Engineering, IEEE Transactions on, vol. 55, no. 2," 2008, p. 478–484.

[13] W. J. T. T. N. a. S. L. V. Afonso, ""Ecg beat detection using filter banks," Biomedical Engineering, IEEE Transactions on, vol. 46, no. 2,," 1999, p. 192–202.

[14] M. Okada, ""A digital filter for the qrs complex detection," Biomedical Engineering, IEEE Transactions on, vol. BME-26, no. 12," 1979. , p. 700–703.

[15] J. P. a. W. J. Tompkins, ""A real-time qrs detection algorithm," Biomedical Engineering, IEEE Transactions on, vol. BME-32," 1985.

[16] P. S. H. a. W. J. Tompkins, ""Quantitative investigation of qrs detection rules using the mit/bih arrhythmia database," Biomedical Engineering, IEEE Transactions on, vol. BME-33, no. 12," 1986, p. 1157–1165.

[17] R. A. S. O. A. R. a. P. L. J. Martinez, ""A waveletbased ecg delineator: evaluation on standard databases," Biomedical Engineering, IEEE Transactions on, vol. 51, no. 4," 2004, p. 570–581.

[18] C. Z. a. C. T. C. Li, ""Detection of ecg characteristic points using wavelet transforms," Biomedical Engineering, IEEE Transactions on, vol. 42, no. 1," 1995, p. 21–28.

[19] B. A. a. H. Seo, ""A new qrs detection method using wavelets and artificial neural networks," Journal of Medical Systems, vol. 35, no. 4," 2011, p. 683–691.

[20] Y. H. a. W. J. T. Q. Xue, ""Neural-network-based adaptive matched filtering for qrs detection," Biomedical Engineering, IEEE Transactions on, vol. 39, no. 4," 1992 , p. 317–329.

[21] S. C. a. G. V. R. Poli, ""Genetic design of optimum linear and nonlinear qrs detectors," Biomedical Engineering, IEEE Transactions on, vol. 42, no. 11," 1995, p. 1137–1141.

[22] K. Harris, D. Edwards and J. Mant, "https://www.ncbi.nlm.nih.gov/pubmed/22518390," 2012. [Online].

[23] Embedded Vision, “https://www.embedded-vision.com/,” [Online].

[24] K. Willems, “https://www.datacamp.com/community/tutorials/deep-learning-python,” 4 February 2019. [Online].

[25] Ö. Genç, “https://towardsdatascience.com/notes-on-artificial-intelligence-ai-machine-learning-ml-and-deep-learning-dl-for-56e51a2071c2,” 26 January 2019. [Online].

[26] S. O. Ali Isin, “Cardiac Arrhythmia detection using deep learning,” *ScienceDirect,* pp. 268-275, 2017.

[27] NCSS2019 Data Analysis, “https://www.ncss.com/software/ncss/diagnostic-tests-in-ncss/,” 2019. [Online].

[28] “https://en.wikipedia.org/wiki/Sensitivity_and_specificity,” [Online].

[29] PhysioNet, “https://physionet.org/challenge/2017/,” 27 April 2017. [Online].

[30] G. Mora-Pabón, “Evaluación de la fibrilación auricular mediante electrocardiograma y Holter,” *Revista Colombiana de Cardiología,* pp. 27-33, 2016.