Assaraf David Septembre 2020

Research Internship Report, PA MAP

I. Introduction

I spent most of my third year at Ecole Polytechnique applying for Universities for my master, and applying for internships in Data Science. For my master, I wanted to reach Harvard or Stanford because those were the only two leading universities providing a program in Computational Data Science. For the internship, I wanted to work on Data Science in Israel. There were several opportunities for these kinds of job, but Israeli people were not very accustomed with the concept of internships. I was in process the the Amazon Research Laboratory in Israel, IBM Research in Haifa and AIM Lab in the Technion. I gave up IBM Research because the missions there were more focused on cloud-architecture and did not involve the use of statistics. For Amazon Research Lab, they wanted me to work with Generative Adversarial Networks, on which I have done my research project in third year with Pierre-Henry Labordère. AIM Lab was in the need of a researcher in order to work in the optimization of Machine Learning algorithms leveraging large databases, in particular XGBoost. They contacted me after having heard that I have been implementing several kinds of optimizers for my Neural Networks (including Particule Swarm Optimization, a heuristic optimizer that becomes widespread these days). Since we could not come to an agreement with Amazon Research, I decided to accept to work with Joachim Behar, leading the AIM Lab. He is a 34 years old Professor, the youngest professor to lead a research laboratory in the Technion, former student of *Les Mines de Saint-Etienne*, phD at Oxford Univeristy (working with Gari Clifford) and Post-doc at the Technion University. His laboratory was in BioMedical Engineering, developing softwares leveraging AI in order to help the decisions of different experts. Because of the Covid, my internship switched and I worked on the PhysioNet competition. During my 6-months long remote internship, I had the opportunity to compete with teams from all over the world, to lead a team of professional researchers and to learn a lot from their expertise. I believe those were the richest and most intense 6 months of my life so far. This competition ended up with our team being invited to present our work to the 2020 Computing in Cardiology Conference and **winning the award of best oral presentation**. Furthermore, we published a paper on *IEEE Physiological Measurement* and *Focus* has invited us to publish our work. Our final ranking in the competition was 41/400 teams (although we could have done so much better if we could not have submission issues). Overall, I am very glad to have been able to pull all of this work off. I have been working on Signal Processing, Feature Selection methods, Architecture methods, Ensemble Learning, Cardiology theory … Whether on a theoritical or a computational point of view, I know that I am now more confident on my capabilities and can begin my Data Science Master in Harvard peacefully.

In this work, I will try to introduce you to the Physionet Competition (Task, Databases, scoring metrics), to our team's approach (Technion_AIMLab), and my experience writing papers and being speakers at the conference.

I hope you will enjoy the reading as much as I enjoyed working on these topics.

II. AIM Lab

The AIM Lab is a BioMedical Engineering laboratory located at the Technion in the department of Electrical Engineering. This laboratory is led by Prof. Joachim Behar Oxford PhD in BioMedical Engineering. The research led at the laboratory is focused on Covid-19, Sleep medicine, Cardiology and fetal cardiography. The team is composed of ~15 Technion researchers, phD candidates of MsC candidates. They work as Research assistants at the lab and all work on a research thesis and are supervised by Joachim Behar. In the lab, there are several IT people/administrative manager that help researchers with administrative/IT questions. The lab has some accomodations for researchers they provide remote computers (in addition to your laptop) in order to be twice as productive, SSH servers with 3*64 cores in order to launch your experiments on a more powerful server while keeping your computer safe. I have grown used to SSH and its benefits. The Lab provides tutorial on how to use PyCharm efficiently for experiments and SSH: I have learnt that too. Every Wednesday, the team gathers and one researcher presents its work to the entire laboratory, in order for others to maybe gain some insights and share their thoughts on the approach. I presented four times the work on which I was working: the first time, I presented my research work on Duality and Wasserstein Generative Adversarial Newtorks, the two next times were in order to present my work in the competition. The last time, Joachim gave me the opportunity to simulate my presentation at the Rimini coneference. I believe this is a great practice in order to build a strong knowledge for researchers: they become accute to several hot research topics in BioMedical Engineering. Unfortunately, due to travel restrictions to Israel, I have not been able to make it to Israel and meet them in person.

III. The PhysioNet Competition

Initially, I was planning on doing my internship working in Optimization. Armand Chocron, working with Assist. Prof. Joachim Behar, was working on the detection of Atrial Fibrillation and different types of sleep apnea. Working with very large datasets (requiring ~80.000 hours of training), he needed someone to help him accelerate his training process and also implementing XGBoost at a scalable level (the availables python libraries available for XGBoost, xgboost or scikit-learn) were very computationally-expensive and the GPUs/CPUs available at the laboratory were already overloaded. Before the official beginning of my internship, I did an extensive research work on XGBoost (that has been introduced in MAP569). In particular, the use of XGBoost in the medical context was very common because of the several papers that have demonstrated the effectiveness of this algorithm and its comparative performances with DL classifiers. Furthermore, some libraries like ELI5 allowed to obtain interpretable results analyzing the results produced by XGBoost, being a large advantage over the Deep Learning Approach (or Representation Learning). However, because of the COVID, I could not go to Israel to do my internship and could not work with Armand on his work.

My internship has therefore been changed to the PhysioNet Competition. The PhysioNet competition is a competition hosted every year by the MIT in coordination with the Computing in Cardiology Conference (referred to CinC). This competition is gathering every year more than 500 teams from all over the world. The teams are invited to work on a common research work, deeply linked to computational cardiology concepts. Teams are often related to a BioMedical Engineering laboratory, and composed of professional researchers working on BioMedical Engineering.

This competition is organized by researchers from the MIT and hosted on a collaborative way (participants are invited to point out to potential improvements of the competition, scoring metrics,

problems in the database, …). Participants are ranked based on the model they submit that is scored on a hidden test set according to a metric decided by the competition. There have been many constraints related to the submission:

- We should submit according to very specific instructions: the training should be done in an online fashion, with a required format for the output
- Reproducibiliy: we should include our training code in order for the organizers of the competition to make sure that our code is reproducible and our model will be evaluated according to the performances of the re-trained model
- Our code should be specifically in Python or Matlab
- Our code should be submitted in a Docker environment
- Our training should not last more than 24 hours
- Any data processing should be included in the training process (include data relabelling, signal processing, …)
- We could not have more than 5 submissions across the 5 months (a rejected submission because our format was not right counts for a submissions)

Therefore, the format of the competition was very complicated to handle, since I barely had any experience with Python, and never used nor Matlab nor Docker.

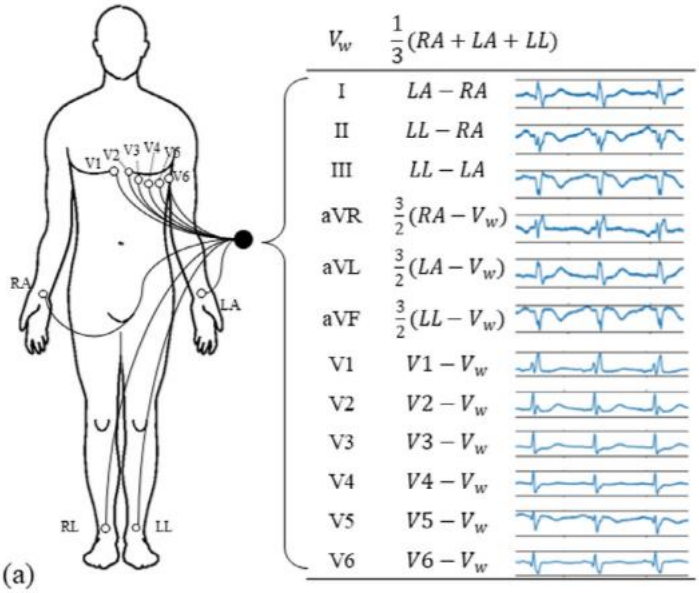
When Joachim proposed me to work on the competition, I directly accepted: this was the opportunity for me to learn a lot of things and this was adapted to the format of my internship (remote). Therefore, a team of 4 researchers enrolled with me : Jeremy Levy, Gal Yefet and Armand Chocron, PhDs at the Technion University and Janmajay Singh, working from Japan. On the team, I was the only one working full time on this project. Therefore, we decided that I would lead the research of our team and be responsible for the schedule of our work and the interaction with the organizers of the competitiojn (being in particular Gari Clifford).

Every year, the Competition was defined by three things:

- The Classification task
- The data
- The scoring metrics

1. The Classification Task

The Classification task of this year's competition was: **The Classification fo 12-leads ECG.**



Location of the different 12-leads ECG in the human body

2. The Data

The data for this Challenge is from multiple sources:

- Southeast University, China, including the data from the China Physiological Signal Challenge 2018
- St. Petersburg Institute of Cardiological Technics, St. Petersburg, Russia
- The Physikalisch Technische Bundesanstalt, Brunswick, Germany
- Georgia 12-Lead ECG Challenge Database, Emory University, Atlanta, Georgia, USA.

The first source is the public and unused data from the China Physiological Signal Challenge in 2018 (CPSC2018), held during the 7th International Conference on Biomedical Engineering and Biotechnology in Nanjing, China. This training set consists of two sets of 6,877 (male: 3,699; female: 3,178) and 3,453 (male: 3,453, female: 1,610) of 12-ECG recordings lasting from 6 seconds to 60 seconds. Each recording was sampled at 500 Hz.

The second source set is the public dataset from St Petersburg INCART 12-lead Arrhythmia Database. This database consists of 75 annotated recordings extracted from 32 Holter records. Each record is 30 minutes long and contains 12 standard leads, each sampled at 257 Hz.

The third source from the Physikalisch Technische Bundesanstalt (PTB) comprises two public databases: the PTB Diagnostic ECG Database and the PTB-XL, a large publicly available electrocardiography dataset. The first PTB database contains 549 records (male: 377, female: 139). Each recording was sampled at 1000 Hz. The PTB-XL contains 21,837 clinical 12-lead ECGs (male: 11,379 and female: 10,458) of 10 second length with a sampling frequency of 500 Hz.

The fourth source is a Georgia database which represents a unique demographic of the Southeastern United States. This training set contains 10,344 12-lead ECGs (male: 5,551, female: 4,793) of 10 second length with a sampling frequency of 500 Hz.

Now, we can see a first difficulty in the data provided: the plurality of sources. The data came from 5 different hospitals from all over the world. Several problems arise with this kind of datasets:

- The quality of the labels: some datasets have produced the labels via an automatic software, thus having low quality and not always being of our advantage to include them our training
- The non-universality of the labelling: practices may vary between hospitals in the detection of pathologies in electrocardiograms. Since some pathologies are quite hard to dissociate between others, some labelling practices may vary according to which hospital/ which continent you are on
- Consistency of sample rate: across databases, the sampling rates allowing to get the ecgs are not consistent (some have a sample rate of 500Hz, others of 257Hz): this will be hard to work on Feature Extraction with various sample rates.
- Consistency of lengths: The INCART database had recordings of 30-minutes long. Creating models capable of analyzing such long-segments is a completely different task from producing models working with 10-seconds long recordings. Furthermore, how can we label a recording of 30 minutes long if he only has one pathological beat ?
- Generalizability: we do not know what the composition of the test set will be (in terms of geographical repartition)

Each ECG recording has a binary MATLAB v4 file for the ECG signal data and a text file in WFDB header format describing the recording and patient attributes, including the diagnosis.

```
A0001 12 500 7500 05-Feb-2020 11:39:16
A0001.mat 16+24 1000/mV 16 0 28 -1716 0 I
A0001.mat 16+24 1000/mV 16 0 7 2029 0 II
A0001.mat 16+24 1000/mV 16 0 -21 3745 0 III
A0001.mat 16+24 1000/mV 16 0 -17 3680 0 aVR
A0001.mat 16+24 1000/mV 16 0 24 -2664 0 aVL
A0001.mat 16+24 1000/mV 16 0 -7 -1499 0 aVF
A0001.mat 16+24 1000/mV 16 0 -290 390 0 V1
A0001.mat 16+24 1000/mV 16 0 -204 157 0 V2
A0001.mat 16+24 1000/mV 16 0 -96 -2555 0 V3
A0001.mat 16+24 1000/mV 16 0 -112 49 0 V4
A0001.mat 16+24 1000/mV 16 0 -596 -321 0 V5
A0001.mat 16+24 1000/mV 16 0 -16 -3112 0 V6
#Age: 74
#Sex: Male
#Dx: 426783006
#Rx: Unknown
#Hx: Unknown
#Sx: Unknown
```

Example of a file's header, giving metadata information

From the first line, we see that the recording number is A0001, and the recording file is A0001.mat. The recording has 12 leads, each recorded at 500 Hz sample frequency, and contains 7500 samples. From the next 12 lines, we see that each signal was written at 16 bits with an offset of 24 bits, the amplitude resolution is 1000 with units in mV, the resolution of the analog-to-digital converter (ADC) used to digitize the signal is 16 bits, and the baseline value corresponding to 0 physical units is 0. The first value of the signal, the checksum, and the lead name are included for each signal. From the final 6 lines, we see that the patient is a 74-year-old male with a diagnosis (Dx) of 426783006. The medical prescription (Rx), history (Hx), and symptom or surgery (Sx) are unknown.

Initially, there was 127 labels in our training data. However, some of them was 'unscored'. This means that, in the scoring metrics that I will present later, misclassification of an example as one of these unscored labels did not penalize us. Furthermore, the output of our classifier on some 'unscored' recordings was not taken into account. We will see later how we did handle these unscored labels.

Here are the scored labels:

Dx	Abbreviation	CPSC	CPSC-Extra	StPetersburg	PTB	PTB-XL	Georgia	Total
1st degree av block	IAVB	722	106	0	0	797	769	2394
atrial fibrillation	AF	1221	153	2	15	1514	570	3475
atrial flutter	AFL	0	54	0	1	73	186	314
bradycardia	Brady	0	271	11	0	0	6	288
complete right bundle branch block	CRBBB	0	113	0	0	542	28	683
incomplete right bundle branch block	IRBBB	0	86	0	0	1118	407	1611
left anterior fascicular block	LAnFB	0	0	0	0	1626	180	1806
left axis deviation	LAD	0	0	0	0	5146	940	6086
left bundle branch block	LBBB	236	38	0	0	536	231	1041
low qrs voltages	LQRSV	0	0	0	0	182	374	556
nonspecific intraventricular conduction disorder	NSIVCB	0	4	1	0	789	203	997
pacing rhythm	PR	0	3	0	0	296	0	299
premature atrial contraction	PAC	616	73	3	0	398	639	1729
premature ventricular contractions	PVC	0	188	0	0	0	0	188
prolonged pr interval	LPR	0	0	0	0	340	0	340
prolonged qt interval	LQT	0	4	0	0	118	1391	1513
qwave abnormal	QAb	0	1	0	0	548	464	1013
right axis deviation	RAD	0	1	0	0	343	83	427
right bundle branch block	RBBB	1857	1	2	0	0	542	2402
sinus arrhythmia	SA	0	11	2	0	772	455	1240
sinus bradycardia	SB	0	45	0	0	637	1677	2359
sinus rhythm	NSR	918	4	0	80	18092	1752	20846
sinus tachycardia	STach	0	303	11	1	826	1261	2402
supraventricular premature beats	SVPB	0	53	4	0	157	1	215
t wave abnormal	TAb	0	22	0	0	2345	2306	4673
t wave inversion	TInv	0	5	1	0	294	812	1112
ventricular premature beats	VPB	0	8	0	0	0	357	365

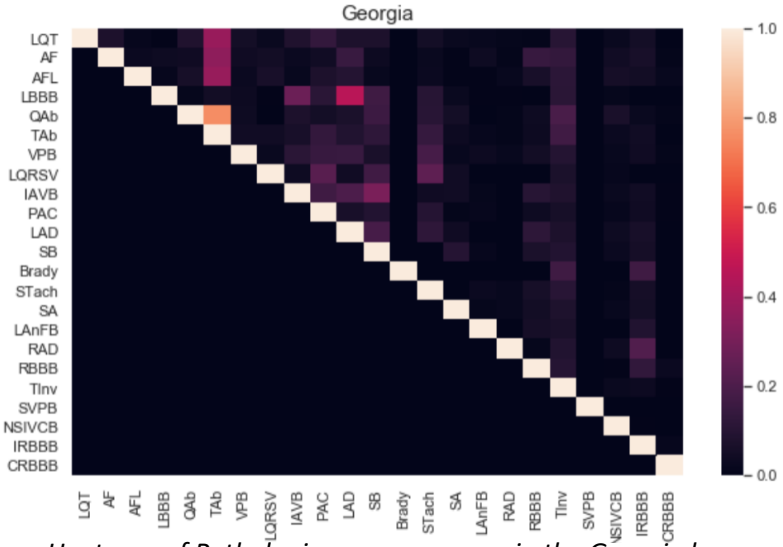
Table gathering the 27 pathologies and their repartition among Databases

Therefore, our task is a very complicated one:

- Classify **27 pathologies** (the biggest Classification task in a open-source competition until then)
- Levergae a Database with more than 40.000 recordings more that 480.000 electrocardiogram leads: the biggest public ecg database released until then
- Work with some very uncommon pathologies (PR, VPB) for which no research has been made
- Work with a Database with highly unbalanced classes (314 AFL labels vs more that 20k Normal Sinus Rhythms labels)
- **Biggest complexity of our problem: MultiLabel classification.** Indeed, examples in our Database had several labels. For instance, one training recording might be labelled simultaneously as SRN, AFL, SVPB …

Dataset Name	2	3	4	5	6	7	8
CPSC-1	448	6	0	0	0	0	0
CPSC-2	708	348	133	32	7	1	0
Georgia	2316	1797	1167	538	236	79	37
PTB	7	4	1	0	0	0	0
PTB-XL	5124	4119	2937	1533	689	279	85
INCART	4	9	12	3	3	0	1

Table with number of recordings having several number of diagnoses, per bases



Heatmap of Pathologies co-occurences in the Georgia base

3. The Scoring Metrics

The last thing in order to completely specify the competition was the scoring metrics that has been used in order to evaluate our models. The one used by the competition was a very uncommon one, trying to reproduce the real-life diagnosis. Indeed, in real-life, the misclassification between some pathologies might be not too harmful (for instance between Complete Right-Bundle Branch Block and Right Bundle Branch Block) and some others might be very severe (misclassification between SNR and Atrial Fibrillation).

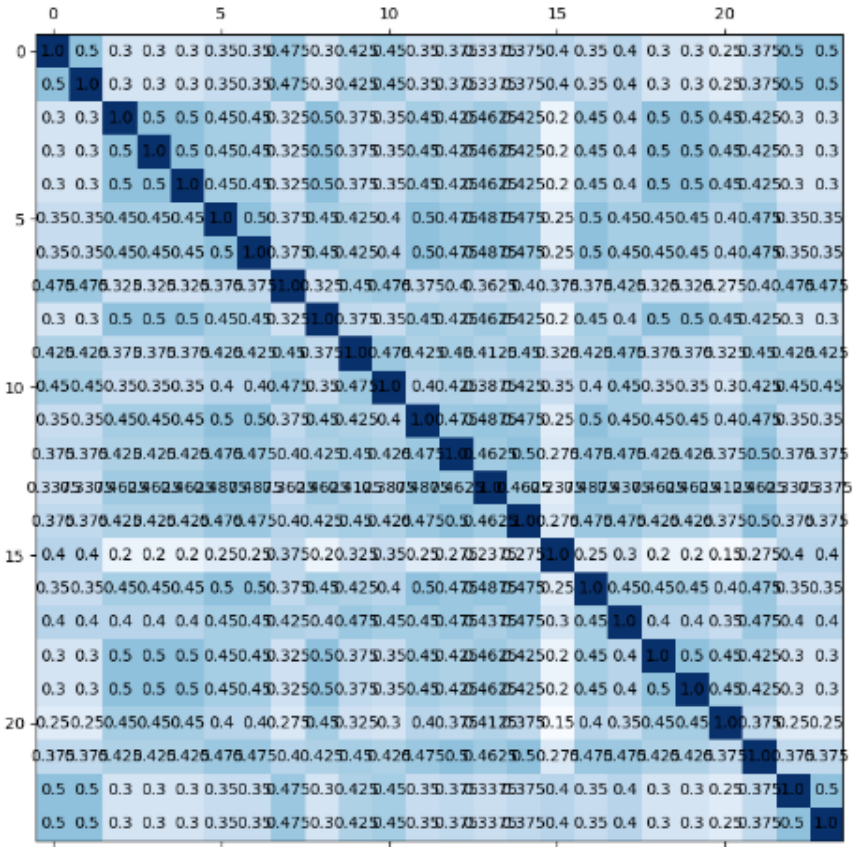
Furthermore, among these 27 pathologies, there are groups of pathologies that are considered exactly the same by the scoring metrics:

- PVC and VPB
- PAC and SVPB
- CRBBB and RBBB

This means that misclassifying CRBBB as RBBB for example is not harmful at all, no difference.

Therefore, there are **24 different scored pathologies (we re-labelled VPB as PVC, SVPB as PAC and CRBBB as RBBB).**

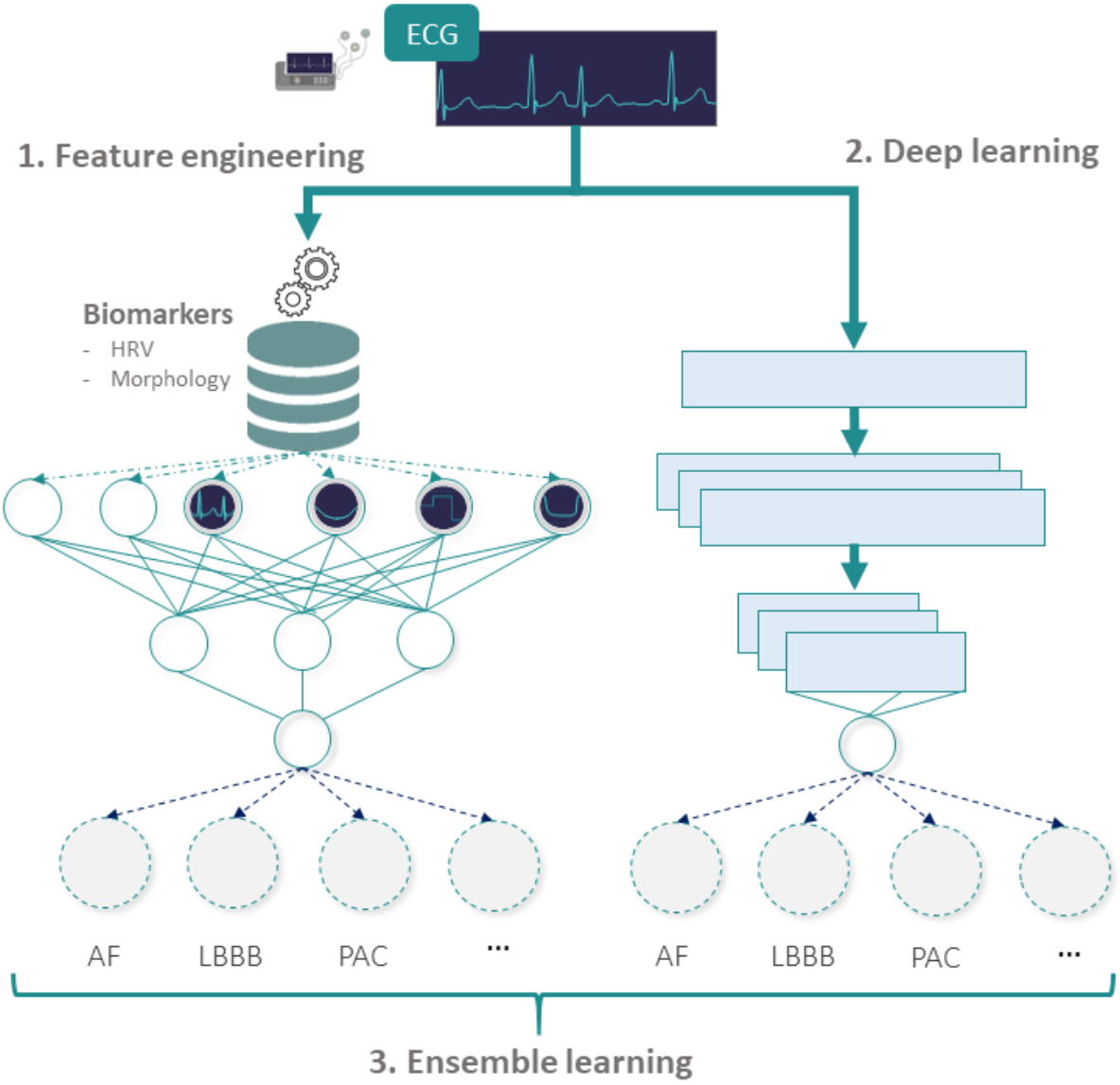
The competition introduced a **weight matrix:**



Weight matrix in order to compute the score

Basically, you compute the confusion matrix of your model and then you sum the scalar products columnwise with the columns of the weight matrix (therefore: high value in the weight matrix means that the misclassification between both pathologies is not very serious).

In order to tackle this Classification process, we have defined an objective model that would ultimately be able to perform Multi-Label Classification from a unbalanced DataSet with 480.000 leads recordings. Here is our model



Picture of the model we wanted to create for our Classification task

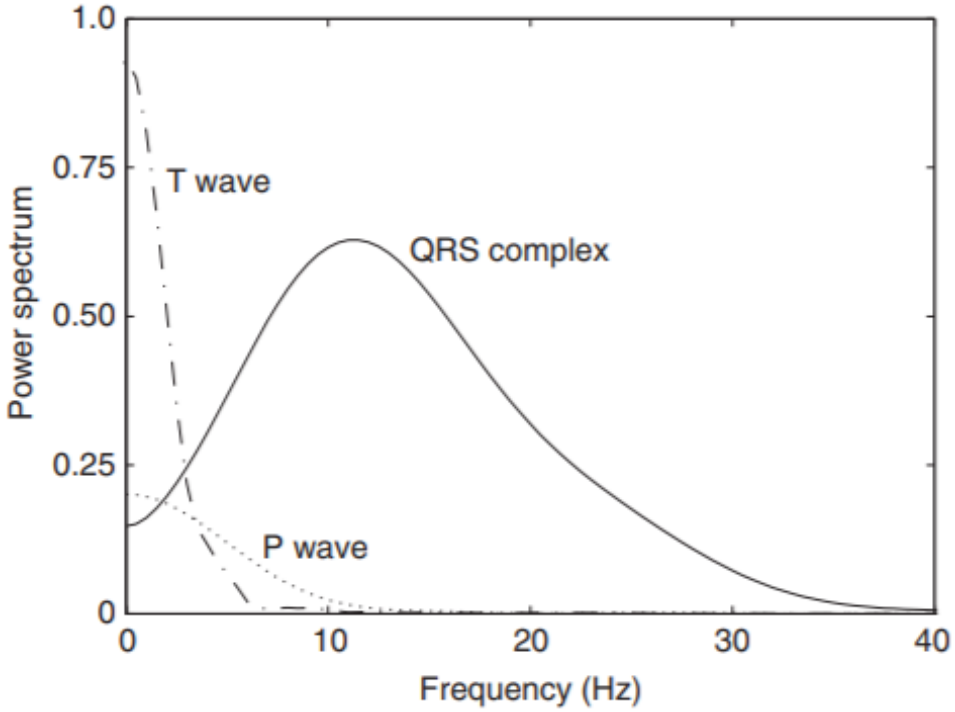
I would focus alone on Feature Engineering and then work with Jeremy, Gal and Jamajay on Deep Learning. But first, we needed to perform some Data Processing steps.

IV. Data Processing

First, we did an extensive work trying to prepocess the recordings from our database. This was done in order to remove the noise inherent to the recording and the movements of the body, and has been implemented through the use of several filters.

I will here present the main causes of ecg noises and the several solutions that have been implemented in order to suppress that noise.

1. Main ECG Noises and Filter Implemented to remove them



Characteristic frequencies in the ecg

A. Baseline wandering

Removal of baseline wander is required in order to minimize changes in beat morphology that do not have cardiac origin, which is especially important when subtle changes in the "low-frequency" ST segment are analyzed for the diagnosis of ischemia. The frequency content of baseline wander is usually in the range below 0.5Hz. However, increased movements of the body during the latter stages of a stress test further increases the frequency content of baseline wander.

The design of a linear, time-invariant, highpass filter for removal of baseline wander involves several considerations, of which the most crucial are the choice of filter cut-off frequency and phase response characteristic. The cutoff frequency should obviously be chosen so that the clinical information in the ECG signal remains undistorted while as much as possible of the baseline wander is removed.

There are two possible approaches:

- Find the lowest frequency component of the ECG spectrum. In general, the slowest heart rate is considered to define this particular frequency component.

- Couple the cut-off frequency to the prevailing heart rate: Linear filtering with time-variable cut-off frequency.

The phase response is also an important consideration. This will dictate our choice of filter design. We do not want any phase distorsion. This is why we will use Forward-Backward IIR filtering.

Once we would have found the best cut-off frequency for the removal of baseline wandering, we will implement a wavelet technique, decomposing the signal according to several frequencies. We will only retain the frequencies above the cut-off frequency. It is argued that this method is better than using a Highpass filter (the phase is not disturbed).

B. Powerline interference

First, a few definitions:

- An adaptive filter presents the propriety of estimating the present noise according to previous estimations. Therefore, the transfer function of the filter is not defined in a static way, it is always defined in order toapproximate in a better way the noise.
- A non adaptive filter is a static filter. You define a priori the design, order of your filter and the signal will be filtered no matter what specificity is encountered during the filtering process.
- Finite Impulse Response filter (FIR) is a discrete time filter, which characterizes its response only with a finite number of input signal's values.
- Infinite Impulse Response filter (IIR) is a continuous time filter, which characterizes its response only with a infinite number of input signal's values

Electromagnetic fields caused by a powerline represent a common noise source in the ECG that is characterized by 50 or 60Hz sinusoidal interference, possibly accompanied by a number of harmonics. It is said that 50Hz interference is for European ECGs whereas 60Hz interference is for American ECGs (**another source of difficulty for our work, due to the fact that the signals come from several locations**). Since our data comes from 11 different hospitals, we will need to try the powerline interference interference for each frequency separately, and then altogether (should not pose problems since we will be using adaptive filters) .Such narrowband noise renders the analysis of the ECG more difficult. A major concern when filtering out powerline interference is the degree to which the QRS complexes influence the output of the filter. The QRS complex acts, in fact, as an unwanted, large-amplitude impulse input to the filter. As linear, time-invariant notch filters are generally more sensitive to the presence of such impulses, powerline filters with a non linear structure may be preferable. This is argued in several experiments where adaptive and non adaptive 60Hz notch filters' performances are compared. The experiments revealed that using Ahlstrom and Tompkins' filter allowed to reduced the signal entropy more than when using a non adaptive filter.

C. Muscle Artifacts = Electromyography Noise

Electrocardiogram recordings are very often contaminated by EMG disturbances due to involuntary muscle contractions (tremor). In order to suppress those high frequency components, it is advocated that using a Savitzky-Golay filter helps to remove this high-frequency noise, and also removing the high-frequency noise. This filter combines the strength of Moving Averages with splines fitting \cite{Savgol}. The parameters of this filter are: polynomial degree and number of points in the window considered. It is suggested to use a degree 3 polynomial approximation. We will fine-tune the window's number of points. Once again, the pathologies "at risk" when smoothing the signal are PAC and PVC.

D. Patient-Electrod Motion Artifacts

Motion artifacts are baseline changes which are caused by electrode motion. Usually vibrations, movement, or respiration of the subject contribute to motion artifacts. The peak amplitude and duration of the artifact depend on various unknown quantities such as the electrode properties, electrolyte properties, skin impedance, and the movement of the patient. In ECG signal, the baseline drift occurs at an unusually low frequency (approximately 0.014Hz), and most likely results from very slow changes in the skin-electrode impedance. This noise can also be observed on the Fourier power spectrum, the large peak nearest to DC. We will see if we effectively remove these Artifacts by analyzing the Spectrum of our signal after preprocessing.

E. Contact noise

Position of the heart with respect to the electrodes (variation) and changes in the propagation medium between the heart and the electrodesinitiateElectrode contact noise.This causes sudden changes in the amplitude of the ECG signal, and low frequency baseline shifts. In addition, poor conductivity between the electrodes and the skin both reduces the signal amplitude of the ECG signal and thereby increases the probability of disturbances (by reducing SNR). The mechanism responsible for baseline disturbances is electrode-skin impedance variation. The larger the electrode-skin impedance, smaller are the relative impedance change which is required to cause a major shift in the baseline of the ECG signal. If the skin impedance is significantly high, it might be impossible to detect the signal features reliably in the presence of body movement. Sudden changes in the skin-electrode impedance induce sharp baseline transients which decay exponentially to the baseline value. This transition may occur only once or rapidly several times in succession. Amplitude of the initial transition and the time constant of the decay are the major characteristics of such noise. The solution commonly designed to remove such Artifacts is a lowpass filter with a cutoff frequency of around 100Hz. Therefore, we will try to implement several lowpass filters with different cutoff frequencies. The pathologies at risk when using a lowpass filter are the ones leveraging R-peaks amplitudes: PAC and PVC.

Until now, we implemented the classic methods of noise removal without taking into account the specificity of our ECG. We will use several decomposition techniques in order to assess the quality of several filtering techniques: the discrete Fourier transform, allowing us to visualize the different frequencies present in the signal, and the Welch approximation of the signal's Power Spectral Density. As a model, the power spectral density of an ECG should contain the following patterns:

2. A measure of quality: signal quality

In addition to performing visual checks for assessing that our filter has the frequency response I expected. I needed a quantitative measure in order to assess the quality of a filtering method. Furthermore, some filters did not have a ‘visible’ frequency response: the Savilsky-Golay filter used signal-smoothing spline functions in order to remove the unwanted noise in our signals: can’t be seen directly in a frequency response fashion.

Therefore, I used the physiological-specific **signal quality measure.** The idea behind this quantity is that I will use two different R-peaks detector and, basd on their different detections, I will estimate the quality of a signal. This intuition is displayed in the following graphic.

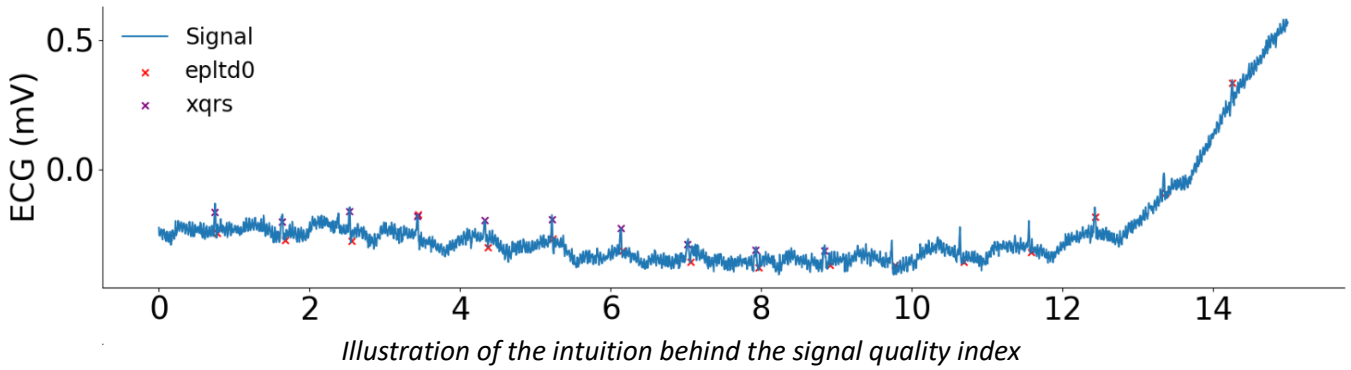
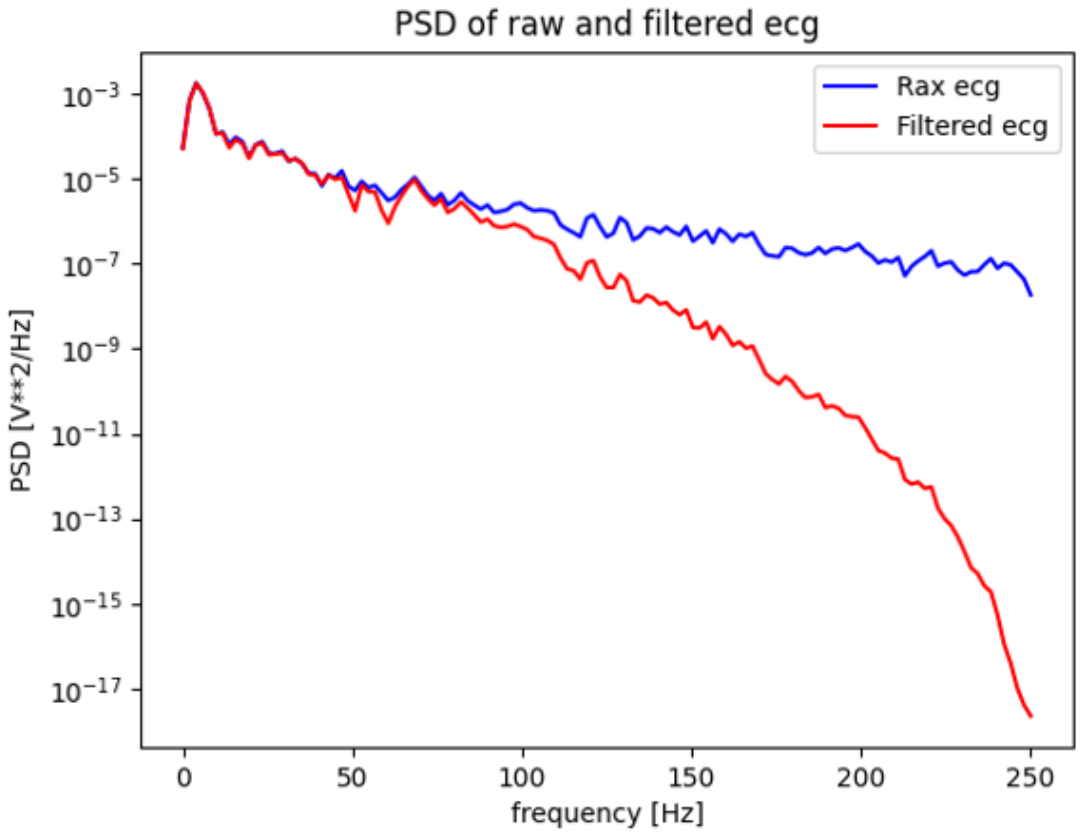


Illustration of the intuition behind the signal quality index

This measure of a signal quality allowed us select the final filter we used for signal processing:

- Low-pass FIR filter with higcut frequency at 0.05 Hz
- Notch IIR filter at 50Hz or 60Hz depending on the frequency of the powerline (estimated via the Welch method)
- Savilsky-Golay smoothing filter of order 3 with a window len of len(signal)/17

The have displayed here the PSD of our filter.



Power Spectrum decomposition of the final filter I used, using Welch method

3. Last step of data processing: Resampling, Split of signals and selection of Training Examples

The last step of data processing we used, before extracting features are the following:

A. Dealing with different sample rates

We pointed out on the presentation of Databases that the different signals from our databases had different sample rates. This was making our task difficult to extract relevant features across databases (either for feature Engineering or Reprensetation Learning). Therefore, we decided to unofrmly resample the signals to 500Hz. This was done in MATLAB using a processing package.

B. Dealing with signals of different lengths

I also pointed out in the presentation of Databases that we needed to deal with signals that were 10s-long and others that were 30-minutes long (from the INCART Database). Dealing with signals that were 30-minutes long was more difficult since sometimes only one pathological beat could render the entire signal labelled as pathological. We decided to split these 30-minutes long signals into several signals of 10 seconds. However, we could not label uniformly all of these signals (since some portions of 10 signals will not be pathological and others will be pathological). In this case, we used an automatic software labelling electrocardiograms. We know that the quality of these labels was inferior, so we assigned less weight to these examples in our training.

C. Dealing with unscored labels

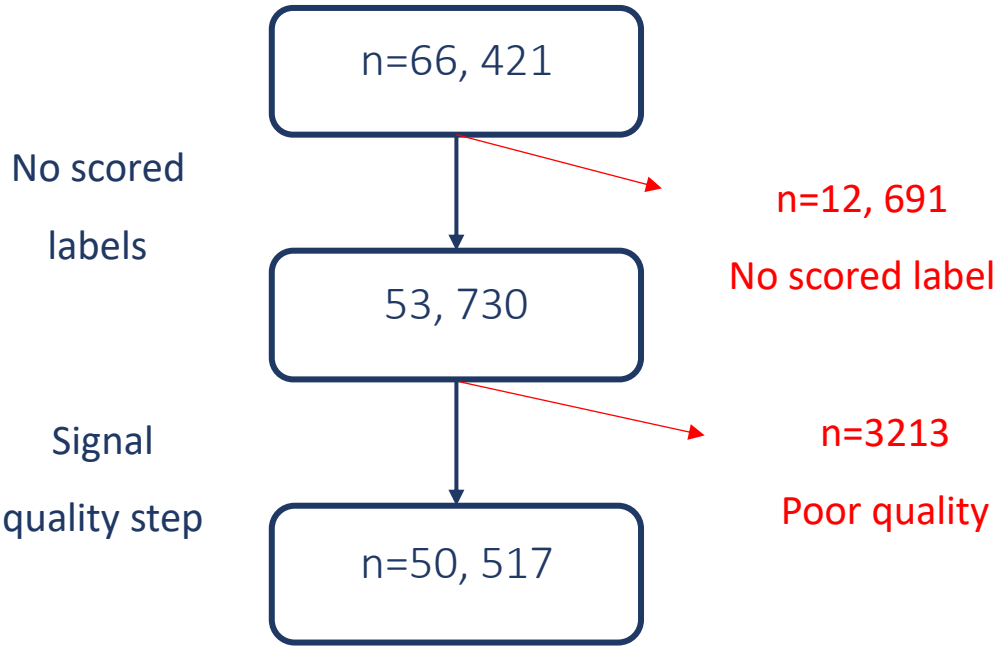
As we have seen previously in the scoring metrics, the output of our classifier on a recording with only unscored labels was not taken into account. However, the output of our classifier on a recording with scored labels was important. For instance, if we classified a scored recording with an unscored label, we would lose points. Therefore, it was not in our interest, in regards to the scoring metrics, to include the unscored labels in the training phase. Therefore, we discarded recordings that only had unscored labels and for recordings that had both scored and unscored labels, we kept the recording while removing the unscored labels.

D. Dealing with noisy signals

Even after our filtering step, some signals remained with a very low signal quality, meaning that they were very noisy. Therefore, we decided to discard the examples with too much noise (signal quality < 0.8).

E. Dealing with manual error

From the way we get the 12-leads electrocardiogram, we realize that there are only 8 initial recordings and that we linearly construct the 12-leads electrocardiogram thanks to these 8 leads. Thereofre, we could perform a **sanity check** on ecg recording by assessing that these linear relations were verified (otherwies, it would mean that the electrodes were switched when recording the ecgs). For such signals, we inverted the polarity of the signals.



Summary of the Different preprocessing steps and exclusion of signals (post Data Augmentation, see below)

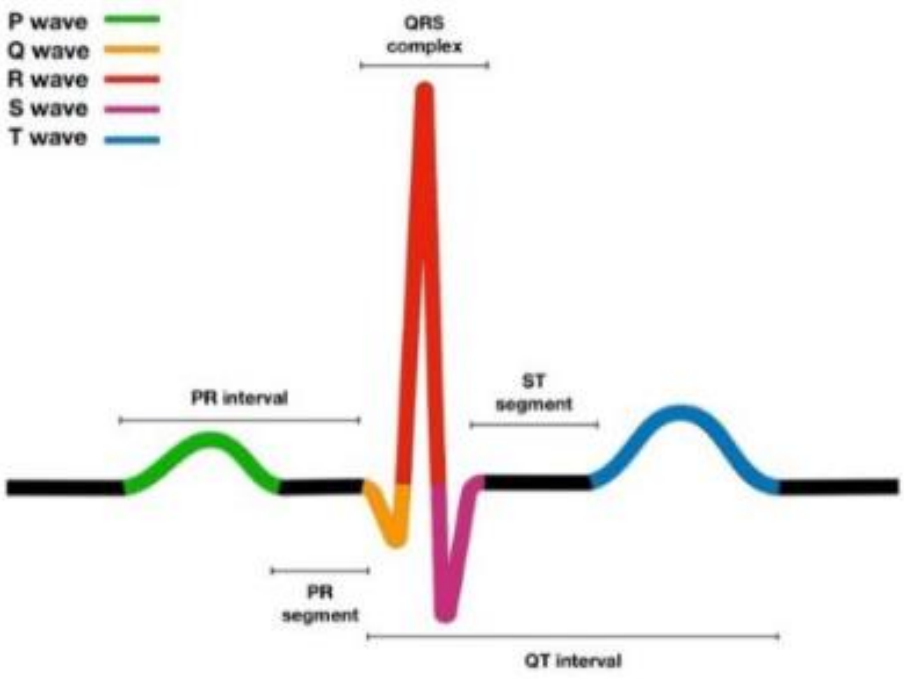
V. The Feature Extraction Approach

The first thing complicated with Feature Extraction is this question: **which features will we extract ? How will they help us to classify between these 27 pathologies ?**

Well, what I did is that I replicated the cardiologist approach. I read through books of cardiology and met with several cardiologists in order to ask them at which part of an ecg will they specifically pay attention to in order to detect a specific pathology. Then, I tried to specifically extract these features.

1. Wave delineation

The first part of the feature detection was waves delineation. Indeed, in every ecg, there is a common structure called the QRS complex. We wish to extract it, alongside with other characteristics, like the P-wave, T-wave and others.



Representation of the different morphological structures in alead of an ecg

In order to do so, I used an open-source algorithm called the **wavedet algorithm**. One problem of this algorithm is that it is only available in MATLAB. In order to use it, I used MATLAB extensions for python and coded several wrappers that allowed me to call MATLAB functions inside the Python code. This was hardly compatible with a Docker environment (we could not download the necessary extensions for MATLAB in Docker out of license problems) and we had to compile these MATLAB files into python packages in order to be able to submit this code to the competition. This was a tremendous work to pull off.

After having extracted the interesting waves, intervals and fiducial points on the ecg, I was able to extract relevant features for every pathology. Here, I will present a toy example in order to show the work I performed for **one pathology (there was 27 pathologies).**

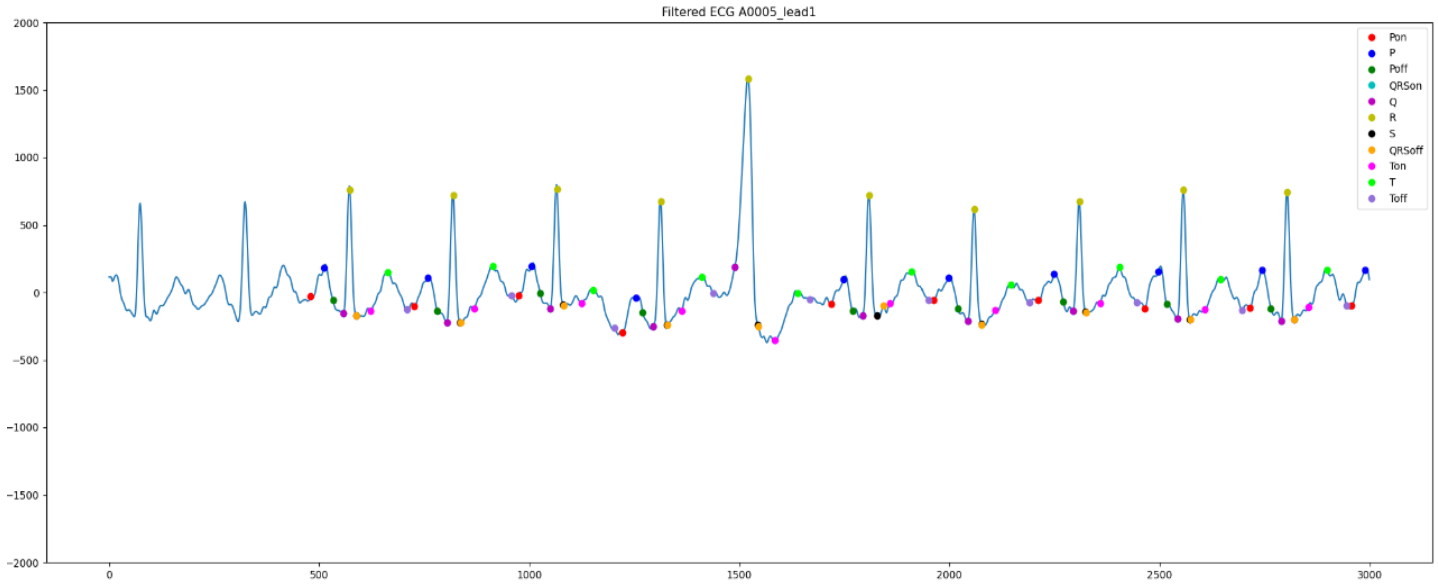


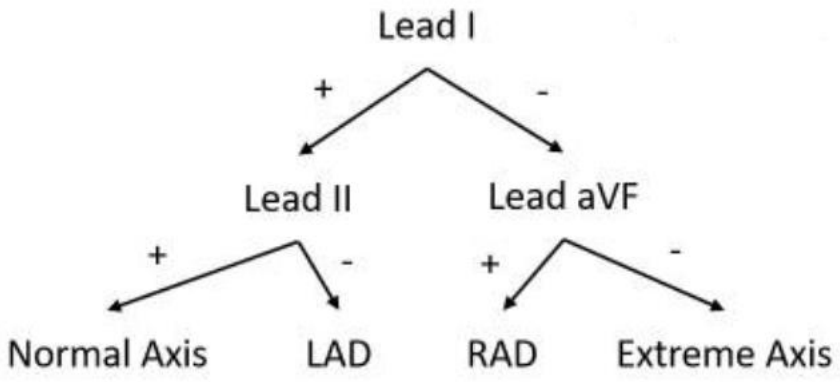
Illustration of the output of our fiducial point detector

2. Feature Extraction, Toy Example: RAD.

In electrocardiography, left axis deviation (LAD) is a condition wherein the mean electrical axis of ventricular contraction of the heart lies in a frontal plane direction between −30° and −90°. This is reflected by a QRS complex positive in lead I and negative in leads aVF and II. .

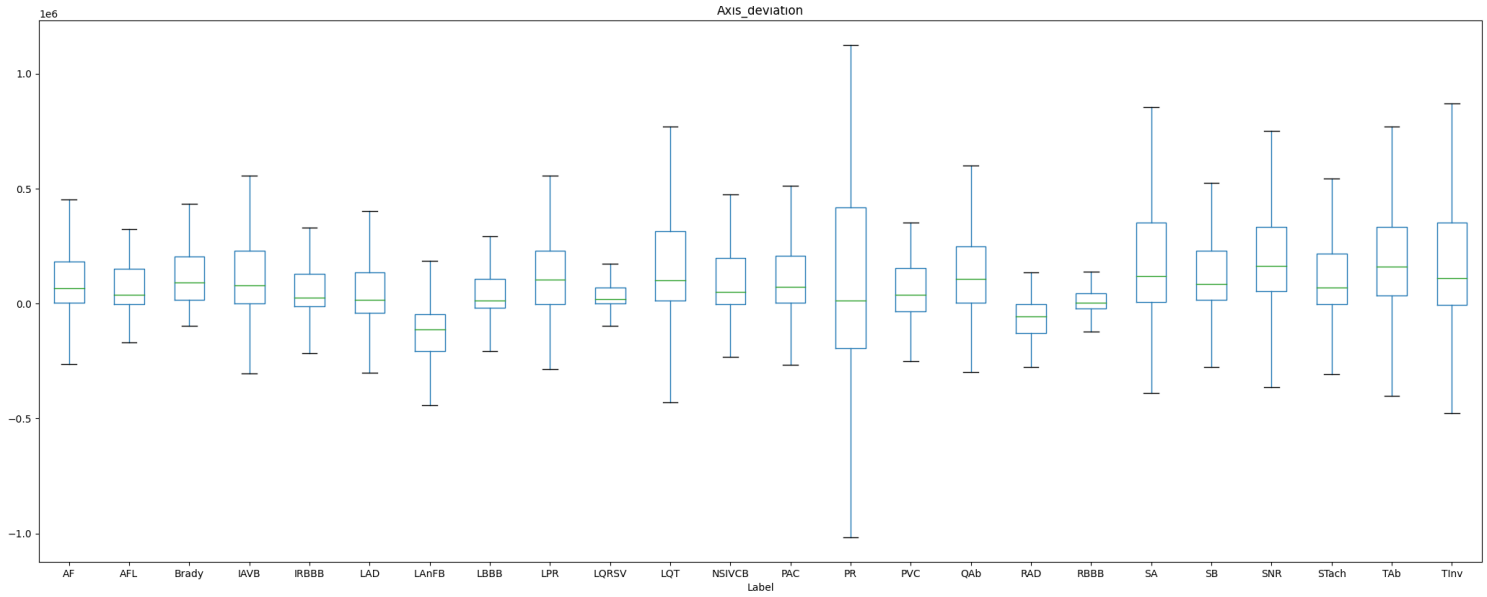
The condition of LAD is usually defined by a QRS electrical axis and an age (the sane QRS electrical axis varies with the patient's age).

Method in order to determine the nature of the QRS axis:



We are going to extract these features for the classification of LAD and RAD. If our classification method is robust, we should perform relatively well because this decision tree structure particularly fits our Random Forest Classifier. The features I will extract for my Classifier are the value of net QRS deflection on leads I, II and aVF and their signs: 6 new features.

Here is the boxplot of: (net_QRS_deflection_lead_I)* (net_QRS_deflection_lead_X) where X = II if the first sign is positive and X = aVF else. Therefore, we should have a negative sign for this feature for the conditions LAD and RAD, with different causes for LAD and RAD.



Boxplot of a discriminative feature for Axis Deviation (RAD)

We can clearly see that this new feature helps to differentiate LAD, LAnFB and RAD from the other pathologies (LAnFB and LAD are the two lowest boxplots): it will help our classification. The values of the LAnFB are differentiated in the boxplot because an ECG characteristics of LAnFB is LAD.

Here is a summary table of the number of features I have extracted for each features. You will find in the Appendix a more detailed approach for several pathologies.

Pathologies	Number of Examples	Number of Features extracted specifically for this pathology per lead
AF	2345	16
AFL	308	16
Brady	259	1
IAVB	1318	3
IRBBB	1221	0
LAnFB	1254	0
LAD	2126	2
LBBB	982	20
LPR	338	6
LQRSV	526	5
LQT	1090	5
NSIVCB	897	0
PR	299	0
PAC	1337	12
PVC	552	26
QAb	824	5
RAD	403	2
RBBB	2018	20
SA	1087	0
SB	1606	0
SNR	12019	0
STach	1555	0
TAb	1865	5
Tinv	832	5

strate

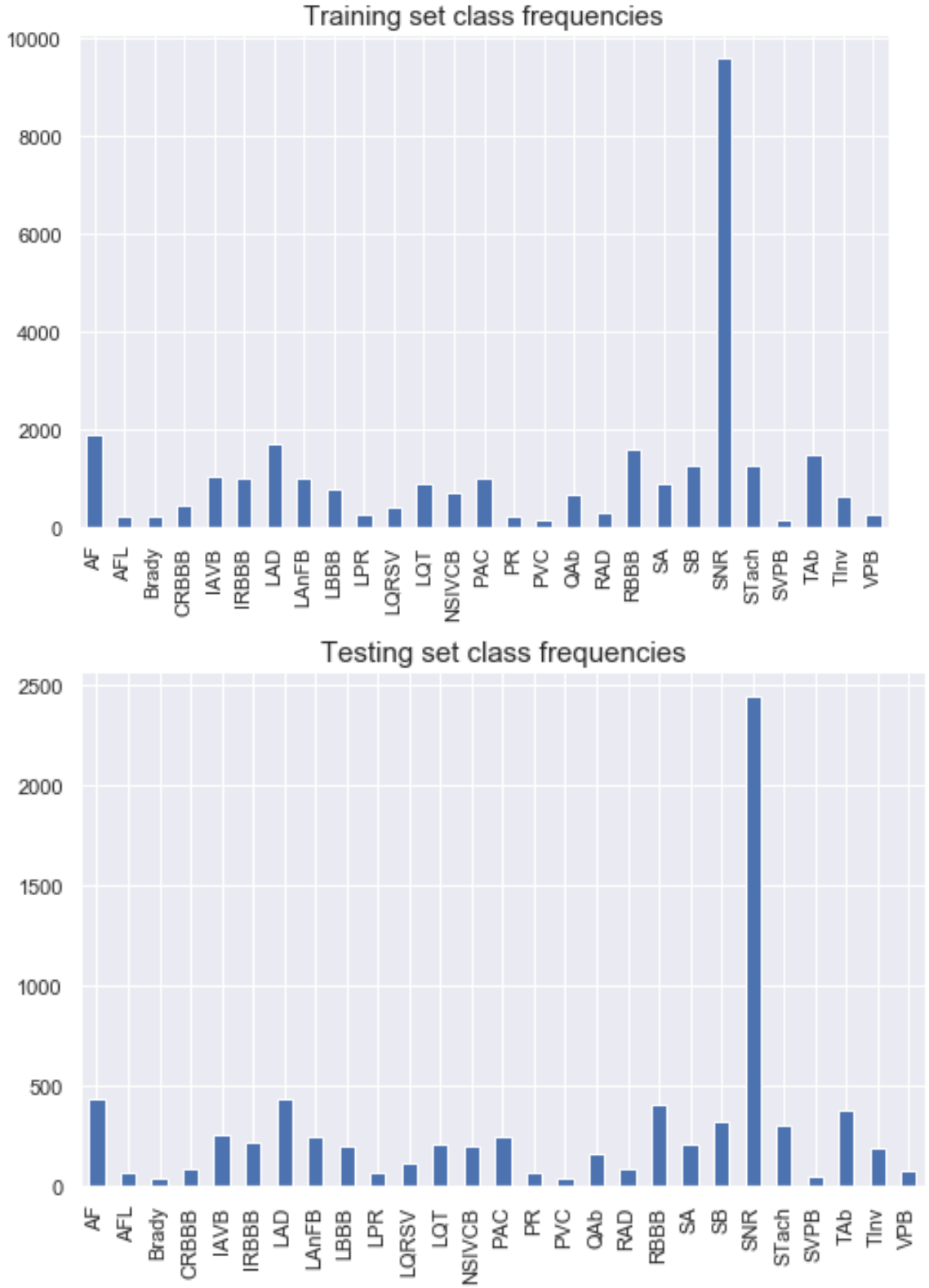
Table of features extracted per lead for every pathology

3. Machine Learning Strategy

A. Training/Testing Split

The first part of our work, and in order to train our Future Ensemble Learning method, was to devise a first training/testing split that would allow us to train + validate our model on the training/validation set and then testi it on the testing set. A challenge of our split was to maintain a sufficient number of examples for every class on both the training and the testing sets.

The result of our split is shown here:



Machine Strategy used: Multi-Label Classification **:** Classification task labelling each sample with x labels with $1 \leq x \leq n_{classes}$. Comparable to running $n_{classes}$ binary classification tasks with sklearn.multioutput.MultiOutputClassifier. However, this approach treats each label independently whereas multilabel classifiers *may* treat the multiple classes simultaneously, accounting for correlated behavior among them (**may learn the prior in the way pathologies can appear simultaneously**): we will try both type of Classifiers.

I used a **One-Vs-The-Rest** approach in order to train our models.

Classifier used: I used a Gradient Boosting in a OneVsRest approach.

The hyperparameters tuned for my model are:

Hyperparameter	**Explanation**	**Value Range**	**Cardinality**
Learning Rate		[0.0001;0.2]	10
Max depth	Depth of One tree	[5:10]	5
Min_child_weight	The minimal weight to split nodes	[10:30]	5
Subsample	Subsample of training data to create a tree	[0.6; 0.9]	5
Col_sample_by_tree	Subsample of features used by tree	[0.6; 0.9]	5
Col_sample_bylevel	Subsamples used for each level of a tree	[0.6; 0.9]	5
Reg_alpha	L2 Regularization term	[1; 5]	5
Reg_lambda	L1 Regularization term	[1; 5]	5
Num_parallel_tree	Number of parallel trees constructed	[200; 800]	10
Scale_pos_weight	Class weighting	#Negative/#Positive	

Data Format: The format of label expected is an array of size $(n_{samples}, n_{classes})$.

I performed Cross Fold Validation on the training Base and evaluate my final model on the Testing Base (the score Cross-Validation used was the **Competition Metrics**, implemented in a generic way for sklearn).

B. Feature Selection

The first results we got were pretty bad. This could be explained by the fact that our classifier tried to predict outcomes with 1360 features. Because of the curse of dimensionnality, we could not leverage efficient features to classify. Therefore, we performed Feature Selection.

Our Feature Selection method leveraged several methods:

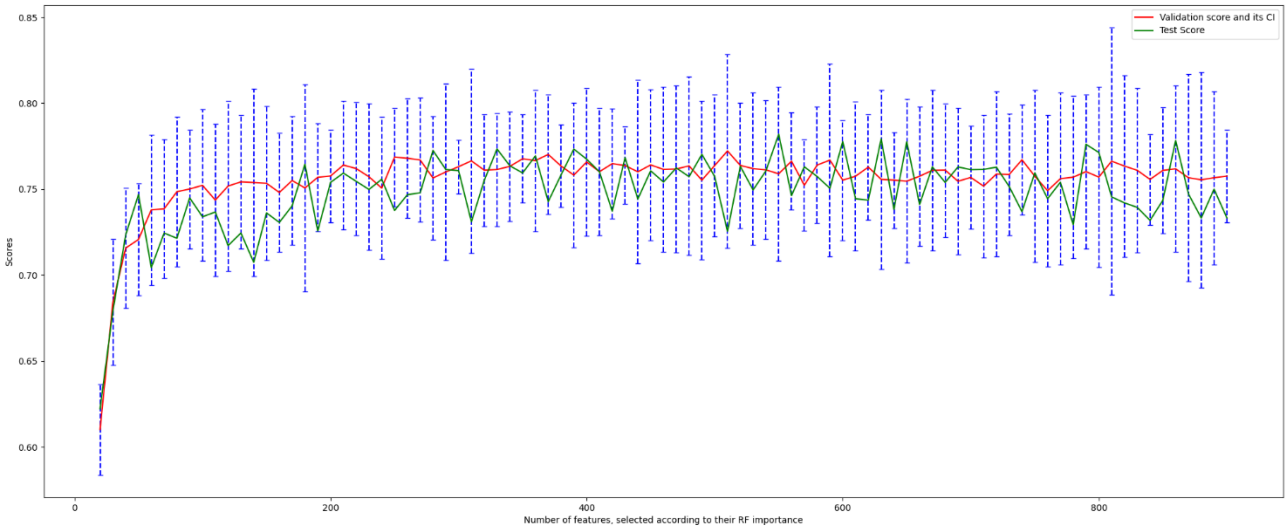
- The filtering methods (independent of the classifier used)
- The wrapper methods (iterative procedures evaluating the classifier at every step)

The advantage of the Filtering methods is that they are way faster than embedded methods (that are very computationally demanding, and time was a precious resource during these 6 months).

In our current model, we are trying to classify patients according to the HrV and morphological features we have extracted from their 12 leads-ECG's. The number of features we have extracted is consequent: 1360 features (around 110 different features we have extracted on every one of the 12 leads).

Therefore, we need to implement some feature selection methods in order to increase the score of our model and prevent overfitting. I have

tried to visualize the overfitting of my global model. In order to do so, I have trained several RF models while increasing the number of features of these models. I have inserted the features in this model by sorting them in decreasing importance according to the RF classifier feature importance. I have then evaluated these models according to their Fbeta performances on the validation set (I have performed 6 fold Cross-Validation with a Random Search) and on the test. The results we got are



Therefore, as we can see, the performances of our model do not improve when we increase the number of features. And we can confidently assess that when increasing the number of features of our model, its performances will drop (we can start to see a decreasing trend on the test set scores at the end of the graph). Therefore, we need to perform Feature Selection. I have implemented

several Feature Selection methods, spanning from heuristics to rigorous approaches, in an increasing complexity fashion.

Prior selection: Filter methods

The filter methods select subsets of features as a pre-processing step, independently of the model. In the majority of filter methods, the feature are ranked according to a statistical performance (correlation with the objective, p-value) and then only the most powerful features are selected. We will explore several filter methods and try to conceive a pre-filtering method allowing to have the greatest score while removing the more features.

Univariate filter methods

First, we will use some methods that study features individually, whether it is statistical characteristics of these features, or information shared with the target pathologies. Nervertheless, we do not compare groups of features, only individual features.

Removing features with low variance:

If a feature has a low variance across all pathologies, this will mean that it does not contribute much to the classification (take for instance a constant feature: useless, and even a burden to our model). Therefore, we will perform a pre-filtering of our data regarding to features variance, and remove the features with the variance lower than a given threshold. The threshold we selected are multiple of the median of the feature variances.

Removing features with low inter-statistical significance between different classes:

Anova Test: The Anova test allows to evaluate the statistical difference between the means of the features' distribution across the different classes. The statistical tests performed are Fisher tests for hypothesis validation. Once again, we need to define a threshold in order to retain the features with the highest scores.

Removing features with a low linear correlation with the target variable:

In this method, we will compute the Pearson Correlation coefficient between every non-categorical feature and the target objective labels. We will then define thresholds in order to discriminate continuous features between the ones highly correlated with our objective and the one not contributing to the evolution of our target.

Removing features with a low non linear correlation with the target variable:

We will compute Spearsman's rank correlation coefficient between features and the target label, measuring how well the relationships between these variables can be described using a monotonic function. Therefore, this measure is stronger than the Pearson Coefficient. By the way, this coefficient is measured by computing the Pearson coefficient between the rank of both variables. In this method, we need to pay attention to the p-values output.

Removing features with low ordinal association with the target :

In statistics, the Kendall rank correlation coefficient, commonly referred to as Kendall's tau coefficient, is a statistic used to measure the ordinal association between two measured quantities. A tau test is a non-parametric hypothesis test for statistical dependence based on the tau coefficient. In the same fashion as above, we will perform feature selection according to the p-value of the statistical significance of the test of ordinal association between the features and the target.

Removing features sharing low information with our target label:

Estimating information between features is more robust than correlation considerations (furthermore, since we are not in the linear context, the advantages of computing the Person coefficient disappear). We define the mutual information between two random variables by:

\begin{equation}

I(X,Y) = \int \int {dxdy \quad \mu(x,y)log\frac{\mu(x,y)}{\mu_{x}(x)\mu_{y}(y)}}

\end{equation}

In this implementation of Mutual Information coefficient, we no longer compute this value by approximating it via a frequency approach, but we rather use a k-Nearest Neighbors in order to approximate it. We therefore insert some additional parameter: the number of neighbours we wish to use in order to compute our approximation. As the intuition dictates, there is a bias/variance trade-off when we choose this parameter. We will see the results of this method using neighbors=3 and neighbors=5.

Please see the Appendix for an example of a table of results for the implementation of anunivariate filter feature selection method.

ALL in One Univariate Filter:

Combining all the strengths from the previous univariate filters, I have devised a final Univariate Filter that would allow me to select the best subset

I performed 10-fold cross validation for this step in order to select the very best pre-filtering method (in terms of the least number of selected features, and best scores on validation set and test set).

	#Features	Validation Score	Test Score
Variance: 0.01, Anova: 60, Mutual Info: 0.06, Spearsman: 0.025, Kendall: 0.1	485	0.83	0.78

After having selected the best subset of features according to our univariate filter, I further leveraged Multivariate Filter Methods, in order to account for multi-correlation.

<u>Multivariate Filter methods</u>

I implemented some multivariate filtering methods. These methods take into account dependencies between features.

Minimum Redundancy Maximal Relevance Filtering Process

In this method, we try to find the subset S solving this problem \cite{mRMR}

\begin{equation}

\max_{S} {\frac{1}{|S|}\sum_{f_i \in S}I(f_i,y) - \frac{1}{|S|^2}\sum_{f_i, f_j \in S}I(f_i,f_j)}

\end{equation}

Correlation Feature Selection

The correlation feature selection (CFS) measure evaluates subsets of features on the basis of the following hypothesis: "Good feature subsets contain features highly correlated with the

classification, yet uncorrelated to each other".\cite{CFS} The following equation gives the merit of a feature subset S consisting of k features:

$$\textit{Merit}_{S_k} = \frac{kr_{cf}}{\sqrt{k + (k-1)r_{ff}}}$$

Here, r_{cf} is the average value of all feature-classification correlations, and r_{ff} is the average value of all feature-feature correlations. The CFS criterion is defined as follows:

$$CFS = \max_{S_k}\frac{r_{cf_1}+...+r_{cf_k}}{\sqrt{k + 2(r_{f_1f_2} + ... + r_{f_if_j} + ... + r_{f_{k-1}f_k})}}$$

Unfortunately, implementing Multivariate Feature Selection (whether it was on top of Univariate Filtering method or not) was not conclusive and did not allow us to efficiently select a subset of features.

Therefore, since we had selected still 465 features, which is still too much (in terms of performances of in terms of lengths of feature extraction for experiments), we needed to perform Wrapper methods.

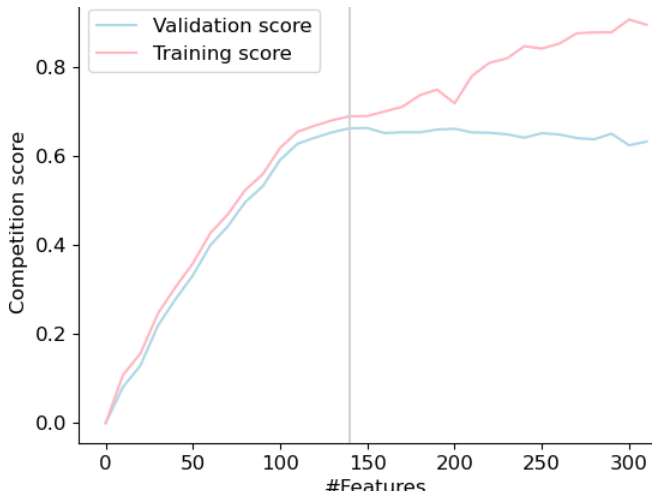
Wrapper: Incremental Feature Selection

I implemented some wrapper methods. These methods are genetic algorithms leveraging heuristic approaches in order to select the best subset of features. These methods are different from the filtering approach since they depend directly on the performances of a selected classifier (in our case, we will use a Random Forest classifier), and on the metrics we select to score a model vs another (in our case, it was the **competition metrics, not the Fbeta anymore**).

Forward floating approach

At every step, when we add a feature, we remove every time the worse feature, meaning that if we can improve the performances of our model by removing a feature, we do it.

The results we got when performing Forward floating feature selection were:



Visualization of the final Floating Forward Approach

Therefore, after the Feature Selection step, we reduced our dataset to a dataset with only 136 features, which is way better.

Before classifying our features, we used a data standardization in order to stabilize the Optimization process:

- Data Standardization for numerical features
- Categorical encoder for categorical features

Pathologies	Number of Examples In the Competition DataBase	Number of Features extracted specifically for this pathology per lead	Fbeta score Multi-Label Classification	Fbeta score Single-Label Classification	Opportunities for Data Augmentation
AF	2345	16	0.91	0.84	
AFL	308	16 (the ones from AF)	0.20	0.25	
Brady	259	1	0.20	0.26	
IAVB	1318	3	0.72	0.61	
IRBBB	1221	0	0.14	0.02	
LAnFB	1254	0	0.60	0.42	
LAD	2126	2	0.72	0.57	
LBBB	982	20	0.80	0.85	
LPR	338	6 (3 AVB, 3 challenge)	0.17	0.11	
LQRSV	526	5	0.03	0.15	
LQT	1090	5 (challenge)	0.75	0.33	
NSIVCB	897	0	0.60	0.04	
PR	299	0	0.03	0.51	
PAC	1337	12	0.0	0.76	
PVC	552	26	0.42	0.18	
QAb	824	5	0.68	0.01	
RAD	403	2	0.79	0.22	
RBBB	2018	20 (the ones from LBBB)	0.80	0.70	
SA	1087	0	0.44	0.50	
SB	1606	0	0.80	0.64	
SNR	12019	0	0.80	0.80	
STach	1555	0	0.90	0.82	
TAb	1865	5	0.46	0.14	
Tinv	832	5	0.80	0.03	

Color code for the Table:

Pathologies for which the score is satisfying: AF, LBBB, RBBB, SBR, SB, STach

Pathologies that have consequently benefitted from the Multi-Label Classification: IAVB, LAnFB, LAD, LQT, NSIVCB, PVC, SB, TAb, Stach, RBBB

Pathologies that have consequently suffered from the Multi-Label Classification: LQRSV, PR, PAC

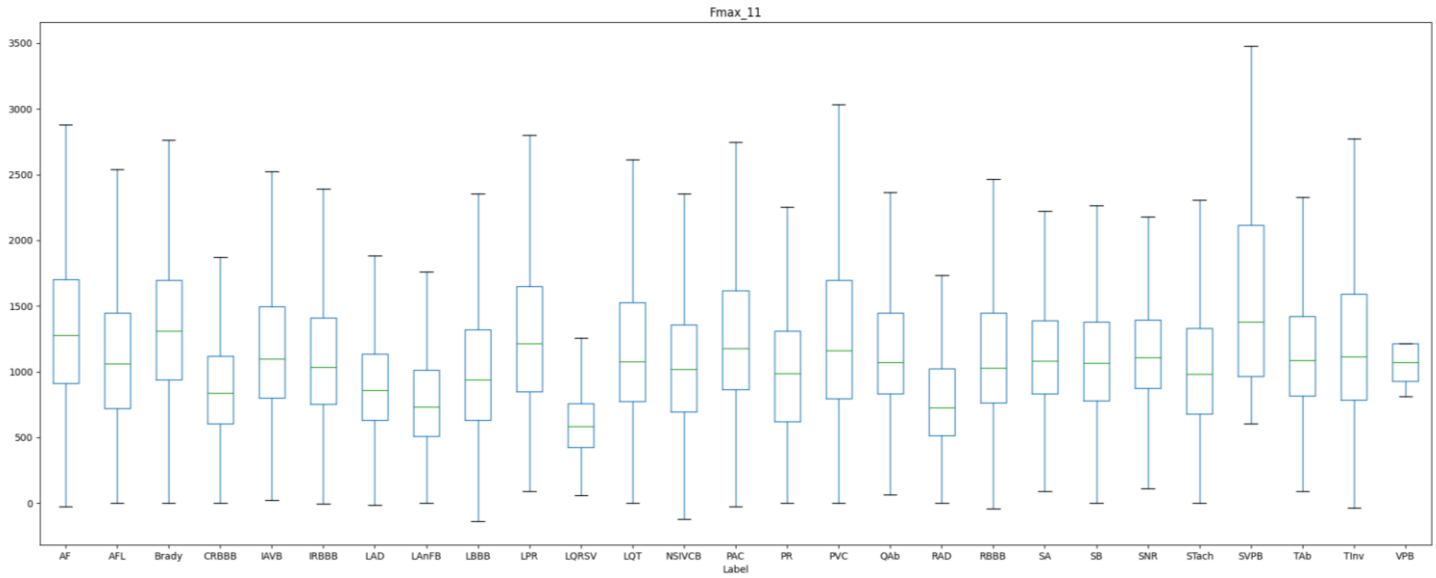
Pathologies fo which Data-Augmentation is a valuable option: AFL, Brady, LPR, LQRSV, PR, PVC, RAD, TInv (pathologies for which I will first extract new features before Augmenting Data)

Pathologies for which I will extract new features: AFL (in order to separate with AF), Brady, IRBBB, LAnFB, LAD, NSIVCB, PR, QAb, RAD, TAb, SA, TInv

C: Motivation for Data Augmentation

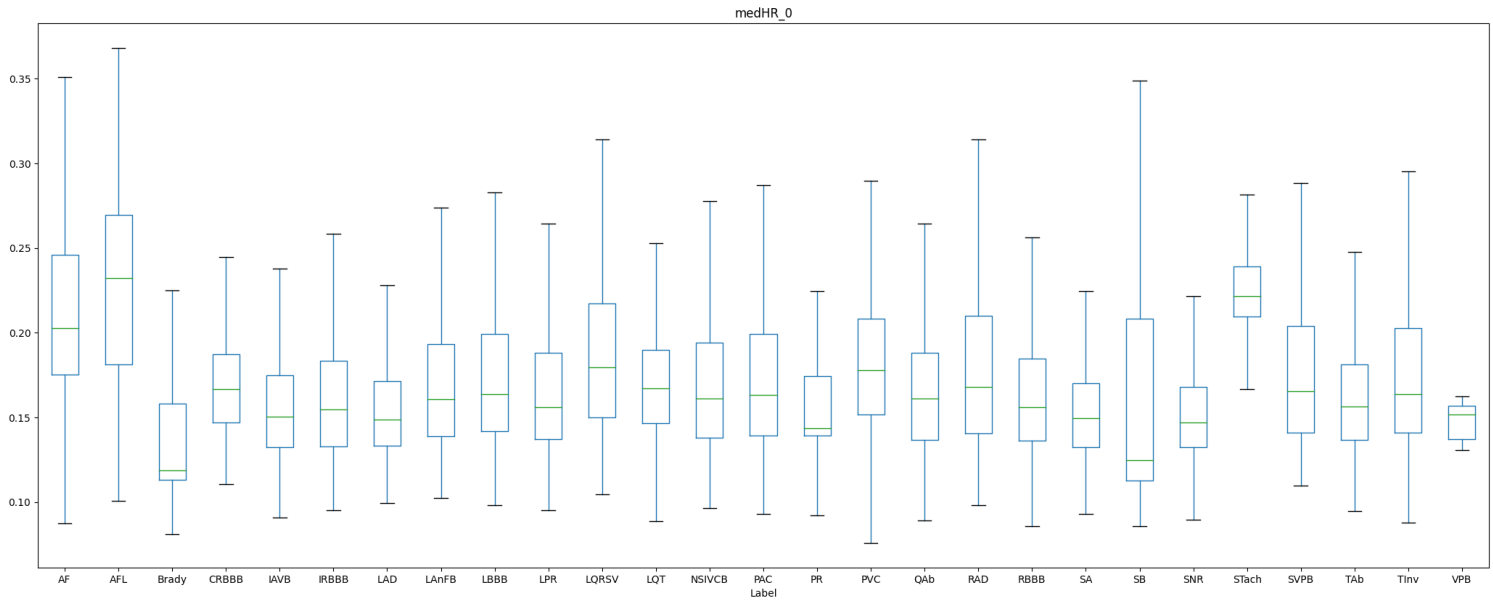
Pathologies for which we extract relevant features but don't have enough examples:

1. LQRSV [errors: Misclassification with SNR]: The low qrs voltages can be detected by extracting the amplitudes of R peaks in the ecg.



We can clearly see that this feature enables us to differentiate LQRSV examples from other pathologies: we need to do data augmentation in order to prevent the misclassification with SNR examples.

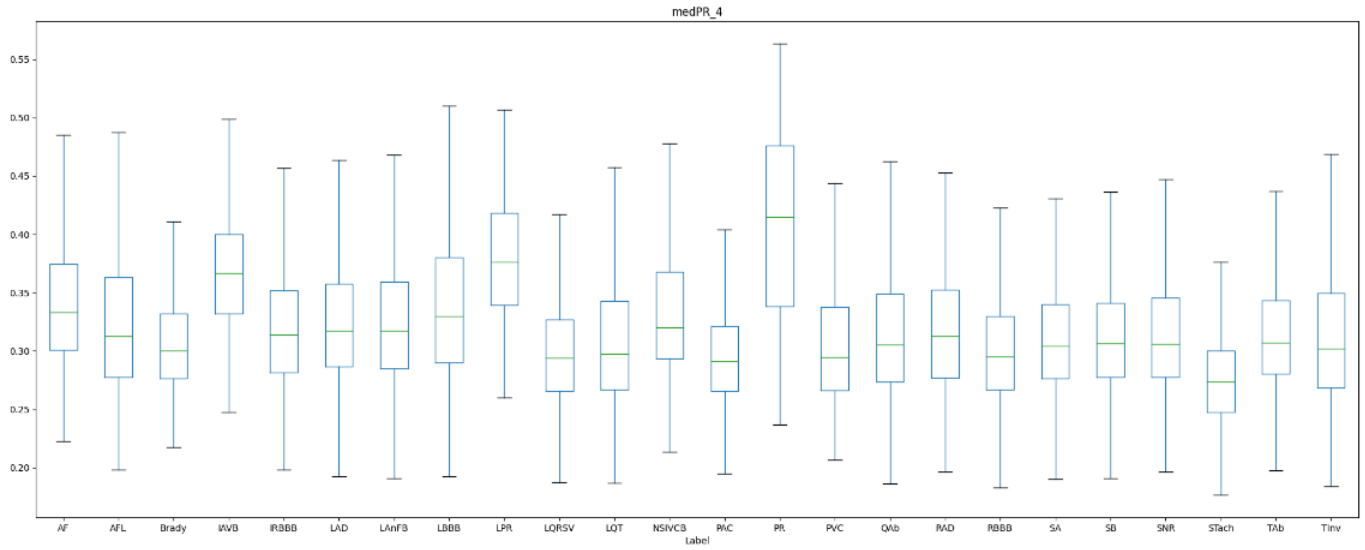
2. Brady [errors: Misclassification with SNR]: The first thing we need to take into account with Bradychardia is that it only co-occurs with PAC, RBBB, IRBBB, PVC. Bradycardia is a condition where an individual has a resting heart rate of less than 60 beats per minute (BPM) in adults. Therefore, the median Heart Rhythm should be a good indicator of Bradychardia.



As we can see from the above boxplot, the median Heart rhythm allows to efficiently separate Bradychardia from the other pathologies. The main source of error from Brady comes from misclassification with SNR, though this feature should help to separate between both. Therefore, we

misclassify as SNR out of bias towards the most represented class. This is why the next step for Brady is Data Augmentation.

3. LPR [error: misclassification with SNR]:



Just with this feature, our model should be able to differentiate between LPR examples and SNR examples: we need to augment the data we have.

Therefore, we looked for open-access (or restricted) databases in order to augment our training database, and we looked especially for databaseswhere the underrepresented classes would be present.

D: Data Augmentation

4 Databases:

Paper: *A 12-lead electrocardiogram database for arrhythmia research covering more than 10,000 patients, JianweiZheng, JianmingZhang, Sidy Danioko, HaiYao, HangyuanGuo Cyril Rakovski*

Raw description:

Provenance	**Number of Examples**	**Mean length of an example**	**Previous filtering (code availability)**	**Multi-Label**	**Sample Rate**
China	**10646**	**10s**	**Yes**	**Yes**	**500**

DataBase composition:

SB	3836
TAb	1869
SNR	1826
AF	1754
STach	1532

RBBB	437
AFL	441
LAD	380
PVC	306
PAC	277
IAVB	246
QAb	233
RAD	221
TInv	157
LBBB	93
LQT	57
LPR	13
LQRSV	3

Pathologies Co-occurences

Sanity check: visualization of the pathologies co-occurences of the first augmenting database

Paper: *A 12-Lead ECG database to identify origins of idiopathic ventricular arrhythmia containing 334 patients JianweiZheng, Guohua Fu, KyleAnderson, HuiminChu Cyril Rakovski*

Raw description:

Provenance	**Number of Examples**	**Mean length of an example**	**Previous filtering (code availability if yes)**	**Multi-Label**	**Sample Rate**
China	**334**		**Yes/Yes**	**No**	**2000**

DataBase composition:

PVC	325
VT (Ventricular tachycardia) = 'other'	9

Paper: *Lobachevsky University Electrocardiography Database*

Raw description:

Provenance	**Number of Examples**	**Mean length of an example**	**Previous filtering (code availability if yes)**	**Multi-Label**	**Sample Rate**
Russia	**243**	**10s**	**No**	**No**	**500**

DataBase composition:

SNR	143
STach	4
SB	25
LAD	66
RAD	3
IAVB	10
IRBBB	29
RBBB	4
LBBB	4
NSIVCB	4

Paper: *Automatic diagnosis of the 12-lead ECG using a deep neural network*

Raw description:

Provenance	**Number of Examples**	**Mean length of an example**	**Previous filtering (code availability if yes)**	**Multi-Label**	**Sample Rate**
Brazil	**200**	**10s**	**No**	**No**	**300-600Hz**

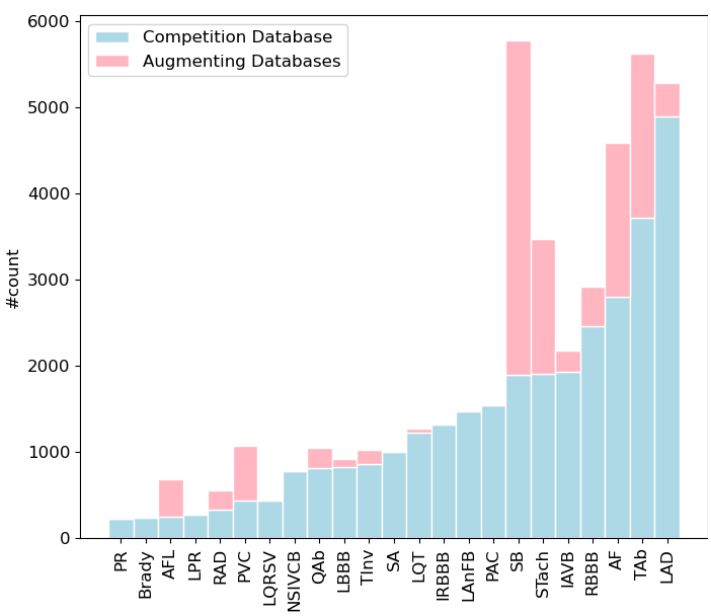
DataBase composition:

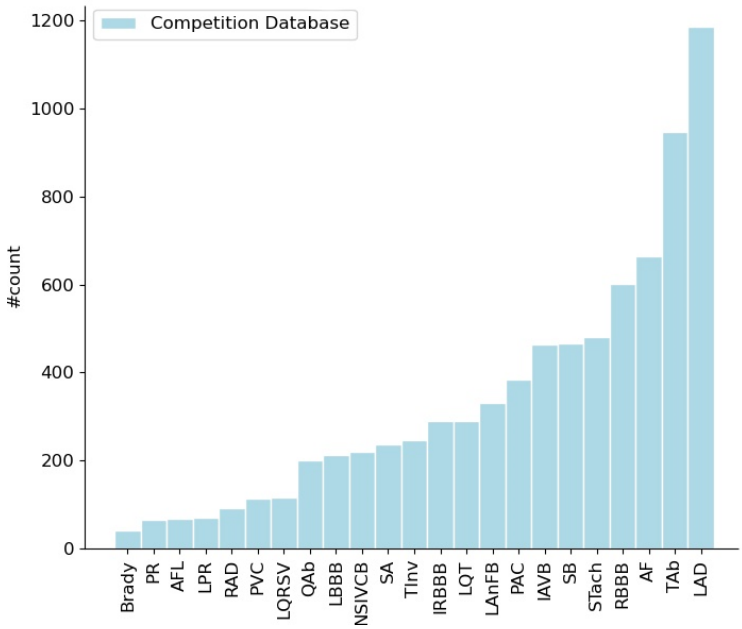
IAVB	28
RBBB	34
LBBB	30
SB	16
AF	13

Summary Data Augmentation

SB	3861
TAb	1869
SNR	1969
AF	1767
STach	1536
RBBB	475
AFL	441
LAD	446
PVC	631
PAC	277
IAVB	284
QAb	233
RAD	224
TInv	157
LBBB	123

LQT	57
LPR	13
LQRSV	3
IRBBB	29
NSIVCB	4





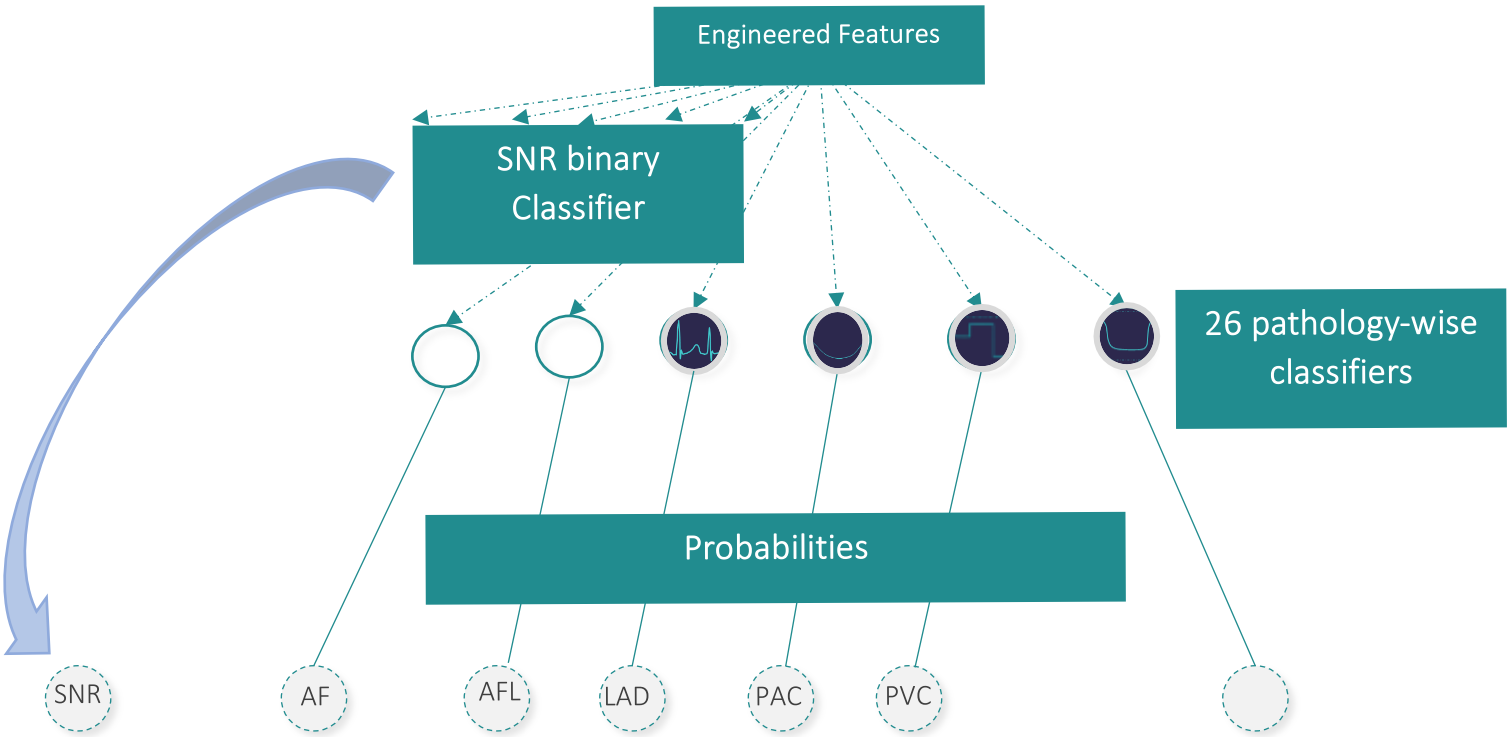
Note: in the vizualisations above, we represented the training and testing sets without the SNR labels for the sake of visualization.

Database	**Included**	**Excluded**	**fs (Hz)**
CPSC	7007	3673	500
Georgia	9458	886	500
PTB	97	419	1000
PTB-XL	21604	233	500
INCART	5940	7380	257
CUSPH (add)	9749	897	500
CUNFH (add)	325	9	2000
TOTAL	53730	12691	

Table 1: Examples that were included and excluded for each database based on the presence of at least on label within the 27 classes that were considered by the Challenge performance measure.

E: Model Architecture

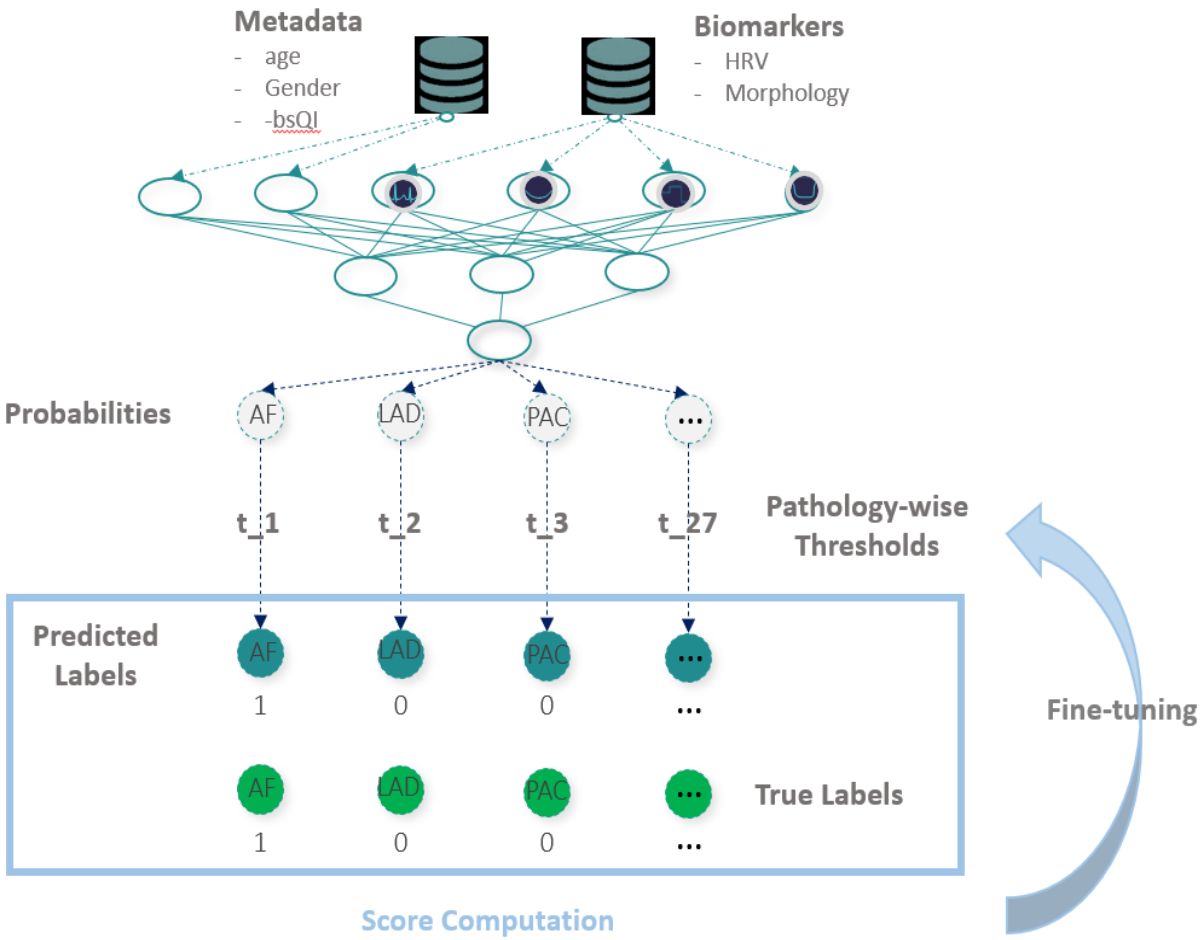
In order to prevent the bias towards the most represented class (SNR), I added a classifier on top of the pathology classifier that would classify for SNR vs non-SNR. The architecture is displayed here. (Note that some examples are multi-labelled, and that an example may be SNR + other pathologies), explaining why this task is complicated.



Visualization of the architecture of our final Feature Extraction model

Last, since our scoring metrics was computed between the predicted labels and the true labels, we needed to convert probabilities into labels. This has been done using some thresholds. We further optimized these thresholds on the validation set, according to the competition metrics. We defined some thresholds pathology-wise.

F: Threshold fine-tuning



Visualization of the fine-tuning of pathology-wise thresholds

Final Competition Score of our Architecture:

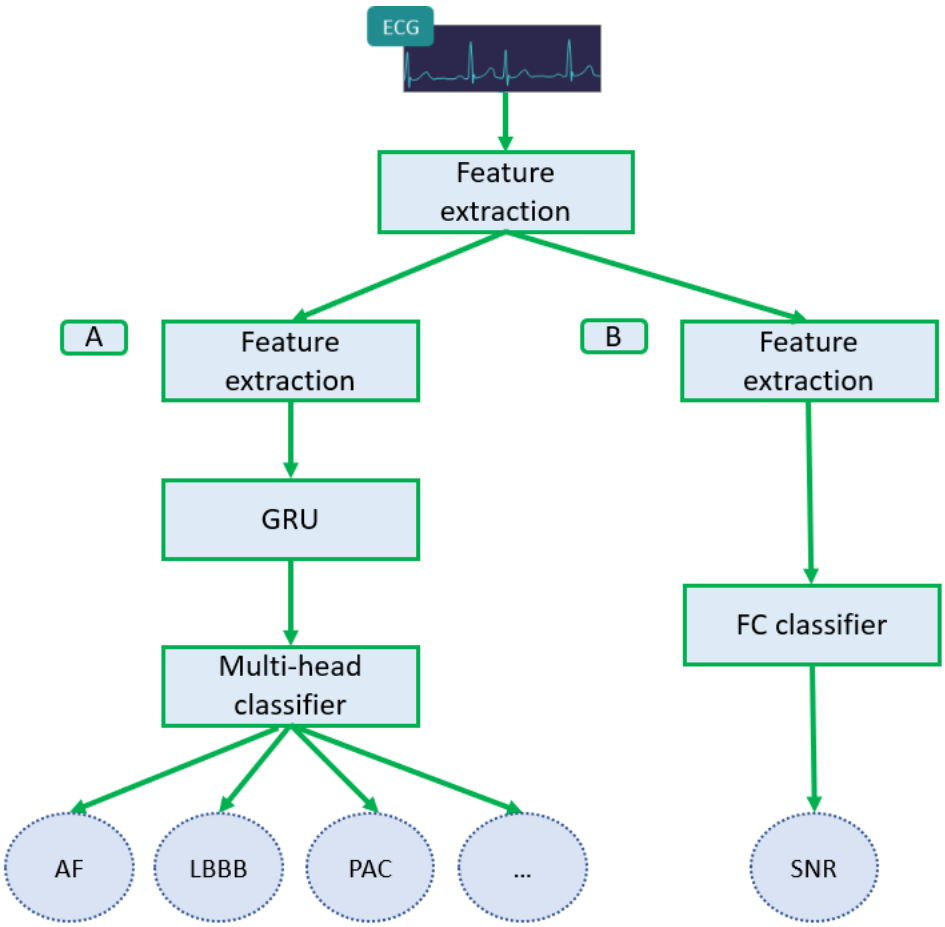
Validation score	Test score
0.66+/-0.02	0.67

VI. The Deep Learning Approach

Within the model, two deep learning (DL) networks were trained. The first network is a binary classifier for SNR against the rest. The second network is used to classify for any of the 23 cardiac abnormalities using a multi-head model. Padding was performed, to the constant length of 40000 samples (i.e. 80 seconds at fs=500Hz).

This way, the huge amount of SNR labels doesn't affect the parameters in the path A. This allow the model to not perform overfitting to the SNR class. The high-level of the architecture can be seen in

.



High-level architecture of Deep Learning approach.

The feature extraction part of both networks is a Convolutional Neural Network (CNN) model. Each CNN cell was composed of: Conv1d, BatchNorm, ReLu, Conv1d, BatchNorm, ReLu, MaxPool, Dropout. The network was trained with shortcut connections to avoid vanishing gradients. A total of 8 CNN cells were used, with shortcuts every 2 cells, followed by a Gated Recurrent Unit (GRU) cell. The GRU takes as input a total of 1250 features produced by the CNN. A multi-head model was used, which means a fully connected layer for each of the 23 cardiac abnormality. Stratified 5-fold cross validation validation was performed to find the best hyperparameters of the model. A combination of two losses was used: BCEWithLogitsLoss, and the F-Beta score. The weight of each loss was considered a hyperparameter of the model. The Adam optimizer was used.

In the pre-processing stage we've tried to automatically detect the QRS peaks (as well as other important points) in each signal, which are very critical part to understand ECG signal.
After having a clearer signals, we've exported features in two ways:

- Generic features (e.g. highest peak, average R peak, variation, etc.)
- Manual features – we investigate each of the given pathologies and how do we recognize them manually and defined for each pathology a specific features set to help recognizing it. (for example, checking whether is there a peak between R and S points is important for ST elevation pathology detection)

With these features we want to build our first model which used RF as the classifier, and trained to maximize the "f beta" metric – this metric is pretty reasonable for this task as it punishes on false positives and false negatives, and can be controlled with the beta value. (in our competition we got the required beta value).

First, we only used a RF to predict one label per example (so 100% accuracy is not achievable by definition), then we went and enlarged it to a model which classifies multi-labels.

As we tuned our hyper parameters using CV, we found out that we have too much features, so we used several algorithms for feature selection, as explained in the Methods section.

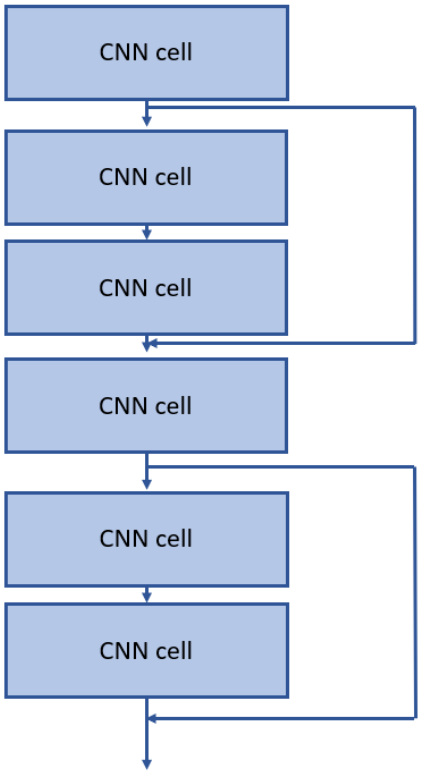
This is our first model, which is working on manual and generic features.

As we know, a CNN should get a constant input shape (there are some ideas how not to, but it seems to hurt the model ability to converge). Thus, we've padded the ECG leads to a constant number of samples – this is one of the hyperparameters of our model. Of course, that "padding" means also to delete samples if the ECG is too long.
We investigated many some padding methods, there are some considerations:

- ECGs are sampled in a specific known frequency, we didn't want to change the frequency while padding, in order to help the model to converge. (if we were changing the frequency, the model also had to understand what is the current frequency)
- If padding before/after the real samples, we don't want the model to falsely get consumptions from the edges.

We chose to pad with zeroes after each ECG and delete samples from the end if it is too long.

One model is working by applying multiple 1D convolution layers, Maxpool and Dropout layers, some of the block have also shortcuts to prevent vanishing gradient (at first we suffered from it, then added the shortcuts).



Architecture of feature extraction part.

After those layers which aims to export features, a GRU unit was applied, which can use their internal state (memory) to process variable length sequences of inputs.

For the classification part, we applied Linear layer with batch-normalization, and activation of LeakyRelu. (we didn't use Relu and Sigmoid since we suffered from models which couldn't be trained – the derivative was too small because at first we arrived the flat areas of the Sigmoid).
Due to some problems with the multi-labels, we separated the model to "multi-head" model which recognizes each pathology alone, and then combining the results to our result vector of pathologies.

This model was trained for 100 epochs, for 32 hours. At each epoch, a checkpoint was saved if the test metric was the highest obtained so far.

Final Competition score of the DL architecture:

Validation score	Test score
0.56+/-0.04	0.58

VII. Ensemble Learning approach

In order to combine the strengths of the FE and the DL approaches an Ensemble Learning (EL) model was evaluated. It consisted in adding a logistic regression unit taking as the input the probability outputs of the FE and DL models.

The Neural Network and the classifiers were trained on the same Train/Test split in order to be able to train the Logistic Regression on the same Train/Test split. The Training Set has been further split into Training/Validation sets in order to tune the hyper parameters of the Logistic Regression layer (set with an elastic net penalty).

Final Competition Score of the Ensemble Learning Approach

Validation score	Test score
0.67+/-0.02	0.67

VIII. Submission of the model and final ranking

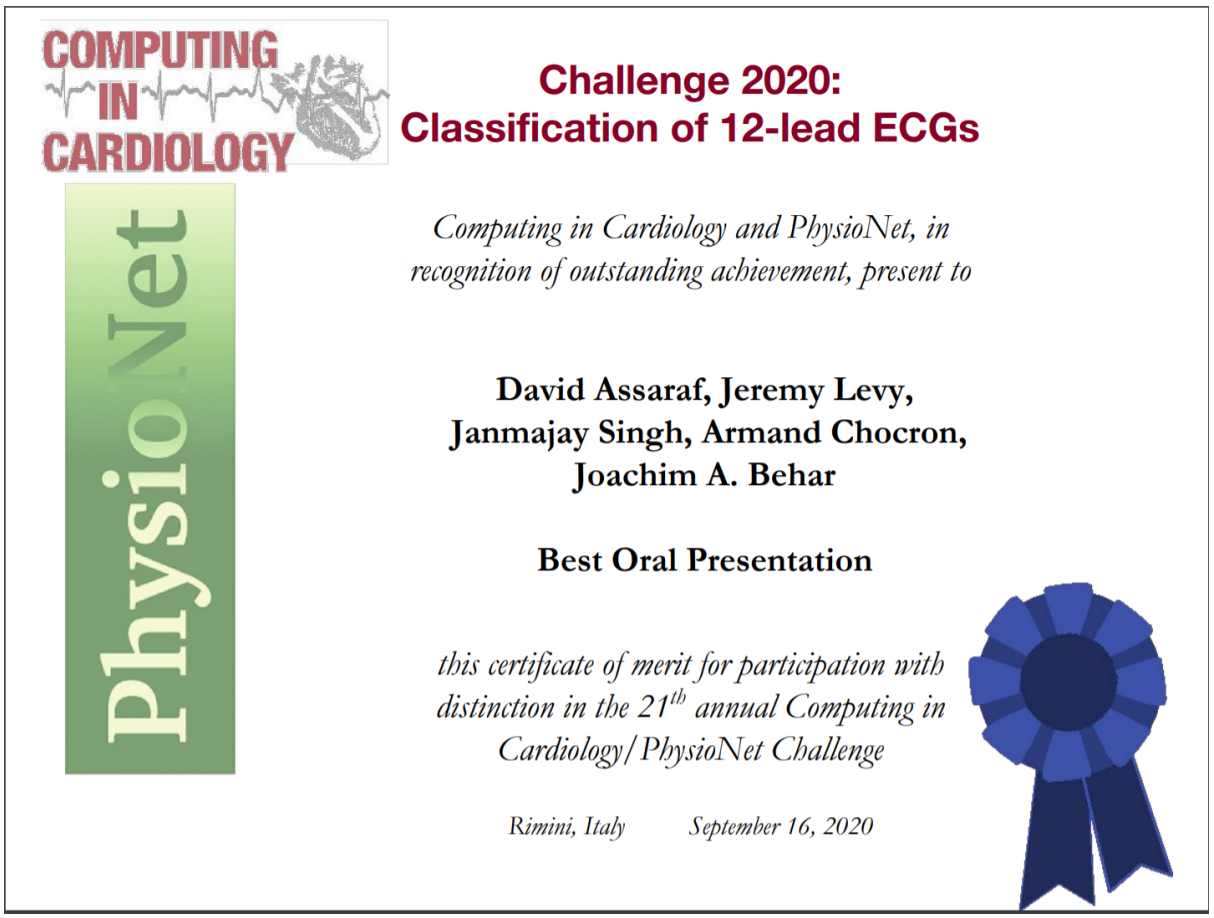
The last (but very not least) step of our work in this competition was to submit our final model in the server of the compeititon. As announced in the introduction, being part of an official competition comes with some hard constraints on the submission format. Indeed, I have used a model that combined MATLAB (for feature extraction and signal processing) and Python (for interpretable Machine Learning). There was no possibility of submitting such a model. Therefore, we tried to build a Docker image that would be compatible with Python and MATLAB. Still not compatible. The last thing we tried was to compile the MATLAB files into python packages. But this was taking a huge amount of efforts because using python libraries did not allow us to use MATLAB for python built-in cast functions. Therefore, I could not submit my model. We submitted the DL model which performance are worse. Neverthelsee, our final ranking in the hidden test set was **41/400.** Out of Cross-Validation scores, we would have been ranked 17/400.

IX. Paper Redaction

After having successfully submitted our model, we have been invited to the CinC conference 2020 in Rimini. I was particularly invited as a speaker for the conference (I have been talking with the organizers of the competition during 5 months and they recommended me as a speaker in order to present the Feature Extraction approach). Being a speaker in the conference required that we submitted a paper, that would later be published in IEEE Physiological Measurement. This was an exciting exercise to do, we were asked to sum up six months of work in no more than 4 pages (including the introduction, the abstract and the conclusion). Prof. Joachim Behar really guided me through these steps and helped me to succintly express my mind. This vas a very instructive experience, and I am glad to have published one paper, to be available as *Classification of 12-lead ECGs Using Digital Biomarkersand Representation Learning: David Assaraf, Jeremy Levy, Janmajay Singh, Armand Chocron and Joachim A. Behar*

X. The conference and awards

I have been invited to speak at the CinC Conference in Rimini. However, due to the Covid, I chose to not go there (I had already started the classes at Harvard). Therefore, I spent three days on Zoom collaborating with researchers from all over the world. My talk has been followed by 400+ researchers. At the end of the competition, **I won the prize of best speaker at the CinC conference!**



XI. Seminar for future research

After the conference and having started Harvard Master of Data Science, I worked during a month with AIM Lab researchers and new PhD in order to explain them how to use my code and whoch were the future avenues of work that I would have done if I had more time. I think that I will continue working with them during the winter break, and publish a second paper (this time, it would be longer, and more exhaustive than the one that would be published in IEEE). Furthermore, we have been contacted by *Focus* in order to publish our work. I will maybe get to work on that this summer should I have some time off.

XII. Impact of work

This work is going to be leveraged at several scales: Joachim is currently using it in order to collect funds in order to sustain further research leveraging ~2M ecgs. The PhDs working on this topic will further improve the performances of the model, by also improving the capacities of the Neural Network we used. Furthermore, since interpretability is crucial in these sectors of application (Medical sector), they will also work on providing interpretable results for the clinician (leveraging attention layers for the Neural Networks). Last, Joachim wants to create a Python library but I do not think that I will be part of this adventure.

XIII. Experience on Leading a research team

I think that these 6 months have been the richest experience in my life. I gained confidence in my capacity to quickly gain knwledge in a very complicated filed (BioMedical Engineering), have been able to lead a team of 4 researchers from all over the world, become proficient and very comfortable with the manipulation of complex tools. I have also been able to interact in an opened way with some professional researchers, that were giving a lot of attention to my work. We managed to successfully meet all the time steps we have scheduled in team, and this was thanks to a clear organization we put in place.

XIV. Conclusion

As a word of conclusion, I would like to thank the AIM Lab for the extensive help they provided me in my research work. Whether it was in terms of infrastructure, human collaboration or specific research guidance, their precious help has been crucial in the development of this work, and of my improvements.

I also wanted to thank all the Ecole Polytechnique professors who gave me the necessary applied mathematics background in order to tackle these difficult concepts (statistical feature selection, statistical signal filtering, …).

I believe that this competition has been veryrewarding: I have learnt more than I had during 5 years of theory. Being let in autonomy and forced to do some research work on my own has given me a lot of confidence in what I am capable of achieving for the years to come. It has also made me think about the ooportunity of pursuing my research work on a PhD after Harvard. Joachim gave me the opportuniy to continue this research work on a PhD with his lab, in partnership with Ecole Polytechnique but I decided that I was not yet ready to take such decision. Maybe after my Master would I have a different perception.

XV. References

[1] Marcel J. E. Golay Abraham Savitzky. Smoothing and differentiation of data by simplified least squaresprocedures. 1964.

[2] Lina Zhao Xiangyu Zhang Feifei Liu, Chengyu Liu. An open access database for evaluating the algorithmsof electrocardiogram rhythm and morphology abnormality detection. 2018.

[3] Patrick S. Hamilton. A comparison of adaptative and nonadaptative filters for reduction of power lineinterference in the ecg. 1996.

[4] S Hargittai. Savitzky-golay least-squares polynomial filters in ecg signal processing. 2005.

[5] Arthur Garson Leo G. Horan James J.Bailey, Alan S. Berson. Recommendations for standardization andspecifications in automated electrocardiography: Bandwith and digital signal processing. 1989.

[6] PABLO LAGUNA LEIF SÖRNMO. Electrocardiogram signal processing. 2006.

[7] Ivanov R Daskalov I Christov I Levkov C, Milhov G. Removal of power-line interference from the ecg: areview of the substraction procedure. 2005.

[8] W.J. TOMPKINS M. L. AHLSTROM. Digital filters for real-time ecg signal processing using micropro-cessors. 1983.

[9] Berson AS Briller SA Brody DA PipBerger HV, Arzbaecher RC. Recommendations for standardizationof leads and of specifications for instruments in electrocardiography and vectorcardiography. 1975.

[10] Ali H. Sayed. Fundamentals of adaptative filtering. 2003.

[11] PAULI TIKKANEN. Characterization and application of analysis methods for ecg and time intervalvariability data. 1999.

[12] Md. Maniruzzaman Uzzal Biswas. Removing power line interference from ecg signal using adaptive filterand notch filter. 2014.

[13] Ivaylo I. Christov Vessela T. Krasteva, Irena I. Jekova. Automatic detection of premature atrial contrac-tions in the electrocardiogram. 2006.

[14] Ahmad Rauf Subhani Wajid Muntaz. Adaptative noise cancellation: A comparison of adaptative filteringalgorithms aiming fetal ecg extraction. 2012.

[15] Juan Pablo Martinez Alba Martin-Yebra. Automatic diagnosis of strict left bundle branch block using awavelet-based approach. 2019.

[16] Lina Zhao Xiangyu Zhang Feifei Lieu, Chengyu Liu. An open access database for evaluating the algorithmsof electrocardiogram rhythm and morphology abnormality detection. 2018.

[17] Luning Mao Jieying Wei Jiangling Song Hao Chen, Jiaqi Bai and Rui Zhang. Automatic identification ofpremature ventricular contraction using ecgs. 2019.

[18] Jiaqi Bai Jieying Wei Qiang Li Luning Mao, Hao Chen and Rui Zhang. Automated detection of first-degreeatrioventricular block using ecgs. 2019.

[19] Fayyaz A. Afsar Muhammad Arif, IjazA. Malagore. Detection and localization of myocardial infarctionusing k-nearest neighbor classifier. 2010.

[20] Awadhesh Pachauri and Manabendra Bhuyan. Wavelet and energy based approach for pvc detection.2009.

[21] Richard B. Reilly Philip de Chazal, Maria O'Dwyer. Automatic classification of heartbeats using ecgmorphology and heartbeat interval features. 2004.

[22] Ivo Viscor Josef Halamek Filip Plesinger Magdalena Matejkova Radovan Smisek, Pavel Jurak. Automaticdetection of strict left bundle branch block. 2019.

[23] Hein J.J. Wellens Sinjin Lee. Paroxysmal atrioventricular block. 2009.

[24] Chih-Han Huang Tsai-Min Chen. Detection and classification of cardiac arrythmias by a challenge-bestdeep learning neural network model. 2020.

[25] Ivaylo I. Christov Vessela T. Krasteva, Irena I. Jekova. Automatic detection of premature atrial contrac-tions in the electrocardiogram. 2006

[26] Frank G. Yanowitz. Introduction to ecg interpretation. 2018

[27] P. Grassberger A. Kraskov, H. Stogbauer. Estimating mutual information. 2004.

[28] Mark A. Hall. Correlation-based feature selection for machine learning. 1999.

[29] Fuhui Long Hanchuan Peng and Chris Ding. Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy. 2005.

[30] N. N. Leonenko L. F. Kozachenko. Sample estimate of the entropy of a random vector:. 1987.

[31] B. C. Ross. Mutual information between discrete and continuous data sets. 2014

[32] Perez Alday EA, Gu A, Shah A, Robichaux C, Wong AKI,Liu C, Liu F, Rad BA, Elola A, Seyedi S, Li Q, Sharma A,Clifford GD, Reyna MA. Classification of 12-lead ECGs:the PhysioNet/Computing in Cardiology Challenge 2020.Physiological Measurement Under Review 2020;.

[33] Zheng J, Zhang J, Danioko S, Yao H, Guo H, Rakovski C. A12-lead electrocardiogram database for arrhythmia researchcovering more than 10,000 patients. Scientific Data 2020;7(1):1–8.

[34] Li Q, Mark RG, Clifford GD. Robust heart rate estima-tion from multiple asynchronous noisy sources using signalquality indices and a Kalman filter. Physiological Measure-ment 2007;29(1):15.

[35] Behar J, Oster J, Li Q, Clifford GD. ECG signal qualityduring arrhythmia and its application to false alarm reduc-tion. IEEE Transactions on Biomedical Engineering 2013;60(6):1660–1666.

[36] Zheng J, Fu G, Anderson K, Chu H, Rakovski C. A 12-LeadECG database to identify origins of idiopathic ventriculararrhythmia containing 334 patients. Scientific Data 2020;7(1):1–10.

[37] Marťınez JP, Almeida R, Olmos S, Rocha AP, Laguna P.A wavelet-based ECG delineator: evaluation on standarddatabases. IEEE Transactions on Biomedical Engineering2004;51(4):570–581.

[38] Oster J, Behar J, Sayadi O, Nemati S, Johnson AE, Clif-ford GD. Semisupervised ECG ventricular beat classifica-tion with novelty detection based on switching Kalman fil-ters. IEEE Transactions on Biomedical Engineering 2015;62(9):2125–2134.

[39] Krasteva VT, Jekova II, Christov II. Automatic detectionof premature atrial contractions in the electrocardiogram.Electrotechniques Electronics E E 2006;9(10).

[40] Chen H, Bai J, Mao L, Wei J, Song J, Zhang R. Auto-matic identification of premature ventricular contraction us-ing ECGs. International Conference on Health InformationScience 2019;143–156.

[41] Talbi M, Chare A. PVC discrimination using the QRSpower spectrum and self-organizing maps. Computer meth-ods and programs in biomedicine 2009;94(3):223–231.

[42] Marťın-Yebra A, Marťınez JP. Automatic diagnosis of strictleft bundle branch block using a wavelet-based approach.PloS one 2019;14(2):e0212971.

[43] Luning Mao HCea. Automated Detection of First-DegreeAtrioventricular Block Using ECGs. International Confer-ence on Health Information Science 2018;156–167.

[44] Dawson D, Yang H, Malshe M, Bukkapatnam ST, BenjaminB, Komanduri R. Linear affine transformations between 3-lead (Frank XYZ leads) vectorcardiogram and 12-lead elec-trocardiogram signals. Journal of Electrocardiology 2009;42(6):622–630.

[45] Chocron A, Oster J, Biton S, Franck M, Elbaz M, Y.Y.Z, Behar J.Remote atrial fibrillation burden estima-tion using deep recurrent neural network. arXiv preprintarXiv200802228 2020;.

[46] Peng H, Long F, Ding C. Feature selection based on mutualinformation: criteria of max-dependency, max-relevance,and min-redundancy. IEEE Transactions on Pattern Analy-sis and Machine Intelligence 2005;27(8):1226–1239

XV. Appendix

A. FEATURE EXTRACTION

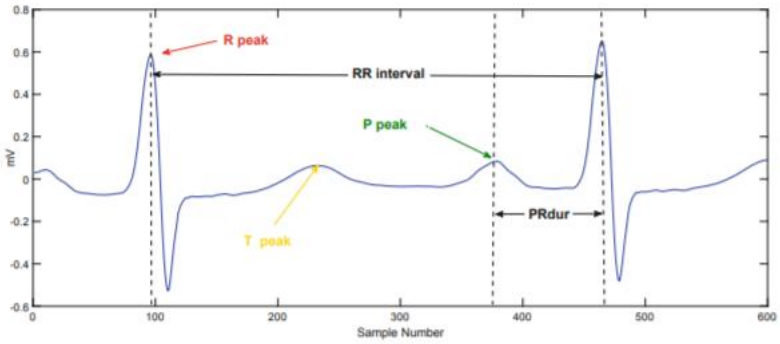
1. First degree Atrioventricular block

Pathology Presentation

First-degree atrioventricular block (AV block) is a disease of the electrical conduction system of the heart in which electrical impulses conduct from the cardiac atria to the ventricles through the atrioventricular node (AV node) more slowly than normal. First degree AV block not generally cause any symptoms, but may progress to more severe forms of heart block such as second- and third-degree atrioventricular block. It is diagnosed using an electrocardiogram, and is defined as a PR interval greater than 200 milliseconds.

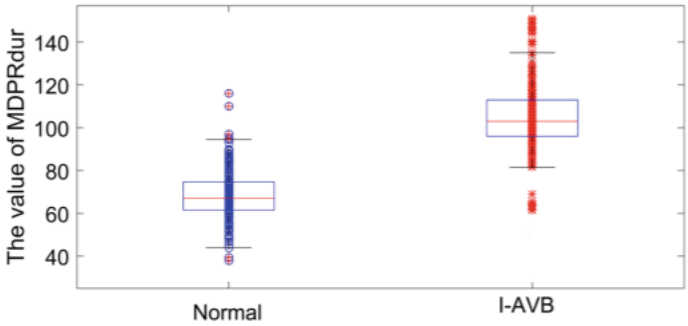
The features extracted focus especially on statistics related to these PR interval durations.

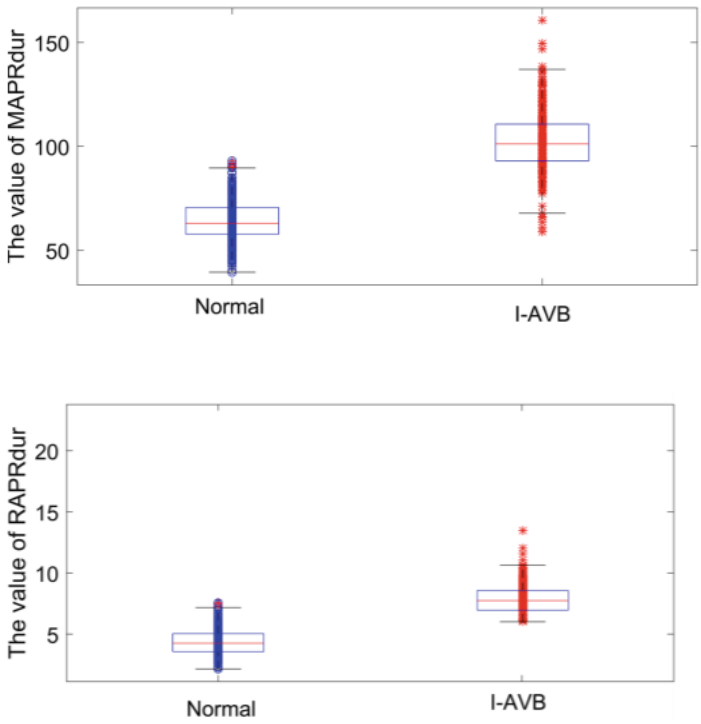
They define the PR interval as the interval separating the P-peak from the R-peak.



I extracted the 3 following features on lead 2:

- MAPR : Average PR interval duration
- MEDPR : Median PR interval duration
- RAPR : Renormalized PR durations



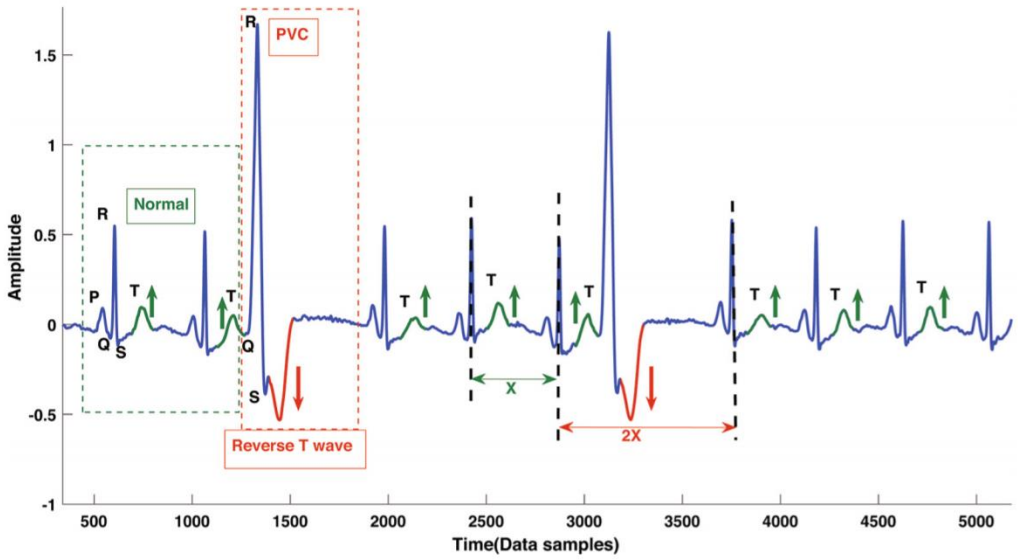


Therefore, those features allowed to differentiate well the Normal ECGs from the one with the AVB pathology.

2. Premature Ventricular Contraction

Pathology presentation

Premature ventricular contraction (PVC) is one of the most common arrhythmia diseases, which is caused by the ventricular activation in advance. Let us see what a PVC ecg looks like.

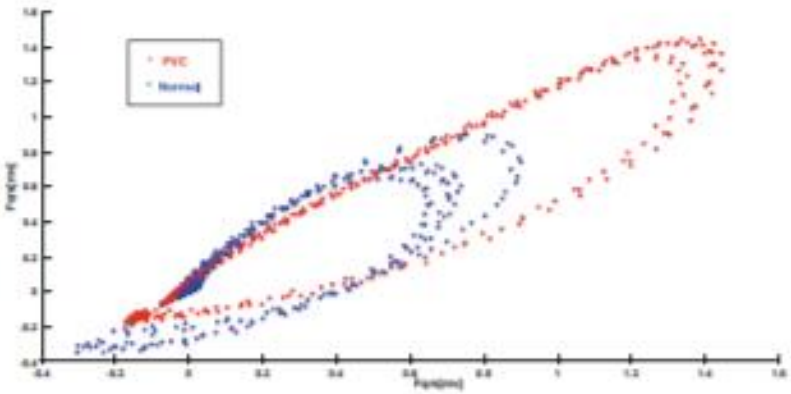


As we can see, PVC is characterized by several things:

- An advanced QRS-T complex (meaning that the RR intervals are not uniform and that we have compensatory pauses between peaks)
- High amplitude R-peaks
- A T-wave with opposite direction from the R-wave.
- A broad QRS complex

Features extracted

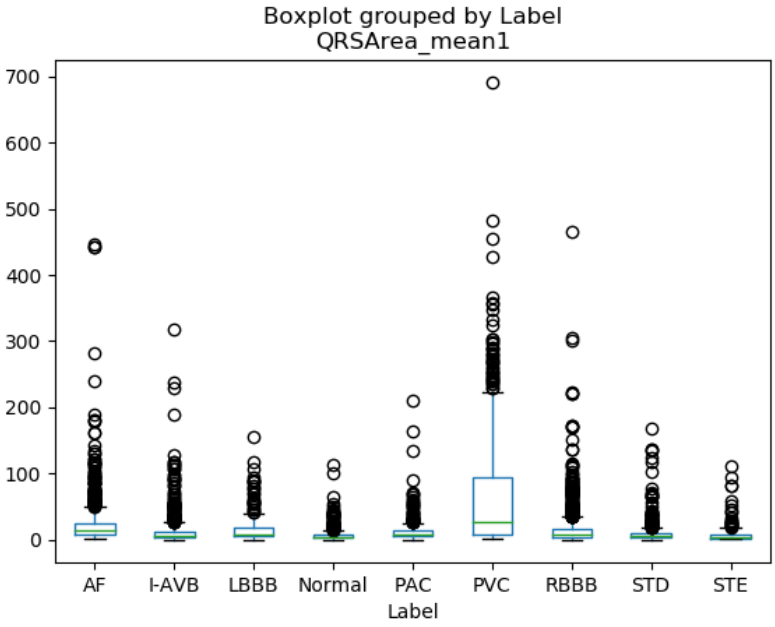
In order to characterize the high amplitude R-peaks, I extract the amplitude of R peaks and the difference between amplitudes. In order to quantify the broad QRS complex, I first focused on the area of the QRS complex, and then the duration of the QRS complex. Then a useful information comes when we plot the Poincare plot of PVC and normal examples.



We therefore see that using useful information from this ellipse will help us to differentiate PVC examples from normal ones: major axis of the ellipse.

Last, in order to quantify those compensatory pauses, I focused on the RR interval ratio, in order to measure the deviation from the mean RR interval ratio.

Therefore, we have 24 features for every of the 12 leads: 288 morphological features per example solely for PVC detection.



3. Premature Atrial Contraction

Pathology presentation

Premature atrial contractions (PACs), also known as atrial premature complexes (APC) or atrial premature beats (APB), are a common cardiac dysrhythmia characterized by premature heartbeats originating in the atria. While the sinoatrial node typically regulates the heartbeat during normal sinus rhythm, PACs occur when another region of the atria depolarizes before the sinoatrial node and thus triggers a premature heartbeat. PACs are often completely asymptomatic and may be noted only with Holter monitoring, but occasionally they can be perceived as a skipped beat or a jolt in the chest.

PACs can be detected in ECGs thanks to precise morphological characteristics:

- Hidden ectopic P-wave, or a different P-wave morphology
- The PR interval can be longer
- The QRS complex can be narrower (usually combined with BBB pathology)
- Incomplete pause after a PAC beat

Features extracted

We have extracted several features directly linked to the morphology of the ecg, and tried to characterize potential deviations in morphologies.

First, the pauses after a PAC beat have been engineered via the Interbeat RR-Interval Difference:

The difference in QRS global morphology have been engineered via the deviation from a reference QRSarea and QRSwidth:

The last feature extracted comes from a representation of the two-leads ecg: the vectorcardiographic plane.

The reference value is always computed as the median of the 5 previous values of this quantity in the signal.

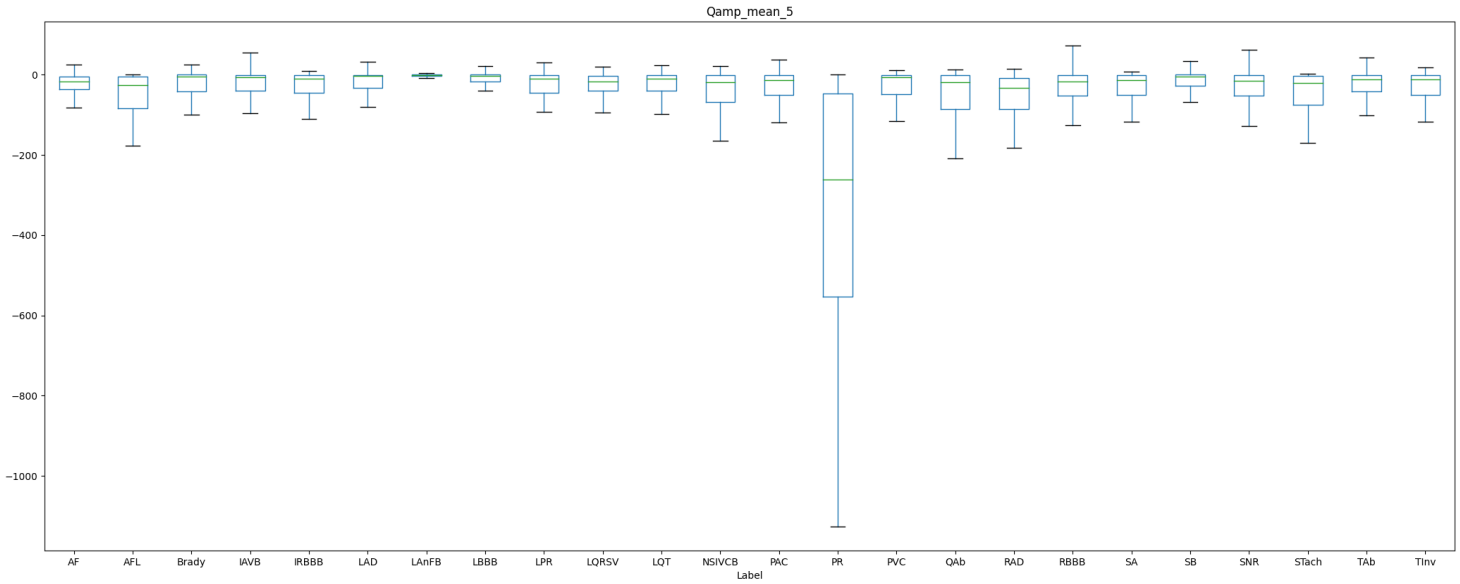
4. T-wave Abnormal and T wave Inversion

I extracted the frequency of T wave typology among every lead. The wavedet-3D classifies every lead's T-waves among {normal, inverted, upwards, downwards, biphasic}. Therefore, it accounts for the majority of abnormalities (except 'Camel-Hump T waves') and also for the T-wave inversion. Therefore, the frequency of every type of morphology in every lead should help to classify the example as if it presents a Twave abnormality (in the broad sense, also accounting for TInv) or not.

5. Q-wave abnormal

The Q-waves are pathologic if they are abnormally wide (>0.2 second) or abnormally deep (>5 mm). Therefore, I will extract features for Q wave duration and Q wave amplitude on every lead. The Q wave amplitude is the difference between the ecg amplitude on Q points and isoelectric lines. The duration of Q wave is the duration between the isoelectric point (QRSon) and the first point x_i such as ecg[x_i]=ecg[$QRSon^i$].

Therefore, I will extract 5 statistical features for the amplitudes of Q waves and Q waves durations: 10 new features on every lead.



As we can see, these new features will help us to differentiate QAb. Moreover, luckily for us, these features will help us differentiate Pacing Rhythm from other pathologies.

6. Left Anterior Fascicular Block

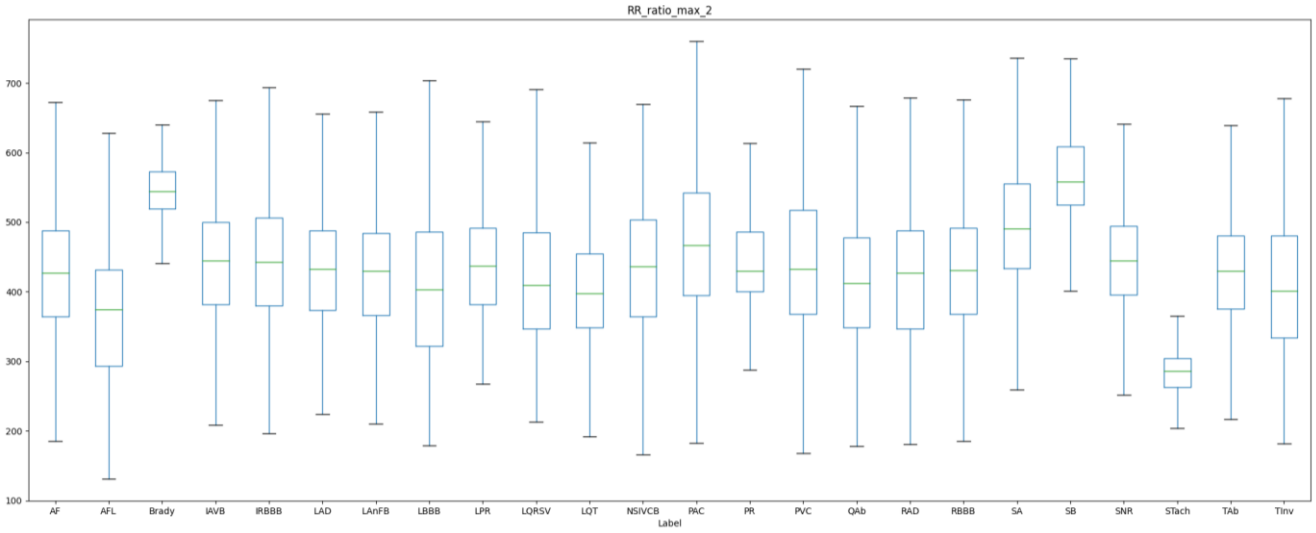
ECG characteristics of left anterior fascicular block include the following: QRS slight widening, left axis deviation, small Q and tall R in leads I and aVL (small Q not essential), deep S wave in leads II, III, and aVF (exceeding the R wave). These are some features I will be extracting for the new pathologies to be studied. I will only add the amplitude of the S_wave: 5 new features on every lead.

7. NonSpecific IntraVentricular Conduction disorder

Intraventricular conduction disorders are a group of conduction disturbances characterized by abnormalities in intraventricular conduction that leads to changes in shape, duration, and/or axis of the QRS complex on the electrocardiogram. Nonspecific intraventricular conduction disorder exists if the ECG displays a widened QRS appearance that is neither a LBBB nor a RBBB. According to the American Heart Association/ American College of Cardiology and the Heart Rhtyhm society recommendations, non specific intraventricular conduction disorder is defined by a ‘QRS duration greater than 110ms in adults, greater than 90ms in children without meeting the criteria of LBBB and RBBB’. Therefore, the features I have been extracting for PVC and PAC regarding QRS duration and Area should be efficient here.

8. Sinus Arrhytmia

The ECG criteria to diagnose sinus arrhythmia is a variation of the R-R interval, from one beat to the next, of at least 0.12 seconds, or 120 milliseconds. Therefore, the maximum of the RR interval should help us to see whether an example present the Sinus Arrhytmia or not.



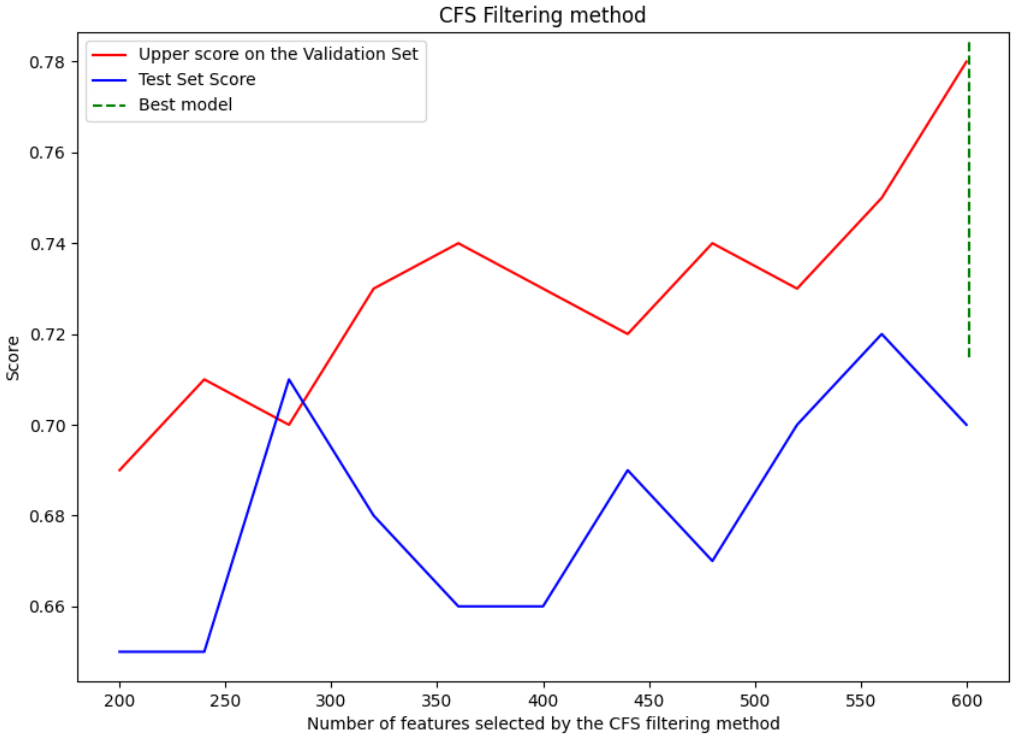
Therefore, we can see that the features of maximal duration between R peaks allow us to separate Sinus Arrhytmias from other pathologies.

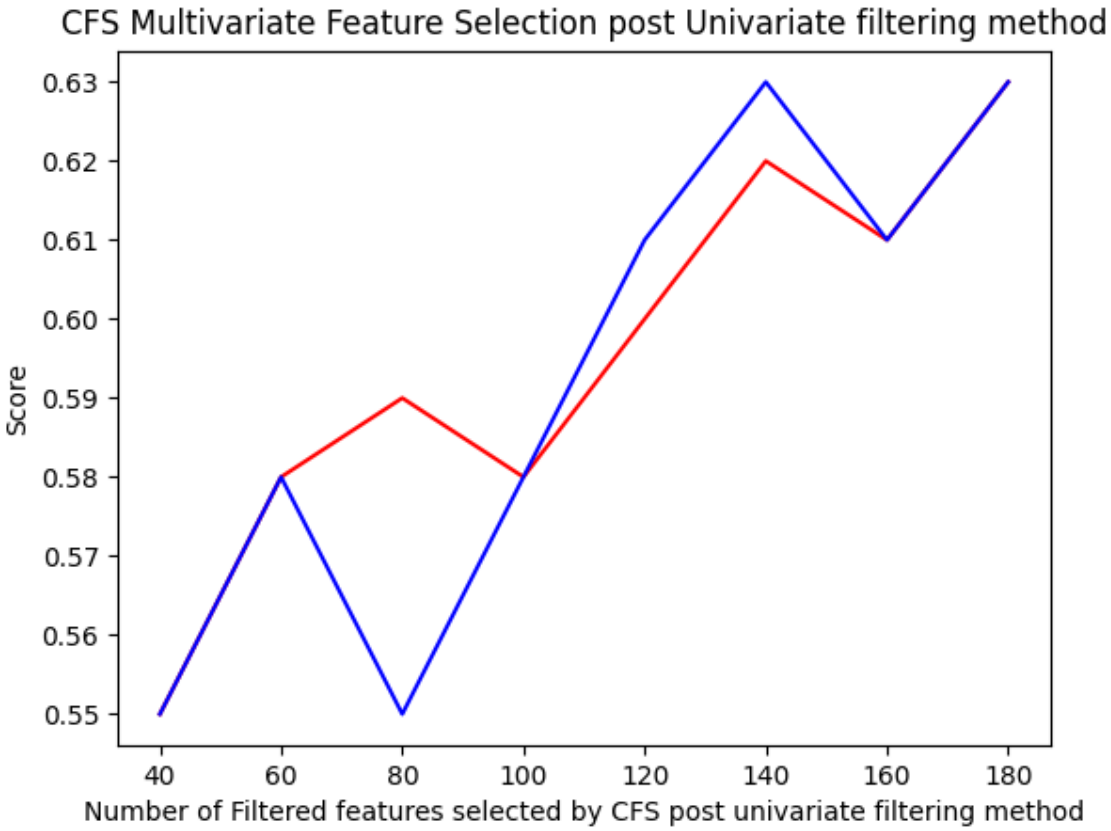
B. FEATURE SELECTION

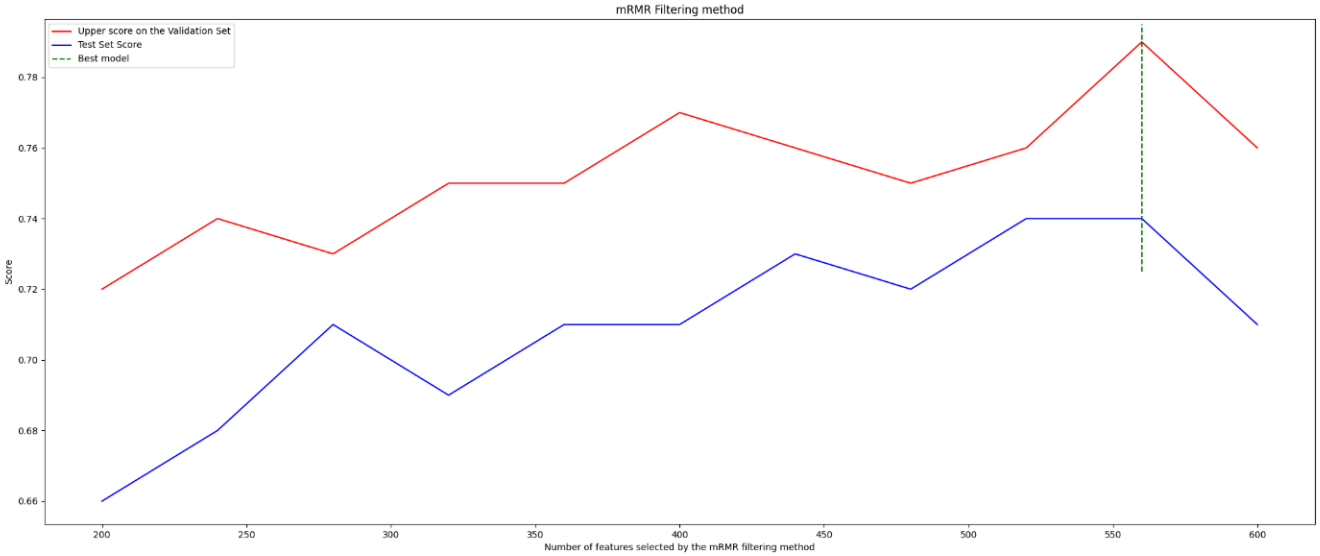
Table 2.16: Best Results of the implementation of Filtering methods for filter 4Bis

	Features	F_β validation set upper CI	F_β test set
0 Variance 40 Percentile 0.02 Spearsman 0.05 Kendall	348	0.79	0.77
0 Variance 40 Percentile 0.05 Spearsman 0.05 Kendall	353	0.79	0.78
0 Variance 50 Percentile 0.01 Spearsman 0.07 Kendall	427	0.78	0.76
0 Variance 50 Percentile 0.04 Spearsman 0.06 Kendall	433	0.79	0.76
0 Variance 50 Percentile 0.1 Spearsman 0.05 Kendall	433	0.80	0.78
0 Variance 60 Percentile 0.03 Spearsman 0.06 Kendall	516	0.78	0.76
0 Variance 70 Percentile 0.02 Spearsman 0.09 Kendall	593	0.79	0.76
0 Variance 80 Percentile 0.04 Spearsman 0.1 Kendall	675	0.79	0.76
0.01 Variance 30 Percentile 0.08 Spearsman 0.07 Kendall	256	0.78	0.76
0.01 Variance 40 Percentile 0.05 Spearsman 0.1 Kendall	339	0.79	0.76
0.01 Variance 40 Percentile 0.05 Spearsman 0.1 Kendall	339	0.79	0.76
0.01 Variance 40 Percentile 0.1 Spearsman 0.1 Kendall	340	0.79	0.76
0.01 Variance 60 Percentile 0.01 Spearsman 0.08 Kendall	490	0.78	0.77
0.01 Variance 60 Percentile 0.1 Spearsman 0.07 Kendall	499	0.80	0.76
0.01 Variance 70 Percentile 0.01 Spearsman 0.1 Kendall	563	0.79	0.77
0.01 Variance 70 Percentile 0.05 Spearsman 0.1 Kendall	575	0.80	0.77
0.01 Variance 80 Percentile 0.0001 Spearsman 0.08 Kendall	606	0.78	0.76
0.01 Variance 80 Percentile 0.01 Spearsman 0.08 Kendall	635	0.78	0.76
0.01 Variance 80 Percentile 0.02 Spearsman 0.06 Kendall	642	0.79	0.76
0.01 Variance 80 Percentile 0.07 Spearsman 0.09 Kendall	652	0.80	0.76
0.01 Variance 80 Percentile 0.04 Spearsman 0.1 Kendall	648	0.80	0.76
0.01 Variance 80 Percentile 0.1 Spearsman 0.1 Kendall	655	0.79	0.76
0.02 Variance 30 Percentile 0.03 Spearsman 0.06 Kendall	246	0.79	0.76
0.02 Variance 50 Percentile 0.0001 Spearsman 0.07 Kendall	390	0.78	0.76
0.02 Variance 50 Percentile 0.05 Spearsman 0.1 Kendall	405	0.78	0.76
0.02 Variance 50 Percentile 0.06 Spearsman 0.08 Kendall	405	0.78	0.77
0.02 Variance 60 Percentile 0.01 Spearsman 0.08 Kendall	476	0.78	0.76
0.02 Variance 50 Percentile 0.06 Spearsman 0.08 Kendall	405	0.78	0.77
0.02 Variance 60 Percentile 0.03 Spearsman 0.06 Kendall	482	0.78	0.76
0.02 Variance 50 Percentile 0.06 Spearsman 0.08 Kendall	405	0.78	0.77
0.02 Variance 60 Percentile 0.08 Spearsman 0.07 Kendall	485	0.80	0.77
0.02 Variance 70 Percentile 0.02 Spearsman 0.06 Kendall	555	0.79	0.77
0.02 Variance 70 Percentile 0.04 Spearsman 0.06 Kendall	559	0.78	0.76
0.02 Variance 70 Percentile 0.06 Spearsman 0.08 Kendall	561	0.77	0.77
0.02 Variance 80 Percentile 0.02 Spearsman 0.1 Kendall	626	0.78	0.77
0.02 Variance 80 Percentile 0.08 Spearsman 0.09 Kendall	637	0.79	0.76
0.03 Variance 30 Percentile 0.07 Spearsman 0.07 Kendall	246	0.78	0.77
0.02 Variance 80 Percentile 0.1 Spearsman 0.05 Kendall	634	0.78	0.77
0.03 Variance 40 Percentile 0.07 Spearsman 0.06 Kendall	325	0.79	0.76
0.03 Variance 50 Percentile 0.03 Spearsman 0.1 Kendall	398	0.78	0.76
0.03 Variance 60 Percentile 0.01 Spearsman 0.05 Kendall	468	0.78	0.77
0.03 Variance 60 Percentile 0.07 Spearsman 0.08 Kendall	478	0.79	0.76
0.03 Variance 70 Percentile 0.0001 Spearsman 0.07 Kendall	521	0.78	0.76
0.03 Variance 70 Percentile 0.02 Spearsman 0.08 Kendall	547	0.78	0.77
0.03 Variance 70 Percentile 0.04 Spearsman 0.07 Kendall	551	0.79	0.77
0.03 Variance 70 Percentile 0.06 Spearsman 0.07 Kendall	553	0.79	0.76
0.03 Variance 80 Percentile 0.06 Spearsman 0.07 Kendall	610	0.78	0.76
0.03 Variance 80 Percentile 0.03 Spearsman 0.06 Kendall	620	0.78	0.76

Multivariate Filtering Feature Selection result







C. CONSISTENCY BETWEEN DATABASES

We were concerned that the plurality of sources for our Competition Database would not help us in our classification task. Therefore, what we did is that we trained our model in an increasing fashion by always adding Databases. Here are our results